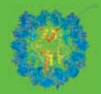


PACIFIC SYMPOSIUM ON
BIOCOMPUTING 2010

The Pacific Symposium on Biocomputing (PSB) 2010 is an international, multidisciplinary conference for the presentation and discussion of current research in the theory and application of computational methods in problems of biological significance. Presentations are rigorously peer reviewed and are published in an archival proceedings volume. PSB 2010 will be held on January 4 – 8, 2010 in Kohala Coast, Hawaii. Tutorials and workshops will be offered prior to the start of the conference.

PSB 2010 will bring together top researchers from the US, Asia Pacific, and around the world to exchange research results and address pertinent issues in all aspects of computational biology. It is a forum for the presentation of work in databases, algorithms, interfaces, visualization, modeling, and other computational methods, as applied to biological problems, with emphasis on applications in data-rich areas of molecular biology.

The PSB has been designed to be responsive to the need for critical mass in sub-disciplines within biocomputing. For that reason, it is the only meeting whose sessions are defined dynamically each year in response to specific proposals. PSB sessions are organized by leaders of research in biocomputing's "hot topics." In this way, the meeting provides an early forum for serious examination of emerging methods and approaches in this rapidly changing field.

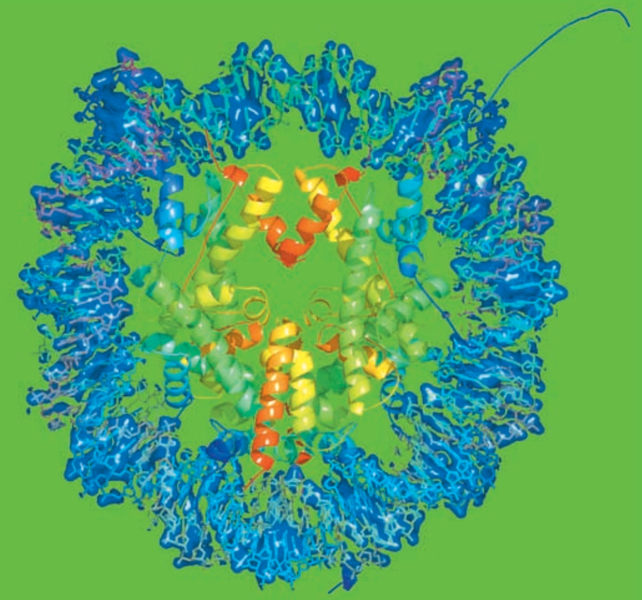
World Scientific
www.worldscientific.com
7628 hc

ISBN-13 978-981-4295-29-1(ebk)
ISBN-10 981-4295-29-9(ebk)

R. B. Altman
A. K. Dunker
L. Hunter
T. Murray
T. E. Klein

PACIFIC SYMPOSIUM ON
BIOCOMPUTING 2010

PACIFIC SYMPOSIUM ON BIOCOMPUTING 2010



Edited by

Russ B. Altman, A. Keith Dunker,
Lawrence Hunter, Tiffany Murray & Teri E. Klein

Cover image:
This image depicts a molecular model of the Nucleosome (PDB ID: 1aoi, Luger *et al.* (1997) *Nature* 389, 251–260) — The nucleosome is the organising principle behind higher ordered chromatin structure. The histone core of the nucleosome exemplifies the many molecular mechanisms that have evolved to regulate access to the DNA in chromatin.

Image by D. Rey Banatao,
Pacific Symposium on Biocomputing.

PACIFIC SYMPOSIUM ON BIOCOMPUTING 2010

This year marks the 15th year of PSB. Started in 1996 by Teri Klein and Larry Hunter, the meeting was a session within the Hawaii International Conference on Systems Sciences Conference (HICSS) for a couple of years. The interest in biocomputing was great and was threatening to alter the balance of attendees within the HICSS meeting, and so the organizers asked Teri and Larry to find an alternative. Taking advantage of the nice meeting opportunities in Hawaii immediately following New Year's Day, they created PSB with a simple formula: oral presentation of peer reviewed papers in emerging areas of biocomputing, interactive poster sessions, and lively discussion sessions on these topics. They invited Keith Dunker and Russ Altman to join the team for the second PSB, and we have been working together ever since.

In fifteen years, we have had some papers that have made remarkable impact. Using Google Scholar, the top 15 papers, in terms of citations over the last 15 years are listed below (with number of citations listed as of September 2009, followed by the reference). Obviously, there are other great papers, especially ones that have been written in the last few years, that have not yet had time to rise to the top of this list. Nevertheless, the list provides highlights of the last 15 years in biocomputing challenges.

(573) REVEAL, A General Reverse Engineering Algorithm for Inference of GeneticNetwork Architectures S. Liang, S. Fuhrman and R. Somogyi; Pacific Symposium on Biocomputing 3:18-29 (1998).

(457) Modeling Gene Expression with Differential Equations T. Chen, H. L. He, and G.M. Church; Pacific Symposium on Biocomputing 4:29-40 (1999).

(422) Principal Components Analysis to Summarize Microarray Experiments: Application to Sporulation Time Series S. Raychaudhuri, J.M. Stuart, and R.B. Altman; Pacific Symposium on Biocomputing 5:452-463 (2000).

(419) The Spectrum Kernel: A String Kernel for SVM Protein Classification Leslie, E. Eskin, and W.S. Noble; Pacific Symposium on Biocomputing 7:566-575 (2002).

(294) EDGAR: Extraction of Drugs, Genes and Relations from the Biomedical Literature T.C. Rindflesch, Lorraine Tanabe, John N. Weinstein, and L. Hunter; Pacific Symposium on Biocomputing 5:514-525 (2000).

(270) Mutual Information Relevance Networks: Functional Genomic Clustering Using Pairwise Entropy Measurements A. J. Butte and I. S. Kohane; Pacific Symposium on Biocomputing 5:415-426 (2000).

(245) Biobibliometrics: Information Retrieval and Visualization from Co-Occurrences of Gene Names in Medline Abstracts B.J. Stapley and G. Benoit; Pacific Symposium on Biocomputing 5:526-537 (2000).

(218) Hybrid Fold Recognition: Combining Sequence Derived Properties with Evolutionary Information D. Fischer; Pacific Symposium on Biocomputing 5:116-127 (2000).

(215) Hybrid Petri Net Representation of Gene Regulatory Network H. Matsuno, A. Doi, M. Nagasaki, and S. Miyano; Pacific Symposium on Biocomputing 5:338-349 (2000).

(198) Discovery of Regulatory Interactions Through Perturbation: Inference and Experimental Design T.E. Ideker, V. Thorsson, and R.M. Karp; Pacific Symposium on Biocomputing 5:302-313 (2000).

(152) Cluster Analysis and Data Visualization of Large-Scale Gene Expression Data G.S. Michaels, D.B. Carr, M. Askenazi, S. Fuhrman, X. Wen and R. Somogyi; Pacific Symposium on Biocomputing 3:42-53 (1998).

(191) ANN-Spec: A Method for Discovering Transcription Factor Binding Sites with Improved Specificity C.T. Workman and G.D. Stormo; Pacific Symposium on Biocomputing 5:464-475 (2000).

(137) Development of a System for the Inference of Large Scale Genetic Networks Y. Maki, D. Tominaga, M. Okamoto, S. Watanabe, and Y. Eguchi; Pacific Symposium on Biocomputing 6:446-458 (2001).

(130) A Comparison of Genetic Network Models L.F.A. Wessels, E.P. Van Someren, and M.J.T. Reinders; Pacific Symposium on Biocomputing 6:508-519 (2001).

(126) Detecting Gene Relations from MEDLINE Abstracts M. Stephens, M. Palakal, S. Mukhopadhyay, R. Raje, and J. Mostafa; Pacific Symposium on Biocomputing 6:483-496 (2001).

We have seen PSB papers cited in the other "high impact journals" and it is satisfying to know that the work presented here is making a real difference. We have spoken with scientists who hesitate to submit their work to PSB for fear that it will not be subsequently consider eligible for a journal. This is an understandable concern, but is mitigated by the excellent penetration of PSB paper as they are listed in Medline and clearly generate some attention, if they are good.

We would like to thank our keynote speakers. Dr. Stanley Nelson, Professor of Human Genetics and Psychiatry at the David Geffen School of Medicine at UCLA will talk about "Whole genome sequencing: issues and progress towards common application." Our keynote in the area of Ethical, Legal and Social Implications of Technology will be Dr. Isaac Kohane, Professor of Pediatrics and Medicine, Harvard Medical School who will discuss issues related to the ethical application of emerging technical capabilities.

PSB provides sessions focusing on emerging areas in biomedical computation. These sessions are frequently conceived at the meeting itself, during the discussion times. Again, the efforts of a dedicated group of researchers has led to an outstanding set of sessions, with associated introductory tutorials. These organizers provide the scientific core of PSB, and their sessions are as follows:

Bernard Moret, Webb Miller, Pavel Pevzner, and David Sankoff
Computational Challenges in Comparative Genomics

Rolf Backofen, Hamidreza Chitsaz, Ivo Hofacker, S. Cenk Sahinalp, and Peter F. Stadler
Computational studies of non-coding RNAs

Tanya Berger-Wolf, Teresa Przytycka, Mona Singh, and Donna Slonim
Dynamics of Biological Networks

Samuel Flores, Julie Bernauer, Xuhui Huang, Ruhong Zhou, and Seokmin Shin
Multi-resolution Modeling of Biological Macromolecules

Can Alkan, Michael Brudno, Evan E. Eichler, Maricel G. Kann, S. Cenk Sahinalp
Personal Genomics

Gil Alterovitz, Silvio Cavalcanti, Taro M. Muso, Marco F. Ramoni, and May Wang
Reverse Engineering and Synthesis of Biomolecular Systems

In addition, we welcome three satellite workshops to PSB this year:

Richard Goldstein, Phil Husbands, Chrisantha Fernando, and Dov Stekel
In silico biology

Peter Sterk, Lynette Hirschman, Dawn Field, John Wooley
Metagenomics, Metadata and Metaanalysis (m3)

Adrien Coulet, Nigam Shah, Larry Hunter, Chitta Baral and Russ B. Altman
GPD-Rxn Workshop: Genotype-Phenotype-Drug Relationship Extraction from Text

Tiffany Murray continues to run the peer review process and assembly of the proceedings. We thank the National Institutes of Health and the International Society for Computational Biology (ISCB) for travel grant support. We are particularly grateful to BJ McKay-Morrison at ISCB for her assistance. We also acknowledge the many busy researchers who reviewed the submitted manuscripts on a very tight schedule. The partial list following this preface does not include many who wished to remain anonymous, and of course we apologize to any who may have been left out by mistake.

We look forward to a great meeting once again.

Aloha!

Pacific Symposium on Biocomputing Co-Chairs,
September 28, 2009

Russ B. Altman
Departments of Bioengineering, Genetics & Medicine, Stanford University

A. Keith Dunker
Department of Biochemistry and Molecular Biology, Indiana University School of Medicine

Lawrence Hunter
Department of Pharmacology, University of Colorado Health Sciences Center

Teri E. Klein
Department of Genetics, Stanford University

Thanks to the reviewers…

Finally, we wish to thank the scores of reviewers. PSB requires that every paper in this volume be reviewed by at least three independent referees. Since there is a large volume of submitted papers, paper reviews require a great deal of work from many people. We are grateful to all of you listed below and to anyone whose name we may have accidentally omitted or who wished to remain anonymous.

Gil Alterovitz
Nice-Sophia Antipolis
Zafer Aydin
Rolf Backofen
Tanya Berger-Wolf
Julie Bernauer
Olivier Bodenreider
Philip Bourne
Greg Bowman
Sharon Browning
Catarina D. Campbell
Alessandra Carbone
Bernard Maigret Carloni
Silvio Cavalcanti
Hamid Chitsaz
John Chodera
Yann Christinat
James Collins
Anne Condon
Gregory M. Cooper
Markus Covert
Anneleen Daemen
Frédéric Dayan
Lisette G. dePillis
Ruxandra Dima
Chuong Do
Son Doan
Nilgun Donmez
Ivan Erill
Nicolas Ferey
Christoph Flamm
Samuel Flores
Wolfgang Gatterbauer
Daniel Gautheret
Romain Gauthier
Robert Giegerich
Santhosh Girirajan
Jérôme Golebiowski
Jan Gorodkin
Andreas Gruber
Habiba
Joerg Hackermueller
Iman Hajirasouliha
Sihyun Ham
Ulrich Hansmann
Robert Harris
Steffen Heyne
Paul Higgs
Andreas Hildebrandt
Ivo Hofacker
Fereydoun Hormozdiari
David Hsu
Hsien-Da Huang
Xuhui Huang
Changbong Hyeon
Andy Itsara
Soonmin Jang
Joël Janin
Xiaoye Jiang
Helen Johnson
Mads Kaern
Ulas Karaoz
Rachel Karchin
Janet Kelso
Attila Kertesz-Farkas
Akio Kitao
Diana Kolbe
Robert Lacey
Mayank Lahiri
Doug Lauffenburger
Sonia Leach
Seunghak Lee
Yuhui Li
JC Liao
Pu Liu
Xiong Liu
Sébastien Loriot
Paolo Magni
Arun Maiya
François Major
Nawar Malhis
Leonardo Marino-Ramirez
Kshitij Marwah
David Mathews
Graham McVicker
Webb Miller
David Mobeley
Bernard Moret
Richard Morris
Alan Moses
Vincent Moulton
Yuguang Mu
Taro Muso
Chris J. Myers
Arcadi Navarro
Karen Nelson
Eric K. Neumann
Bill Noble
Ruth Nussinov
Yuko Okamoto
Anna Panchenko
Marc Parisien
Nicholas Pattengale
Dmitri Pervouchine
Pavel Pevzner
Son Pham
David Pincus
Jed W Pitera
Predrag Radivojac
Ben J. Raphael
Jens Reeder
Noah Rosenberg
Cenk Sahinalp
Yasubumi Sakakibara
Raheleh Salari
David Sankoff
Michael Schnieders
Gunnar Schroeder
Chaok Seok
Keith Sevcik
Saad Sheikh
Robert D. Skeel
Christina Smolke
Peter Stadler
Jens Stoye
Peter Sudmant
Ryan Taft
Shoji Takada
Chayant Tantipathananandh
Ron Taylor
May Wang
Todd Wareham
Robert Warren
Bruce Weir
John Wilbur
Sebastian Will
Tom Woolf
Eric Xing
Wei Yang
Yuan Yao
Seungtai Yoon
Michaela Zavolan
Troy Zerr
Ziuwei Zhang

CONTENTS

WORKSHOPS

COMPUTATIONAL CHALLENGES IN COMPARATIVE GENOMICS SESSION INTRODUCTION

BERNARD M.E. MORET

Laboratory for Computational Biology and Bioinformatics
EPFL (Swiss Federal Institute of Technology)
EPFL-IC-LCBB INJ 230, Station 14, CH-1015 Lausanne, Switzerland
E-mail: bernard.moret@epfl.ch

WEBB C. MILLER

Department of Biology
Pennsylvania State University
University Park, PA 16823, USA
E-mail: wcm2@psu.edu

PAVEL A. PEVZNER

Department of Computer Science & Engineering
University of California, San Diego
9500 Gilman Drive, Mail Code 0404, La Jolla, CA 92093-0404, USA
E-mail: ppevzner@cs.ucsd.edu

DAVID SANKOFF

Department of Mathematics and Statistics
University of Ottawa
585 King Edward, Ottawa, ON K1N 6N5, Canada
E-mail: sankoff@uottawa.ca

Comparative methods have long been a mainstay of biology, particularly evolutionary biology; they are also at the core of medical research based on animal models of human physiology. They find their most challenging and most fitting application, however, in the study of whole genomes, as they are the main tools through which we can study the billions of base-pairs forming the sequence of animal and other genomes. Comparing whole genomes, which is necessarily done through computational methods due to the size of the genomes, has given rise to the research area known as comparative genomics.

Comparative genomics is the tool of choice for identifying genes in both well studied and newly sequenced genomes; for studying the acquisition of virulence or drug resistance in pathogens; for tracking down gene complexes responsible for inheritable diseases or susceptibilities; and for engineering desirable new traits in crops; and for studying many forms of cancers. More generally, comparative genomics is the tool of choice to elucidate how the genetic blueprint translates into specific functions and how that blueprint evolves in populations and into various species.

Comparative genomics uses not just whole-genome sequences, but also dense single-nucleotide polymorphism (SNP) maps, genetic maps, and sequences of individual genes, but it is characterized by its emphasis on a whole-genome approach. Its computational methods include combinatorial optimization, machine learning, and data mining, while much work has also been devoted to visualization of its findings—witness, for example, the many spectacular full-color figures illustrating the correspondences between the human and mouse genomes.

The focus of our session is on computational models and algorithms; this session at PSB'10 follows a previous session on the same theme at PSB'09, which also featured five papers. The five papers included in our session all exemplify the genome-wide approach of the area.

Two of the papers focus on ancestral reconstruction, a topic that has recently attracted much interest, but where assessing the validity of results is obviously very difficult. Hickey and Blanchette, in "A practical algorithm for estimation of the maximum likelihood ancestral reconstruction expected error," provide the first

systematic approach to such an assessment, using a direct analog of the phylogenetic bootstrapping process. Gavranovic and Tannier, in "Guided genome halving: provably optimal solutions provide good insights into the preduplication ancestral genome of S. cerevisiae," discuss a specialized application of ancestral inference in which, given a contemporary genome whose lineage is known to have seen a whole-genome duplication, and a closely related genome whose lineage diverged before that duplication and serves as a guide, the preduplication ancestor is inferred.

Two other papers are concerned with metagenomics, where samples are taken of an entire biota (seawater, soil, animal gut, etc.), the samples sequenced, and the resulting sequences (mostly unassembled) placed within a phylogeny of related organisms—a process that has already enabled us to discover very large numbers of new species. Clemente, Jansson, and Valiente, in "Accurate taxonomic assignment of short pyrosequencing reads," puth forth the proposition that the common strategy of assigning metagenomic sequences to the root of an entire clade can be advantageously replaced by a strategy of assigning these sequences to internal nodes that optimize some ROC characteristics. and demonstrate of the use of their strategy on marine and gut data. Essinger and Rosen, in "Benchmarking BLAST accuracy of genus/phylum classification of metagenomic reads," also address the question of proper assignment of metagenomic reads to a phylogeny, but examine the even more common strategy of identifying subsequences with high similarity by using BLAST on an entire database.

Finally, the fifth paper showcases a fascinating application of comparative genomics to what might at first be viewed as a problem in population genetics: how to optimize the choice of a founder population to repopulate a species through captive breeding. Miller, Wright, Zhang, Schuster, and Hayes, in "Optimization methods for selecting founder populations for the captive breeding of endangered species," present formulations and algorithms for selecting a founder population from an existing wild population. Such problems deal with very small populations and target specific collections of alleles, many of which will be represented in only a few individuals—thus genomic methods, which deal with individual genomes, are better suited than standard population genetics methods, which tend to deal with distributions over sizeable populations.

We are very pleased to feature such work at this PSB'10 session and want to take this opportunity to thank attendees, presenters, all submitting authors, and the referees who together made it possible.

ACCURATE TAXONOMIC ASSIGNMENT OF SHORT PYROSEQUENCING READS

JOSÉ C. CLEMENTE

Center for Information Biology and DNA Databank of Japan
National Institute of Genetics
Yata 1111, Mishima, Japan
E-mail: jclement@lab.nig.ac.jp

JESPER JANSSON

Graduate School of Humanities and Sciences
Ochanomizu University
2-1-1 Otsuka, Bunkyo-ku, Tokyo 112-8610, Japan
E-mail: jesper.jansson@ocha.ac.jp

GABRIEL VALIENTE

Algorithms, Bioinformatics, Complexity and Formal Methods Research Group
Technical University of Catalonia
E-08034 Barcelona, Spain
E-mail: valiente@lsi.upc.edu

Ambiguities in the taxonomy dependent assignment of pyrosequencing reads are usually resolved by mapping each read to the lowest common ancestor in a reference taxonomy of all those sequences that match the read. This conservative approach has the drawback of mapping a read to a possibly large clade that may also contain many sequences not matching the read. A more accurate taxonomic assignment of short reads can be made by mapping each read to the node in the reference taxonomy that provides the best precision and recall. We show that given a suffix array for the sequences in the reference taxonomy, a short read can be mapped to the node of the reference taxonomy with the best combined value of precision and recall in time linear in the size of the taxonomy subtree rooted at the lowest common ancestor of the matching sequences. An accurate taxonomic assignment of short reads can thus be made with about the same efficiency as when mapping each read to the lowest common ancestor of all matching sequences in a reference taxonomy. We demonstrate the effectiveness of our approach on several metagenomic datasets of marine and gut microbiota.

Background

The advent of next-generation sequencers has been accompanied by new computational challenges to deal with the ever increasing amounts of data produced.[1] In particular, metagenomic analysis of microbial communities[2,3] has resulted in a plethora of tools for comparative studies, such as determining the richness and diversity of communities,[4–7] or test their similarity.[8–11]

A more fundamental problem is how to determine the composition of a particular community given the set of pyrosequencing reads obtained from a sample, that is, what species (or strains) are present, and in what proportion. Two strategies have been proposed, based on whether a taxonomy is assumed or not. *Binning* approaches discard the use of bacterial taxonomies since they tend to be biased towards cultivable species, and apply instead some unsupervised classification method (clustering) on the reads to determine the structure of the population. Self-Organizing Maps,[12,13] Support Vector Machines,[14] z-score correlations,[15] or nearest neighbors[16] have been successfully utilized for this purpose. *Taxonomy-based* approaches, on the other hand, map the reads to known species in a given taxonomy, usually based on the 16S rRNA. Ambiguous fragments that cannot be unequivocally assigned to a specific taxon are mapped to an inner node of the taxonomy, usually the lowest common ancestor (LCA) of all sequences to which the read might be assigned.[17–20]

Both binning and taxonomy-based methods need to define a measure to compare sequences. *Similarity-based* methods use sequence identity to determine how alike sequences are: BLAST[18,19,21] and number of mismatches[22–25] are commonly used measures. *Sequence composition* methods use instead intrinsic features of the sequences to determine their similarity, such as their GC-content[26] or k-nucleotide frequencies.[14,15,20]

In this paper, we address the problem of how to assign ambiguous short reads with a taxonomy-based

approach and a measure of similarity based on the number of mismatches between sequences. The hidden assumption made by previous studies when assigning these fragments to the LCA is that a higher coverage should be preferred to a higher accuracy (see Fig. 1). Our work is a generalization aimed at maximizing the F-measure in order to assign ambiguous reads at inner nodes of the taxonomy that are not necessarily the LCA. Notice that the use of the F-measure in this context is just one of the several possible assignment strategies and it does not reflect the accuracy of the global assignment schema, which would also include unambiguously assigned reads and be affected by the chosen measure of similarity between fragments.

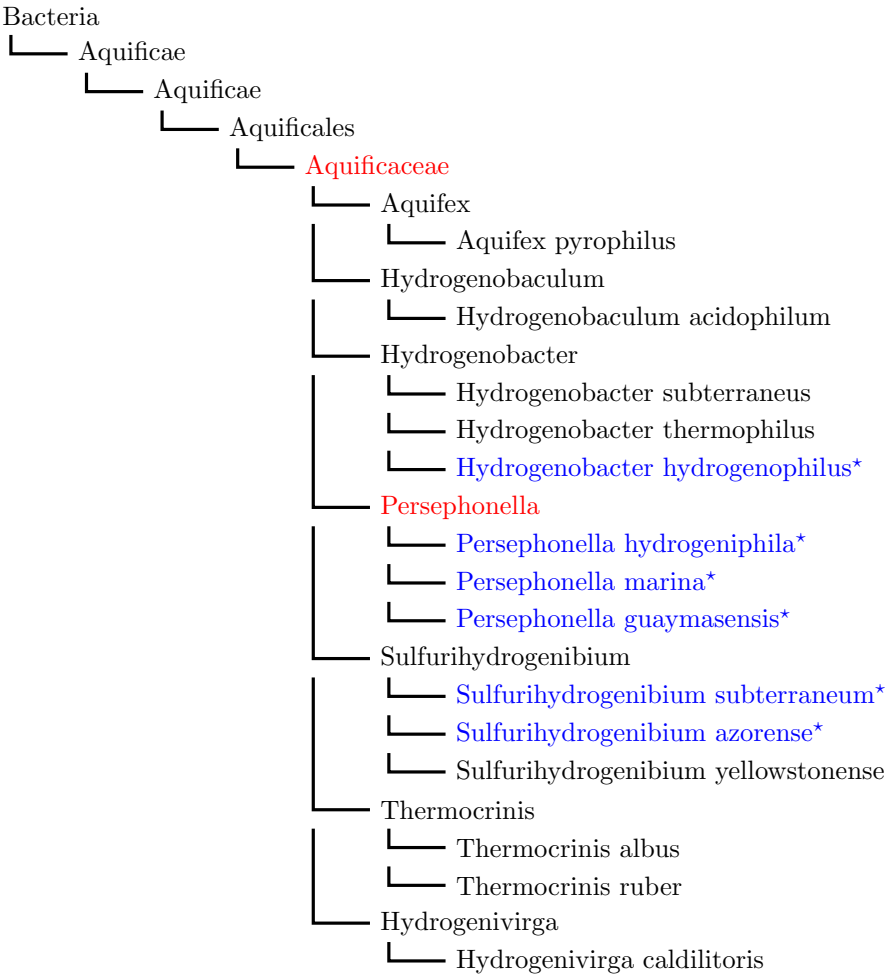


Fig. 1. Coverage and accuracy of assigning ambiguous reads to the LCA. The assignment of an ambiguous read to the family *Aquificaceae*, the LCA of the six matching taxa *H. hydrogenophilus*, *P. hydrogeniphila*, *P. marina*, *P. guaymasensis*, *S. subterraneum*, and *S. azorense*, marked up with a star, has a 100% coverage (recall) but a $6/14 = 43\%$ accuracy (precision). The assignment to the genus *Persephonella*, instead, has a coverage of $3/6 = 50\%$ and an accuracy of 100%.

Methods

Given a reference taxonomy T, a set R of short reads, and a threshold value k of sequence similarity, let R_i be the ith read, let M_i be the leaves of T matching R_i with up to k mismatches, let T_i be the subtree of T rooted at the lowest common ancestor of M_i, and let N_i be the leaves of T_i not matching R_i with up to k mismatches. Let also $L_i = M_i \cup N_i$.

Further, consider some arbitrary, but fixed, ordering of the nodes of T, say in postorder, let $T_{i,j}$ be the subtree of T rooted at the jth node of T_i in postorder, let $M_{i,j}$ be the leaves of $T_{i,j}$ matching R_i with up to k mismatches, and let $N_{i,j}$ be the leaves of $T_{i,j}$ not matching R_i with up to k mismatches.

For the ith read and the jth node of T_i in postorder, the leaves of T_i can be partitioned in the following four subsets (see Fig. 2):

- $TP_{i,j} = M_{i,j}$ (true positives)
- $FP_{i,j} = N_{i,j}$ (false positives)
- $TN_{i,j} = N_i \setminus N_{i,j}$ (true negatives)
- $FN_{i,j} = M_i \setminus M_{i,j}$ (false negatives)

Then, the precision of classifying R_i as T_j is $P_{i,j} = |TP_{i,j}|/(|TP_{i,j}| + |FP_{i,j}|)$, and the recall is $R_{i,j} = |TP_{i,j}|/(|TP_{i,j}| + |FN_{i,j}|)$. The combined F-measure of precision and recall is $F_{i,j} = 2P_{i,j}R_{i,j}/(P_{i,j} + R_{i,j})$.

It is easy to see that $F_{i,j} = 2P_{i,j}R_{i,j}/(P_{i,j} + R_{i,j}) = 2|TP_{i,j}|/(2|TP_{i,j}| + |FP_{i,j}| + |FN_{i,j}|) = 2|TP_{i,j}|/(|TP_{i,j}| + |FP_{i,j}| + |M_i|) = 2|M_{i,j}|/(|L_{i,j}| + |M_i|)$. This gives a simple algorithm for computing the best possible taxonomic rank to which each read can be assigned, in time linear in the size of T_i. Given the set M_i of matching sequences for a read R_i, it suffices to compute the sets $L_{i,j}$ and $M_{i,j}$ for each node j in T_i during a bottom-up traversal of T_i.[27,28] Notice that it takes time linear in the size of M_i to find the root of T_i, because T has constant height, and no additional preprocessing of T is required.[29]

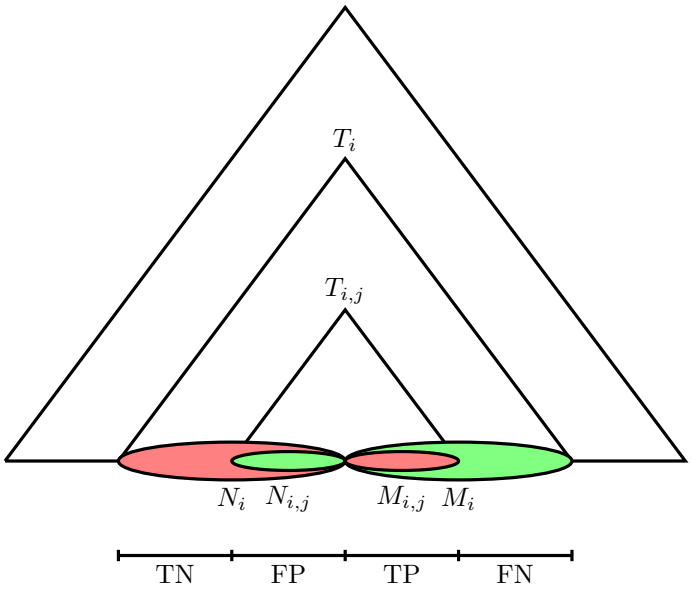


Fig. 2. Precision and recall of assigning the ith read to the jth node of T_i in postorder.

Results and Discussion

It is not completely clear yet what microbial community structure different environments possess, but results so far seem to indicate a high degree of variability both in environmental[30] and gut samples,[31] with significant differences between gut and other microbiomes.[32] The distribution of functions seems to be more conserved though,[33] indicating that a core functionality can be achieved through different species distributions. Understanding this correlation requires accurate measurements of both variables, and our work aims at reducing the amount of error introduced by the assignment of ambiguous fragments to the LCA of a group of species.

While feature-based and binning approaches[13,14] require long fragments (more than 1K bp), taxonomical methods can work with shorter reads, which can be as effective as longer sequences for taxonomic assignment provided that the region of the 16S rRNA is adequately chosen.[34,35] The algorithm we introduce here is also very efficient and can process large number of fragments in time at most linear in the number of reference sequences for each fragment, providing a useful tool to quickly test hypotheses about microbial communities.

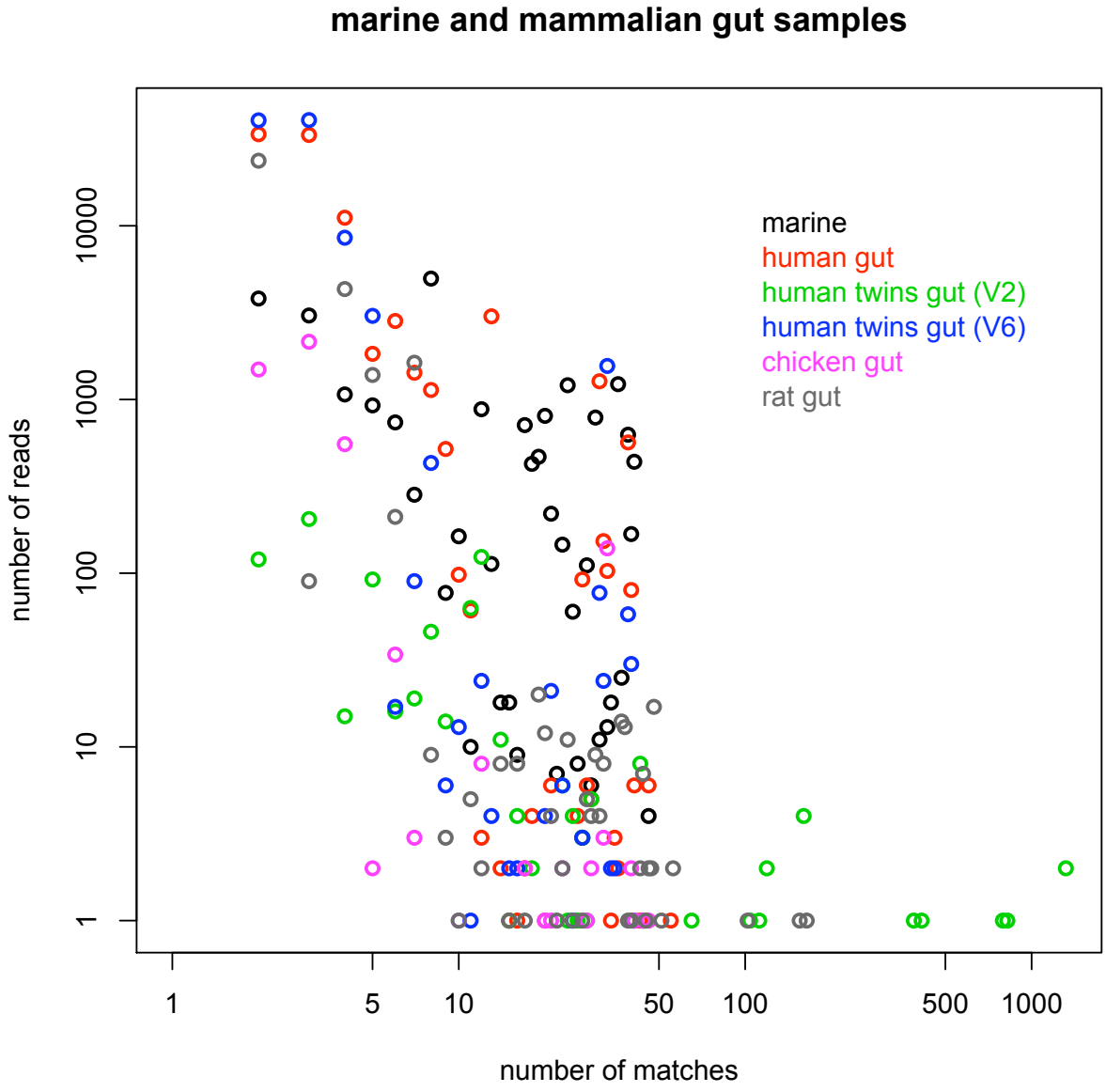


Fig. 3. Distribution of the 23,621 marine, 91,335 human gut, 776 human twins gut (V2 region), 94,999 human twins gut (V6 region), 4,395 chicken gut, and 31,509 rat gut pyrosequencing reads ambiguously matched with up to 2 mismatches to two or more of the 5,165 sequences in the reference bacterial taxonomy.

In order to demonstrate the effectiveness of our approach, we have studied taxonomic assignment in several microbial communities: marine,[30] human gut,[36] lean and obese human twins gut,[31] chicken gut,[37]

and rat gut[38] samples. The samples themselves contain 454 pyrosequencing tags for a variable region of 16S rRNA, between 50 and 329 bp in length, and for each of these bacterial communities, we have used both the LCA approach and our approach to assign each of the pyrosequencing reads at the best possible taxonomic rank, using a reference bacterial taxonomy of 5,165 near-full-length type cultures of high quality[17] with a uniform scheme of seven taxonomic ranks (domain, phylum, class, order, family, genus, species).

The taxonomy covers the whole spectrum of known bacteria, and the dominant phyla are Proteobacteria, Actinobacteria, Firmicutes, Bacteroidetes, and Tenericutes, with 1,925, 1,285, 1,178, 355, and 160 species, respectively. The near-full-length 16S reference sequences range from 1,202 to 1,780 bp.

The marine (V6 region) samples themselves range from 50 to 100 bp, with an average length of 62 bp; the human gut (V6 and V3 regions) samples range from 50 to 165 bp, average 101 bp; the human twins gut (V2 region) samples range from 50 to 317 bp, average 231 bp; the human twins gut (V6 region) samples range from 50 to 129 bp, average 60 bp; the chicken gut (V6 region) samples range from 55 to 75 bp, average 60 bp; and the rat gut (V4 region) samples range from 50 to 329 bp, with an average length of 231 bp.

We have built a suffix array for the 5,165 sequences in the reference bacterial taxonomy using the GEM-do-index tool,[39] and have matched each of the pyrosequencing reads to the 5,165 reference sequences using the GEM-mapper tool[39] with appropriate parameter settings for finding all matching sequences with up to 2 mismatches, which is about 99% identity for reads of 200 bp. The distribution of those pyrosequencing reads that could not be unambiguously matched to a single sequence in the reference bacterial taxonomy is given in Fig. 3.

The pyrosequencing reads that matched two or more sequences in the reference bacterial taxonomy were assigned to the LCA of the matching sequences in the taxonomy, and they were also assigned at the best possible taxonomic rank using our method. The distribution of reads assigned at the taxonomic rank of domain, phylum, class, order, family, and genus using the LCA of the matching sequences in the taxonomy is shown in Table 1, and the distribution of reads assigned at the taxonomic rank of class, order, family, genus, and species using the new method is shown in Table 2.

Table 1. Number of ambiguous pyrosequencing reads assigned at various taxonomic ranks using the LCA of the matching sequences in the reference bacterial taxonomy of 5,165 sequences.

taxonomic rank	number of reads					
	marine V6	human V6, V3	twins V2	twins V6	chicken V6	rat V4
domain			40			1
phylum	29	5,498	3	13,133	130	49
class	12,099	2,354		1,854	154	3
order	976	5	13	8	8	35
family	3,428	49,647	371	2,343	1,441	3,582
genus	7,089	33,831	349	77,661	2,662	27,839
	23,621	91,335	776	94,999	4,395	31,509

Table 2. Number of ambiguous pyrosequencing reads assigned at various taxonomic ranks using our method in the reference bacterial taxonomy of 5,165 sequences.

taxonomic rank	number of reads					
	marine V6	human V6, V3	twins V2	twins V6	chicken V6	rat V4
class			2			
order			4			2
family	860	2,150	16	195	3	57
genus	17,705	8,441	411	2,353	210	3,622
species	5,056	80,744	343	92,451	4,182	27,828
	23,621	91,335	776	94,999	4,395	31,509

These results show that only 3,213 of the 23,621 marine ambiguous reads (13.60%), 4,231 of the 91,335 human gut ambiguous reads (4.63%), 35 of the 776 human twins gut (V2 region) ambiguous reads (4.51%), 635 of the 94,999 human twins gut (V6 region) ambiguous reads (0.67%), 45 of the 4,395 chicken gut ambiguous reads (1.02%), and 48 of the 31,509 rat gut ambiguous reads (0.15%) were actually assigned to the LCA of the matching sequences using our method.

The remaining 96.67% of the ambiguous reads were assigned at a deeper taxonomic rank than the LCA of the matching sequences using the new method. While assigning a read to the LCA of the matching sequences in the taxonomy tends to produce assignments at the ranks of class, order, family, and genus, the new method produces more accurate assignments at the ranks of genus and species.

Conclusions

We have shown in this paper that ambiguities in the taxonomy dependent assignment of pyrosequencing reads can be resolved in an accurate way by mapping each read to the node of a reference taxonomy with the best combined value of precision and recall, in time linear in the size of the taxonomy subtree rooted at the lowest common ancestor of the matching sequences, given a suffix array for the sequences in the reference taxonomy. We have demonstrated the effectiveness of this approach on several metagenomic datasets of marine and gut microbiota, by showing that most reads are actually assigned at a deeper taxonomic rank than the LCA of the matching sequences in the reference taxonomy.

The experimental results were obtained using a reference bacterial taxonomy of 5,165 near-full-length type cultures of high quality.[17] The incompleteness and bias towards cultivable species of the taxonomy might affect these results. Most species in the gut of an individual are rare,[40] and the microbiome has a small number of deep-branching taxa with large diversity at the leaves, with different humans showing different patterns of abundance of microbial species.[41] As our knowledge of the human microbiome expands, we expect the number of unclassified species to diminish and the effectivenes of taxonomical methods to improve consequently.

Acknowledgements

JC was supported by Grant-in-Aid for JSPS Fellows from the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan, No. 2008086. JJ was supported by the Special Coordination Funds for Promoting Science and Technology. GV was supported by the Spanish government and the EU FEDER program under projects MTM2006-07773 COMGRIO and PCI2006-A7-0603. We want to thank Chaysavanh Manichanh for several discussions on the topic of this paper, and Paolo Ribeca for developing GEM.[39]

References

1. J. Shendure and H. Ji, *Nature Biotechnology* **26**, 1135 (2008).
2. P. J. Turnbaugh, R. E. Ley, M. Hamady, C. M. Fraser-Liggett, R. Knight and J. I. Gordon, *Nature* **449**, 804 (2007).
3. J. Venter, K. Remington, J. Heidelberg, A. Halpern, D. Rusch, J. Eisen, D. Wu, I. Paulsen, K. Nelson, W. Nelson, D. Fouts, S. Levy, A. Knap, M. Lomas, K. Nealson, O. White, J. Peterson, J. Hoffman, R. Parsons, H. Baden-Tillson, C. Pfannkoch, Y. Rogers and H. Smith, *Science* **304**, 66 (2004).
4. P. D. Schloss and J. Handelsman, *Applied and Environmental Microbiology* **71**, 1501 (2005).
5. P. D. Schloss and J. Handelsman, *Applied and Environmental Microbiology* **72**, 6773 (2006).
6. P. D. Schloss and J. Handelsman, *BMC Bioinformatics* **9**, p. 34 (2008).
7. V. Seguritan and F. Rohwer, *BMC Bioinformatics* **2**, p. 9 (2001).
8. C. Lozupone and R. Knight, *Applied and Environmental Microbiology* **71**, 8228 (2005).
9. A. Martin, *Applied and Environmental Microbiology* **68**, 3673 (2002).
10. P. D. Schloss and J. Handelsman, *Applied and Environmental Microbiology* **72**, 2379 (2005).
11. D. Singleton, M. Furlong, S. Rathbun and W. Whitman, *Applied and Environmental Microbiology* **67**, 4374 (2001).
12. T. Abe, H. Sugawara, M. Kinouchi, S. Kanaya and T. Ikemura, *DNA Research* **12**, 281 (2005).

13. C. Martin, N. N. Diaz, J. Ontrup and T. W. Nattkemper, *Bioinformatics* **24**, 1568 (2008).
14. A. C. McHardy, H. G. Martín, A. Tsirigos, P. Hugenholtz and I. Rigoutsos, *Nature Methods* **4**, 63 (2007).
15. H. Teeling, A. Meyerdierks, M. Bauer, R. Amann and F. O. Glöckner, *Environmental Microbiology* **6**, 938 (2004).
16. N. N. Diaz, L. Krause, A. Goesmann, K. Niehaus and T. W. Nattkemper, *BMC Bioinformatics* **10**, p. 56 (2009).
17. J. R. Cole, Q. Wang, E. Cardenas, J. Fish, B. Chai, R. J. Farris, A. S. Kulam-Syed-Mohideen, D. M. McGarrell, T. Marsh, G. M. Garrity and J. M. Tiedje, *Nucleic Acids Research* **37**, 141 (2009).
18. D. H. Huson, A. F. Auch, J. Qi and S. C. Schuster, *Genome Research* **17**, 377 (2007).
19. Z. Liu, T. Z. DeSantis, G. L. Andersen and R. Knight, *Nucleic Acids Research* **36**, p. e120 (2008).
20. Q. Wang, G. M. Garrity, J. M. Tiedje and J. R. Cole, *Applied and Environmental Microbiology* **73**, 5261 (2007).
21. L. Krause, N. N. Diaz, A. Goesmann, S. Kelley, T. W. Nattkemper, F. Rohwer, R. A. Edwards and J. Stoye, *Nucleic Acids Research* **36**, 2230 (2008).
22. B. Langmead, C. Trapnell, M. Pop and S. L. Salzberg, *Genome Biology* **10**, p. R25 (2009).
23. H. Li and R. Durbin, *Bioinformatics* **25**, 1754 (2009).
24. R. Li, Y. Li, K. Kristiansen and J. Wang, *Bioinformatics* **24**, 713 (2008).
25. N. Malhis, Y. Butterfield, M. Ester and S. J. M. Jones, *Bioinformatics* **25**, 6 (2009).
26. H. G. Martín, N. Ivanova, V. Kunin, F. Warnecke, K. W. Barry, A. C. McHardy, C. Yeates, S. He, A. A. Salamov, E. Szeto, E. Dalin, N. H. Putnam, H. J. Shapiro, J. L. Pangilinan, I. Rigoutsos, N. C. Kyrpides, L. L. Blackall, K. D. McMahon and P. Hugenholtz, *Nature Biotechnology* **24**, 1263 (2006).
27. G. Valiente, *Algorithms on Trees and Graphs* (Springer, 2002).
28. G. Valiente, *Combinatorial Pattern Matching Algorithms in Computational Biology using Perl and R* (Taylor & Francis/CRC Press, 2009).
29. M. A. Bender, M. Farach-Colton, G. Pemmasani, S. Skiena and P. Sumazin, *Journal of Algorithms* **57**, 75 (2005).
30. M. L. Sogin, H. G. Morrison, J. A. Huber, D. M. Welch, S. M. Huse, P. R. Neal, J. M. Arrieta and G. J. Herndl, *Proc. Natl. Acad. Sci. USA* **103**, 12115 (2006).
31. P. J. Turnbaugh, M. Hamady, T. Yatsunenko, B. L. Cantarel, A. Duncan, R. E. Ley, M. L. Sogin, W. J. Jones, B. A. Roe, J. P. Affourtit, M. Egholm, B. Henrissat, A. C. Heath, R. Knight and J. I. Gordon, *Nature* **457**, 480 (2009).
32. R. E. Ley, C. Lozupone, M. Hamady, R. Knight and J. I. Gordon, *Nature Reviews Microbiology* **6**, 776 (2008).
33. E. A. Dinsdale, R. A. Edwards, D. Hall, F. Angly, M. Breitbart, J. M. Brulc, M. Furlan, C. Desnues, M. Haynes, L. Li, L. McDaniel, M. A. Moran, K. E. Nelson, C. Nilsson, R. Olson, J. Paul, B. R. Brito, Y. Ruan, B. K. Swan, R. Stevens, D. L. Valentine, R. V. Thurber, L. Wegley, B. A. White and F. Rohwe, *Nature* **452**, 629 (2008).
34. Z. Liu, C. Lozupone, M. Hamady, F. D. Bushman and R. Knight, *Nucleic Acids Research* **35**, p. e120 (2007).
35. C. Manichanh, C. E. Chapple, L. Frangeul, K. Gloux, R. Guigó and J. Dore, *Nucleic Acids Research* **36**, 5180 (2008).
36. L. Dethlefsen, S. Huse, M. L. Sogin and D. A. Relman, *PLoS Biology* **6**, p. e280 (2008).
37. VAMPS, Visualization and analysis of microbial population structure project, AGT_CKN_Bv6—Chicken intestinal microbiota, (2009).
38. C. Manichanh, Rat intestinal microbiota, Private communication, (2009).
39. P. Ribeca, GEM—GEnomic Multi-tool, http://gemlibrary.sourceforge.net/, (2009).
40. L. Dethlefsen, M. McFall-Ngai and D. A. Relman, *Nature* **449**, 811 (2007).
41. R. E. Ley, D. Peterson and J. I. Gordon, *Cell* **124**, 837 (2006).

BENCHMARKING BLAST ACCURACY OF GENUS/PHYLA CLASSIFICATION OF METAGENOMIC READS

STEVEN D. ESSINGER AND GAIL L. ROSEN

Electrical & Computer Engineering, Drexel University, 3141 Chestnut Street
Philadelphia, PA 19141, USA

Metagenomics is the study of environmental samples. Because few tools exist for metagenomic analysis, a natural step has been to utilize the popular homology tool, BLAST, to search for sequence similarity between sample fragments and an administered database. Most biologists use this method today without knowing BLAST's accuracy, especially when a particular taxonomic class is under-represented in the database. The aim of this paper is to benchmark the performance of BLAST for taxonomic classification of metagenomic datasets in a supervised setting; meaning that the database contains microbes of the same class as the 'unknown' query fragments. We examine well- and under-represented genera and phyla in order to study their effect on the accuracy of BLAST. We conclude that on fine-resolution classes, such as genera, the accuracy of BLAST does not degrade very much with under-representation, but in a highly variant class, such as phyla, performance degrades significantly. Our analysis includes five-fold cross validation to substantiate our findings.

1. Introduction

The relatively new field of metagenomics has been rapidly expanding over the past several years [1, 2]. This field focuses on DNA obtained from an environmental sample rather than from pure cultures in a laboratory. This markedly substantial difference from conventional microbial genomics poses a unique set of problems that are now gaining attention. Instead of asking the question "How does one organism work?" we are now interested in "Who is here in this sample and what are they doing?". Since greater than 99% of microbes cannot be cultivated in isolation [3], metagenomics is a necessity if we wish to understand the microbial diversity of our planet.

Examples of metagenomic applications include human health, soil fertility and forensics. The National Institute of Health has created an initiative called The Human Microbiome Project to examine microbes associated with health in several areas of the human body [2]. For example it is hypothesized that the human gastrointestinal tract contains microbes that outnumber human cells 10 to 1 [2]. Many of these microbes are believed to be involved with the digestive process. Most of these microbes cannot be isolated in the laboratory. Therefore they cannot be cultured for abundance so that their DNA can be extracted and amplified for genomic analysis. Instead we turn to metagenomics where we obtain the DNA of the environmental sample, extract and amplify the DNA, sequence the samples, assemble the samples and finally attempt to annotate the sequences. Annotation is certainly an elusive task since we do not know which microbes are in the sample to begin with. So we turn to sequence alignment tools such as BLAST [4, 5] which aid us in answering a fundamental question in metagenomics, namely "Who is here?". Before we can fully trust the results of BLAST for taxonomic classification, we seek to benchmark how database representation affects its performance.

2. Background on Taxonomy

Answering the question "Who is here?" is an issue of taxonomy. Taxonomy refers to the science of naming and classifying organisms. The National Center for Biotechnology Information (NCBI) maintains a taxonomy database which is considered a well respected source by the scientific community for taxonomic information [4]. The standard hierarchy of the taxonomy used in this paper is Phyla, Order, Family, Genus, Species, Strains as recommended by the NCBI. As of September 2009 there are over 339,500 taxa represented in the database. Of these taxa 968 are completely sequenced genomes of microbial organisms. Clearly, this is only a small fraction of the microbes inhabiting our planet today, however, the databases are expanding rapidly and as the field of metagenomics becomes more pervasive we shall see substantial increases in the number of taxa maintained in these databases.

When an organism's DNA or metagenomic sample has been sequenced it is a natural step to compare this new sequence to existing, annotated sequences in the databases for similarity [6, 7]. BLAST (Basic Local Alignment Search Tool) is both a web based and standalone tool developed by the NCBI for comparing sequence similarity

between two nucleotide or protein sequences [5]. The most popular way researchers use the tool is to input a sequence as a query against the public sequence databases which include NCBI Taxonomy (http://www.ncbi.nlm.nih.gov/Taxonomy/). BLAST returns sequences that are similar to the input query. BLAST will attempt to align the query with the sequences in the databases and then issue a statistical report to provide a level of confidence in the alignment. BLAST is actively maintained by the NCBI and can be found here (http://blast.ncbi.nlm.nih.gov/Blast.cgi).

The first alignment in the report returned by BLAST is supposedly the sequence in the database with the greatest similarity to the query sequence. When the query sequence is small (e.g. < 500bp), BLAST tends to produce multiple ambiguous top-hits. It has been found that the closest BLAST hit is often not the nearest-neighbor [8]. Generally speaking, microbiologists rely on the BLAST results without question [9, 10, 11]. Researchers have now begun to analyze and compare the performance of BLAST for metagenomic datasets. The findings are indicating that classifying genome sequence fragments based on the best BLAST hit only yield reliable results if there are close relatives represented in database for comparison [12, 13].

3. Method

A total of 635 distinct microbial strains downloaded in 2008 from the NCBI Genbank database were considered for our experiments. We have found that each of the 635 strains in our database can be classified to one of 19 different phyla and 272 different genera. In order to partition the database for our experiments we decided to focus on two well-represented and two under-represented classes each for the levels of phyla and genus. Thus two separate experiments were performed; one for the level of phyla and the other for genus. Table 1 shows the composition of each class for each experiment.

Table 1. The class composition for the phyla and genus five-fold cross validation experiments are provided below. A total of 463 strains were included in the phyla experiment. We chose to use two phyla having well-representation and two having under-representation in the database. For example, Proteobacteria (well) accounted for 315 (68%) of the 463 strains included in the experiment. These strains were partitioned into five groups each containing 63 strains. The remaining three classes were partitioned in the same manner ensuring that approximately 20% of the strains belonging to the class were in each group. The first group from all four classes was combined and BLAST against the remaining four groups. This procedure was repeated five times so that each group was used for query once. An identical procedure was used at for the genus experiment.

Phyla					
Total Strains – 463		Database (80%) – 370		Query (20%) – 93	
Well-Represented			Under-Represented		
Class	# of Strains	# Queries Sampled	Class	# of Strains	# Queries Sampled
Proteobacteria (well1)	315 (68%)	63	Crenarchaeota (under1)	15 (3%)	3
Fermicutes (well2)	116 (25%)	23	Tenericutes (under2)	17 (4%)	4
Genus					
Total Strains – 64		Database (80%) – 51		Query (20%) – 13	
Well-Represented			Under-Represented		
Class	# of Strains	# Queries Sampled	Class	# of Strains	# Queries Sampled
Steptococcus (well1)	26 (40%)	5	Yersinia (under1)	10 (16%)	2
Staphylococcus (well2)	18 (28%)	4	Synechococcus (under2)	10 (16%)	2

The two well-represented classes were chosen to be the two classes at each level that contained the greatest amount of microbial strains. For example, the phyla class Proteobacteria contained 315 strains out of the 635 strains in the overall database. The two under-represented classes were chosen arbitrarily so that they each contain no more than 20 strains. Many classes in the database contained only 1 strain; however the five-fold cross validation statistical measure necessarily requires that we have a minimum of 5 strains. We chose under-represented classes containing 10 to 17 strains as shown in Table 1.

The five-fold cross validation experiments proceeded as the following for phyla using 500bp query fragments. The identical procedure was followed for the level of genus thus yielding two separate experiments. The distribution

of the classes for the experiments can be found in Table 1. To measure the possible effect of class bias, we also performed an equal class representation experiment at the phyla level as discussed in the results section.

Since we have chosen five-fold cross validation, we randomly partitioned the strains from each class into five groups. The first group from each class was combined to create a set of query strains. To simulate a metagenomics dataset obtained using the next generation of 454 pyrosequencing technology [14], each query strain's genome was randomly sampled extracting 100 fragments each 500bp in length. Each fragment was annotated with its membership class so that we could determine if BLAST correctly matched the fragment. These sampled fragments were used as queries for BLAST sequence alignment. The whole-genomes of the remaining strains were used to construct the BLAST training database in which BLAST would attempt to align against the query sequences. For example, in the phyla experiment, 93x100 (20%) query fragments were BLAST against a database of 370 (80%) whole-genomes comprised of the remaining strains belonging to the 4 phyla. The percent accuracy is calculated as the number of query fragments correctly identified by BLAST over the total number of query fragments. This procedure was repeated a total of five repetitions so that each strain was in the query test set once. The results from the five partitions were averaged and the standard deviation was calculated. A survey of cross validation methods can be found from these sources [15, 16, 17].

BLAST may potentially return multiple ambiguous hits meaning that all of the top scores returned have the same statistical expect value (e-value). In these instances all of the aligned sequences must be from the true taxonomic class otherwise the BLAST result was marked incorrect for the corresponding query sequence. Additionally, BLAST may not return a report for a query sequence that it has determined to be a low-complexity region. In these few instances we marked the query as incorrect. While this filter may be turned off we've found that BLAST consumes significantly more resources; therefore we've chosen to leave it in the default setting.

4. Results

4.1 Well/Under Representation Experiments

The results of the two cross-validation experiments with well/under representation are summarized in Table 2. Each experiment had four classes; two classes that were well-represented by strains in the dataset and two classes that were under-represented. The percent accuracy is the number of strain fragments that BLAST matched with the correct class over the total number of fragments. The average score reported is the average of all five repetitions of the cross validation experiment. The standard deviation is calculated in a similar manner. Individual scores for each repetition, for all experiments are provided in Appendix A.1.

In addition to the percent accuracy of BLAST across all strains for each experiment, Table 2 lists the accuracy of BLAST on the four individual classes as well as the accuracy on the combined well and under represented classes. Each of these combined groups contains two classes.

Table 2. The percent accuracy scores of BLAST for the genus and phyla experiments are provided below. BLAST was marked correct if it matched the query fragment to the correct class and incorrect otherwise. It was also marked incorrect if it provided multiple ambiguous hits whereupon these hits belonged to two or more different classes. The percent accuracy for each cross validation repetition is the number of correct matches over the total number of query fragments. The percent accuracy scores over all five repetitions were average and are provided below along with the standard deviation of scores.

Percent Accuracy		All	Well	Well 1	Well 2	Under	Under 1	Under 2
500bp Genus	AVG	95.87	96.60	95.65	97.87	94.15	96.90	91.40
	STD	2.10	3.10	4.91	3.03	3.57	4.56	8.51
500bp Phyla	AVG	87.21	90.06	92.67	83.01	48.74	36.80	59.38
	STD	2.29	2.30	0.79	7.80	9.64	16.43	14.52

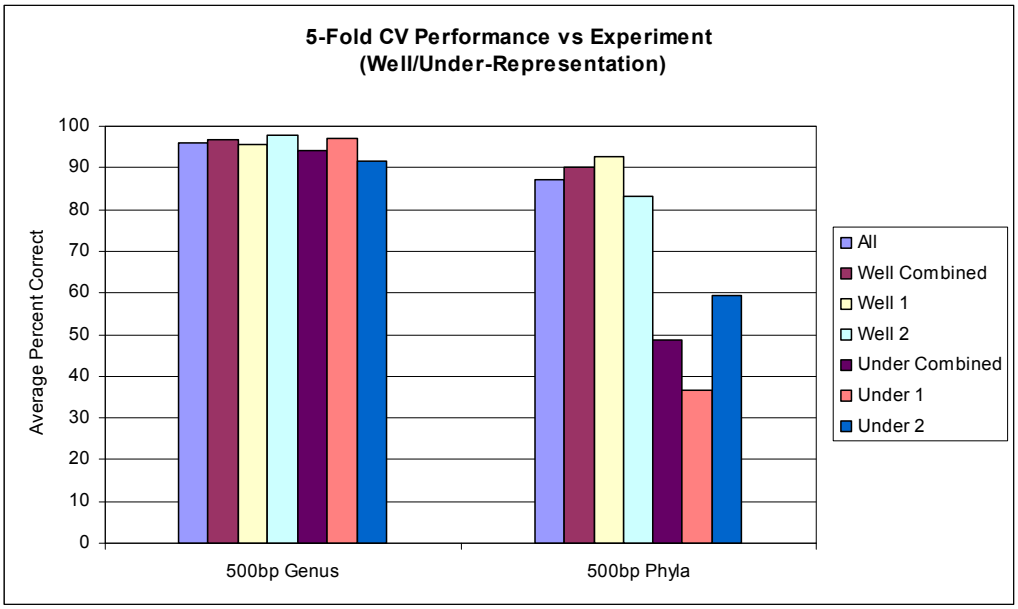


Figure 1. This bar graph illustrates the data provided in Table 2. All four classes in the genus experiment exhibited similar percent accuracy scores. However, there is a clear difference in percent accuracy between the well- and under-represented classes in the phyla experiment. We've found that this is due in part to the genus level having less diversity than the phyla level.

All seven different scores for percent accuracy are plotted against the two experiments in Figure 1. The trend across the two experiments indicates that accuracy increasingly diminishes moving from 500bp genus to 500bp phyla. It is evident from Figure 1 that the percent accuracy of all strains for each experiment is highly dependent on the accuracy of BLAST correctly identifying the fragments belonging to strains having membership in the under-represented classes. This is evident from the disparity between the under-represented scores for genus and phyla. Genus under-represented has an accuracy nearly 40% higher than the phyla under-represented group. Predictably, genus percent accuracy for all strains is nearly 10% higher than phyla.

4.1.1 Phyla

There were a total of 463 strains considered in the phyla experiments. Each cross-validation repetition consisted of 93 (20%) strains chosen at random without replacement to BLAST against the remaining 370 (80%) strains in the dataset (see Table 1). Accordingly, every strain in the dataset was used as a test strain once. Each test strain was sampled randomly 100 times; each sample consisted of a fragment 500bp in length. These 100 fragments were used in place of the test strain. The phyla experiment shows that well-represented strains scored approximately 40% higher than under-represented strains.

Strains belonging to the under-represented class Crenarchaeota were misclassified 78% on average. These misclassified fragments were frequently matched with strains belonging to the well-represented phyla. For example, BLAST classified 74% of fragments belonging to *Pyrobaculum aerophilum* to Proteobacteria (well) rather than Crenarchaeota (under) as expected.

In general, when BLAST misclassified fragments, 5% of the misclassifications belonged to a strain in the under-represented classes. Of the remaining 95% misclassified fragments, approximately 72% of the misclassifications went to strains belonging Proteobacteria (well) alone. This is similar to chance since the database is comprised of 93.3% well-represented strains. Furthermore, Proteobacteria (well) makes up for about 73% of the combined well-represented classes. Generally speaking, BLAST frequently confused under-represented fragments with well-represented phyla.

4.1.2 Genus

There were a total of 64 strains considered in the genus experiments. Each cross-validation repetition consisted of 13 (20%) strains chosen at random without replacement to BLAST against the remaining 51 (80%) strains in the dataset (see Table 1). Accordingly, every strain in the dataset was used as a test strain once. Each test strain was sampled randomly 100 times; each sample consisted of a fragment 500bp in length. The genus experiment shows that well-represented strains scored marginally higher than under-represented strains.

BLAST misclassified 25% of *Synechococcus CC9311* (under) 500bp fragments with strains belonging to the other three genera. 76% of these misclassifications went to a well-represented genus. When BLAST misclassified a fragment 21% of the misclassifications belonged to strains in the under-represented classes. Of the remaining 79% misclassified fragments, 48% went to strains belonging to the well-represented to Staphylococcus (well) alone. Generally speaking, BLAST frequently confused under-represented fragments with well-represented genus.

4.2 Equal Representation Experiments

The result of the phyla cross validation experiment with equal class representation is presented in Table 3. A dataset consisting of 60 strains was constructed to have equal representation among the four phyla classes. 15 strains were randomly sampled from each class from our original phyla dataset of 463 strains. Each cross-validation repetition consisted of 12 (20%) strains chosen at random without replacement to BLAST against the remaining 48 (80%) strains in the dataset (see Appendix A.2). Accordingly, every strain in the dataset was used as a test strain once. Each test strain was sampled randomly 100 times; each sample consisted of a fragment 500bp in length. These 100 fragments were used in place of the test strain.

Table 3. The result of the equal-representation cross validation experiment is provided below. The first score column is the percent accuracy of BLAST for all four classes while the four columns to the right, labeled with the phyla's abbreviated name, refer to the individual scores for each class considered in this experiment. These are the same four phyla considered in the well/under represented experiment except now we have selected 15 strains from each class for this experiment so that each repetition will have equal-representation among the four classes in the database. The cross validation procedure used here is identical to the one used for the well/under experiments.

Percentages		All	Prot	Firm	Cren	Ten
500bp Phyla	AVG	63.75	73.33	61.53	53.46	66.67
	STD	8.23	13.88	10.35	10.78	13.44

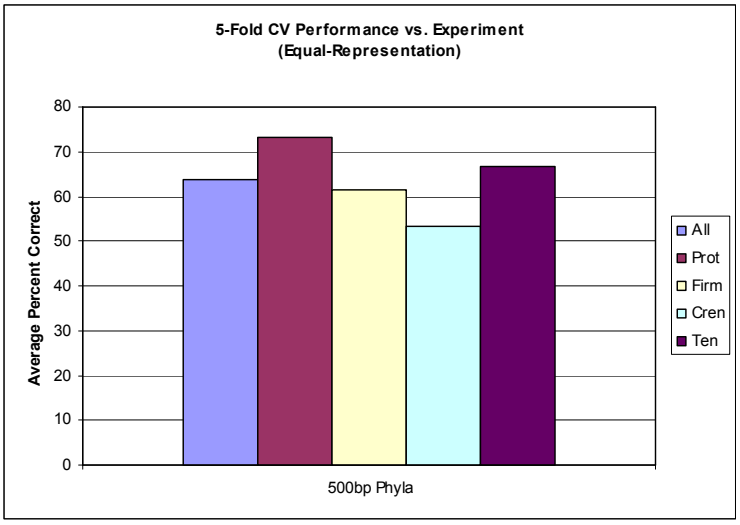


Figure 2. This bar graph illustrates the data provided in Table 3. The four classes in the equal-representation experiment have percent accuracy performance much more similar to one another than in the well/under representation experiment. This finding indicates that class composition in the database affects the performance of BLAST with well-represented classes having higher accuracy than under-represented classes.

The average percent accuracy for the experiment is shown in Figure 2. The overall percent accuracy decreased by ~25% from the well/under represented phyla experiment while the standard deviation increased from ~2.5% to ~8%. The percent accuracy decrease is mostly contributed to the decrease in accuracy (~29%) of Proteobacteria. The increase in standard deviation is also most significantly affected by Proteobacteria whose standard deviation increased by ~14%. This increase was expected since Proteobacteria was considered a well-represented class in the previous experiments and we've found that variance decreases with increasing representation.

5. Discussion

It is clear from our experiments that the accuracy of BLAST is highly dependent on the composition of the training database. The well/under phyla experiment confirmed that the well-represented classes have nearly 40% higher accuracy than the under-represented classes. Still BLAST is performing much better than chance on all classes. For phyla (500bp) we see that Proteobacteria (well) scored 92.67%. With a database composition of 252/370 we confirm that this score is much higher than chance which would be about 68%. This can also be verified for under-represented classes. For instance BLAST scored 59.38% for Tenericutes (under). Given its database composition we would expect a percent accuracy of 14/370 or 3.7% by chance.

Upon further examination of the database's composition we observe the ratio of well to under-represented strains in the phyla database is nearly 14:1 (Figure 3). Incidentally, by chance, if we rolled a die we would expect BLAST to classify a strain to a well-represented class 14/15 or 93.3% percent of the time. While we found through our experiments that BLAST is able to classify much better than chance, the allocation of BLAST misclassifications follows a different trend. For example, in the phyla experiments when BLAST misclassified a fragment ~ 95% (nearly chance) of the fragments were assigned to a well-represented class.

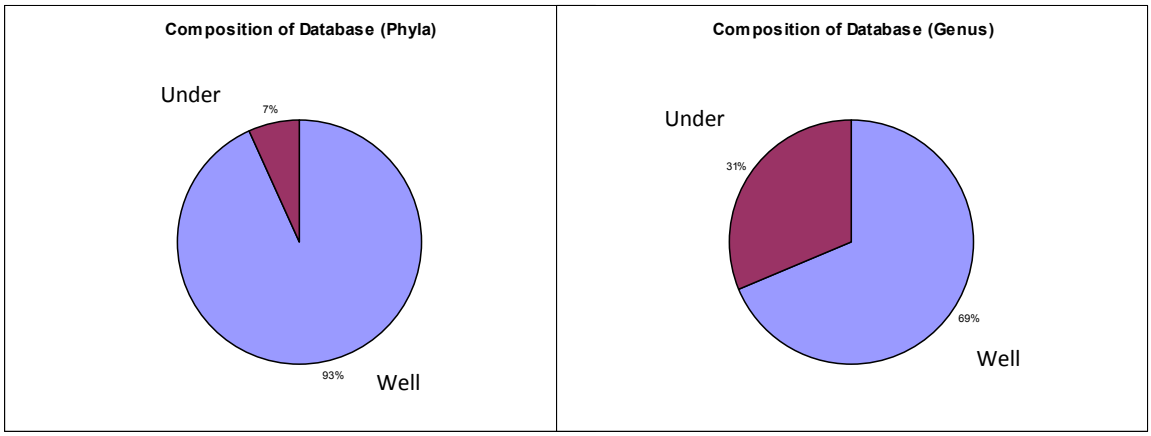


Figure 3. The pie charts above show the class composition for both the phyla and genus well/under experiments. The well-represented classes in the phyla experiment had nearly 40% higher percent accuracy scores than the under-represented classes. The difference in percent accuracy scores between the classes in the genus experiment was marginal.

These trends are also reflected in the genus experiments. For example for genus (500bp) we find that Streptococcus (well) scored 96.6%. By chance we would observe 21/51 or 41.1% accuracy. For Yersinia (under) BLAST scored 91.4% while a score by chance would be 8/51 or 15.6%. As shown in Figure 3 the genus database composition is about 2.2:1 predicting that BLAST would classify a strain to a well-represented class about 69% of the time by chance. This is reflected in the allocation of BLAST misclassifications where about 76% of the BLAST misclassified fragments went to a well-represented class.

The chart in Figure 1 indicates that the genus experiment outperformed the phyla experiment. We hypothesized that this was due to the difference in the well/under database composition (Figure 3) between the sets of experiments. Since the phyla under-represented composition is only 7% of the entire database as opposed to 31% for

the genus experiments, we wanted to find out if phyla overall percent accuracy score would approach the scores for the genus experiments if we increased the percent composition of phyla's under-represented classes.

We conducted the equal-representation experiment among the four classes at the phyla level to further test for compositional bias. The results show that the scores for the two under-represented classes increased while the scores for the two well-represented classes decreased substantially resulting in similar performance for all 4 phyla classes. Therefore we find that class composition size affects performance; improvement for the well-represented classes and degraded accuracy for under-represented classes. Proteobacteria's (well1) percent accuracy decreased 29% while Firmicutes (well2) decreased 22%. We infer that strains belonging to the genus level have less diversity than at the phyla level since there was such a substantial decrease in scores at the phyla level when the dataset was reduced to the size of the under-represented genus classes. There is proof to show that 16s rRNA sequences have 6% divergence at the genus level and 3% for species so we infer that this percentage is higher at the phyla level [18].

The standard deviation for classes in the phyla well/under represented experiment is shown in Figure 4. It is clear that class size has a significant effect on the standard deviation of percent accuracy scores. Proteobacteria with 252 strains had a standard deviation of 0.79% while Crenarchaeota with 12 strains had a standard deviation of 16.43%. The overall standard deviation for phyla increased from 2.29% to 8.23% moving from the well/under represented to the equally represented experiment. Database composition remains an important consideration in BLAST experiments in addition to the certainly of class labels.

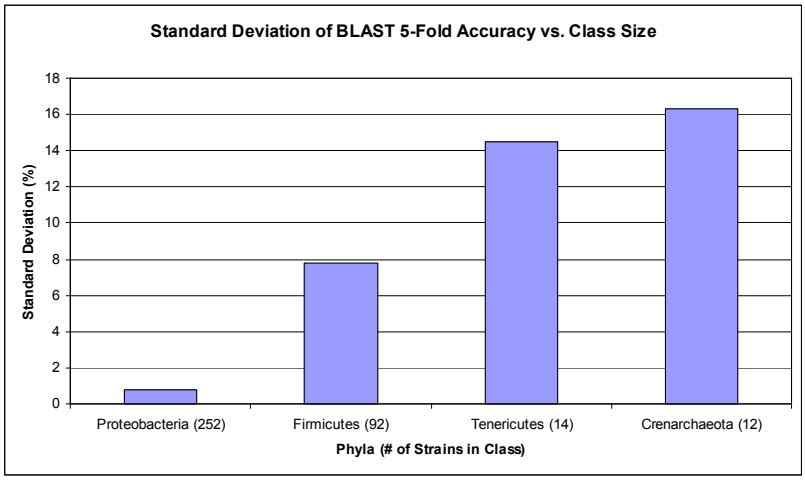


Figure 4. The bar graph above shows the standard deviations of BLAST's percent accuracy for each class over all five repetitions in the phyla well/under represented experiment. This graph clearly shows that for phyla the standard deviation of percent accuracy scores reported from BLAST decreases with increasing examples in the database.

6. Conclusion

Several five-fold cross validation experiments were examined in this study. Our analysis has shown that database composition contributes substantially to the accuracy of the BLAST algorithm. Overall we've found that the standard deviation of percent accuracy scores decreases with increasing class representation in the database. We've also demonstrated that BLAST performs much better than chance even when representation in the database is much smaller than the majority as shown in the phyla experiments. However, when BLAST misclassifies a fragment, it appears to assign the fragment by chance.

As shown in Figure 1 the genus experiment scored higher than the phyla experiment. We've shown through our equal-representation phyla experiment that this marked difference is most likely attributed to the genus level having better definition and less diversity than the phyla level. The study demonstrates the intuitive result that a user of BLAST would want to build a database having as much representation as possible to increase accuracy and decrease the standard deviation of scores. As the numbers from the experiments show, the output of BLAST is clearly not always the correct match and we suggest the use of additional, external information to form a consensus of matches.

Acknowledgments

The work in this paper was supported by the National Science Foundation CAREER award #0845827.

Appendix

A.1 CV Results - Well/Under Representations

Table 4. Results of the well/under cross validation experiment at the genus level, 500bp fragments

Genus Results – 500bp			
Overall Dataset			
Repetition	Training	Test	% Correct
1	51	13	97.15
2	51	13	93.23
3	51	13	98.62
4	52	12	94.67
5	51	13	95.69
		AVG	**95.87**
		STD	**2.10**

Genus Well-Represented				**Genus Under-Represented**			
Combined				**Combined**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	35	9	99.00	1	16	4	93.00
2	35	9	93.44	2	16	4	92.75
3	35	9	99.11	3	16	4	97.50
4	36	8	93.00	4	16	4	98.00
5	35	9	98.44	5	16	4	89.50
		AVG	**96.60**			**AVG**	**94.15**
		STD	**3.10**			**STD**	**3.57**
Streptococcus				**Yersinia**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	21	5	100.00	1	8	2	100.00
2	21	5	88.20	2	8	2	89.00
3	21	5	99.00	3	8	2	97.50
4	21	5	93.20	4	8	2	98.00
5	20	6	97.83	5	8	2	100.00
		AVG	**95.65**			**AVG**	**96.90**
		STD	**4.91**			**STD**	**4.56**
Staphylococcus				**Synechococcus**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	14	4	97.75	1	8	2	86.00
2	14	4	100.00	2	8	2	96.50
3	14	4	99.25	3	8	2	97.50
4	15	3	92.67	4	8	2	98.00
5	15	3	99.67	5	8	2	79.00
		AVG	**97.87**			**AVG**	**91.40**
		STD	**3.03**			**STD**	**8.51**

Table 5. Results of the well/under cross validation experiment at the phyla level, 500bp fragments

Phyla Results – 500bp			
Overall Dataset			
Repetition	Training	Test	% Correct
1	370	93	88.51
2	370	93	89.94
3	371	92	84.05
4	371	92	87.62
5	370	93	85.94
		AVG	**87.21**
		STD	**2.29**

Phyla Well-Represented				**Phyla Under-Represented**			
Combined				**Combined**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	345	86	90.98	1	25	7	58.14
2	345	86	93.47	2	25	7	46.57
3	345	86	87.50	3	26	6	34.67
4	345	86	89.72	4	26	6	57.50
5	344	87	88.63	5	26	6	46.83
		AVG	**90.06**			**AVG**	**48.74**
		STD	**2.30**			**STD**	**9.64**
Proteobacteria				**Crenarchaeota**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	252	63	91.94	1	12	3	50.67
2	252	63	93.29	2	12	3	28.00
3	252	63	91.71	3	12	3	14.67
4	252	63	93.44	4	12	3	36.00
5	252	63	92.97	5	12	3	54.67
		AVG	**92.67**			**AVG**	**36.80**
		STD	**0.79**			**STD**	**16.43**
Firmicutes				**Tenericutes**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	93	23	88.35	1	13	4	63.75
2	93	23	93.96	2	13	4	60.50
3	93	23	75.96	3	14	3	54.67
4	93	23	79.52	4	14	3	79.00
5	92	24	77.25	5	14	3	39.00
		AVG	**83.01**			**AVG**	**59.38**
		STD	**7.80**			**STD**	**14.52**

A.2 CV Results - Equal Representation

Table 6. Results of the equal-representation cross validation experiment at the phyla level, 500bp fragments

Phyla Results – 500 bp			
Overall Dataset			
Repetition	Training	Test	% Correct
1	48	12	61.50
2	48	12	73.50
3	48	12	69.42
4	48	12	52.08
5	48	12	62.25
		AVG	**63.75**
		STD	**8.23**

Proteobacteria				**Crenarchaeota**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	12	3	72.00	1	12	3	45.33
2	12	3	91.33	2	12	3	71.00
3	12	3	61.67	3	12	3	56.33
4	12	3	58.67	4	12	3	45.33
5	12	3	83.00	5	12	3	49.33
		AVG	**73.33**			**AVG**	**53.46**
		STD	**13.88**			**STD**	**10.78**
Firmicutes				**Tenericutes**			
Repetition	# Training	# Test	% Correct	Repetition	# Training	# Test	% Correct
1	12	3	57.00	1	12	3	71.67
2	12	3	56.33	2	12	3	75.33
3	12	3	80.00	3	12	3	79.67
4	12	3	58.00	4	12	3	46.33
5	12	3	56.33	5	12	3	60.33
		AVG	**61.53**			**AVG**	**66.67**
		STD	**10.35**			**STD**	**13.44**

References

1. V. Kunin, A. Copeland, A. Lapidus, K. Mavromatis and P. Hugenholtz, *Micro Mol Biol Rev.* **72**, 4 (2008).
2. G. Rosen, B. Sokhansanj, R. Polikar, M. Bruns, J. Russell, E. Garbarine, S. Essinger, and N. Yok, *Current Genomics*. **10**, 7 (2009).
3. J. Handelsman, Committee on Metagenomics: Challenges and Functional Applications, N. R. Council, Ed. *The National Academies Press*, (2007).
4. S. Altschul, W. Gish, W. Miller, E. Myers, and D. Lipman, *J. Mol Biol*. **215**, 3 (1990).
5. T. Madden, *The NCBI Handbook.* Ch. 16, 1-17 (2003).
6. J. Venter, K. Remington, J. Heidelberg, A. Halpern, D. Rusch, J. Eisen, D. Wu, I. Paulsen, K. Nelson, W. Nelson, D. Fouts, S. Levy, A. Knap, M. Lomas, K. Nealson, O. White, J. Peterson, J. Hoffman, R. Parsons, H. Tillson, C. Pfannkoch, Y. Rogers, and H. Smith, *Science*. **304**, 5667 (2004).
7. M. Tress, D. Cozzetto, A. Tramontano, and A. Valencia, *BMC Bioinformatics*. **7**, 213 (2006).
8. L. Koski and G.Golding, *J. Mol Evol.* **52**, 6 (2001).
9. A. Andersson, M. Lindberg, H. Jakobsson, F. Backhed, P. Nyren, and L. Engstrand, *PLoS One.* **3**, 7 (2008).

10. D. Huson, A. Auch, J. Qi, and S. C. Schuster, *Genome Res*. **17**, 3 (2007).
11. S. Havre, B. Webb-Roberston, A. Shah, C. Posse, B. Gopalan, and F. Brockman, *Proceedings of the 2005 IEEE Computational Systems Bioinformatics Conference*, 341–350 (2005).
12. K. E. Wommack, J. Bhavsar, and J. Ravel, *Appl Environ Microbiol.* **74**, 5 (2008).
13. C. Manichanh, C. Chapple, L. Franguel, K. Gloux, R. Guigo and J. Dore, *Nucleic Acids Res*. **36**, 16 (2008).
14. L. Krause, N. Diaz, A. Goesmann, S. Kelley, T. Nattkemper, F. Rohwer, R. Edwards, and J. Stoye, *Nucleic Acids Res*. **36**, 7 (2008).
15. G. L. Rosen, E. M. Garbarine, D. A. Caseiro, R. Polikar, and B. A. Sokhansanj, *Hindawi Adv Bioinfo*. **2008**, (2008).
16. R. Kohavi, *Proceedings of Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*, 1137–1143 (1995).
17. P. Devijver and J. Kittler. Pattern Recognition: A Statistical Approach, *Prentice-Hall*, London. (1982).
18. J. Clarridge, *Clin Microbiol Rev*. **17**, 4 (2004).

GUIDED GENOME HALVING: PROVABLY OPTIMAL SOLUTIONS PROVIDE GOOD INSIGHTS INTO THE PREDUPLICATION ANCESTRAL GENOME OF *SACCHAROMYCES CEREVISIAE*

HARIS GAVRANOVIĆ

Faculty of Natural Sciences, University of Sarajevo

ERIC TANNIER

INRIA Rhône-Alpes ; Université de Lyon ; UMR CNRS 5558, Laboratoire de Biométrie et Biologie Evolutive ; Université de Lyon 1 ; F-69622, Villeurbanne

We present theoretical and practical advances on the Guided Genome Halving problem, a combinatorial optimisation problem which aims at proposing ancestral configurations of extant genomes when one of them has undergone a whole genome duplication. We provide a lower bound on the optimal solution, devise a heuristic algorithm based on subgraph identification, and apply it to yeast gene order data. On some instances, the computation of the bound yields a proof that the obtained solutions are optimal. We analyse a set of optimal solutions, compare them with a manually curated standard ancestor, showing that on yeast data, results coming from different methodologies are largely convergent: the optimal solutions are distant of at most one rearrangement from the reference.

1. Motivations

The genomes of extant species underwent different changes in the course of evolution. These changes are studied at various levels: from modifications at a nucleotide-base level to rearrangements of large pieces of DNA, or duplication of the whole genome.

Genome rearrangement phylogeny methods have matured over the past decade[7] and reached the point where their results can be interpreted together with the results of biologists.

At a low level of resolution, the usually considered mutations that alter genomes are reversals, translocations, fusions and fissions of chromosomes, often included in a general abstract operation called Double Cut-and-Join (DCJ), which has the advantage of being computationally easy to handle and of modelling many realistic rearrangements.[16]

Following El-Mabrouk and Sankoff,[5] Zheng *et al*[18] have generalized the genome rearrangement problems by introducing the possibility of considering a whole genome duplication in the genome histories. They provide a heuristic algorithm[19] which is able to propose the organization of a set of genes along the history of yeast genomes.

In the same time, Gordon *et al*[8] chose not to use an automatic method to reconstruct the preduplication ancestor of *Saccharomyces cerevisiae*. Their arguments, among others, were that (i) only a small subset of the genes (less than 20%) can be taken into account, those which have retained two copies after the WGD, and (ii) the optimization problems are computationally complex and the methods are still in development.

In this paper, we report some advances on the theoretical study of the guided genome halving problem, and apply a heuristic algorithm on "double conserved syntenies" of different yeast species. As shown in a previous study,[12] the computation of double syntenies allows to apply the guided halving problem on instances that cover a good ratio of the exant genes (over 95%), and the computation of a lower bound proves that on some instances, the algorithm reaches an optimal solution. It is a solution to the two mentioned objections of Gordon *et al*,[8] and the comparison of the obtained solutions with the manually constructed one shows a good convergence. There were other objections in the paper of Gordon *et al*, as well as in a comment of Sankoff[9] in the same journal, as the possible huge number of optimal solutions, or the inability of the models to account for certain types of rearrangements, like telomeric translocations. This still calls for a theoretical answer (though telomeric translocations are actually taken into account by the DCJ framework) whereas the surprising convergence between all the solutions we find on one instance and the manually reconstructed ancestor shows that some dataset are already accessible.

In the next section, we present the mathematical definitions of the genomes and the rearrangements that may alter them. Then in Section 3, we propose a lower bound on the guided halving solutions, based on the so-called "double distance" computation. This bound may be reached, so is able to prove optimality of some solutions. Its usage in a branch and bound algorithm calls for efficient ways to compute it, since the double distance computation is NP-hard for the DCJ distance, and we are only able to compute its exact value for the easiest instances here. In Section 4, we describe the principle of our heuristic algorithm, and compare its efficiency with the state of the art one of Zheng *et al*[19] on some common instances build from yeast genomes, but with a low coverage of the whole genomes. Finally in Section 5, we apply this algorithm on good coverage syntenies on yeast genomes, and provide some information on the history of *Hemiascomycetes*, which we can compare to the ones Gordon *et al*. They are very similar, and give the hope that algorithmic rearrangement studies can provide good insights into genome evolution.

2. Genomes and rearrangements

2.1. *Genes and genomes*

We use the standard algorithmic definitions of genes and genomes[3,13]*. A *gene* A is an oriented sequence of DNA nucleotides, identified by its *tail* A^t and its *head* A^h. Tails and heads are called *extremities* of the gene. An *adjacency* is an unoriented pair of gene extremities. A *genome* Π is a set of adjacencies on a set of genes, such that every gene extremity participates in at most one adjacency. In a genome, an adjacency means that two genes are consecutive on the DNA molecule. In a genome Π, a gene extremity which is not adjacent to another gene extremity is called a *telomere*. A telomere a is also written as an adjacency $a\circ$, where $\circ$ is an abstract symbol not related to a gene, and called a *telomeric adjacency*.

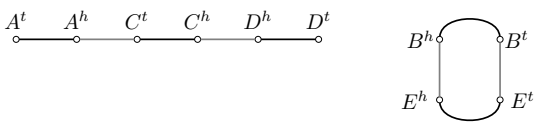
For given genome Π, the *genome graph* G_Π has vertex set the set of all gene extremities, and edge set the union of non telomeric adjacencies and edges A^tA^h, called *gene edges*, for every gene A. This graph has vertices of degree one or two. Thus, connected components are paths and cycles, and are called the *chromosomes* of Π. Paths are *linear* chromosomes, whereas cycles are *circular* chromosomes. Telomeres are degree one vertices of the genome graph. A genome with only linear, or only circular, chromosomes is called linear or circular genome, respectively.

A genome can also be represented as a set of strings, by writing the genes for each chromosome in the order in which they appear in the paths and cycles with a bar over the gene if the head of the gene appears before the tail and none if the tail appears before the head (this depends on an arbitrary direction of reading). For each linear chromosome, there are two possible equivalent strings, for two opposite traverses of the path. We have also two oposite circular strings for circular chromosomes.

Example 2.1. Let

$$\Pi = \{A^hC^t, C^hD^h, B^hE^h, E^tB^t\}$$

be a genome with five genes $\{A, B, C, D, E\}$. The corresponding genome graph is the following:



and the string representation consists in the linear string $AC\bar{D}$ (or $D\bar{C}\bar{A}$) and the circular string $B\bar{E}$ (or $E\bar{B}$).

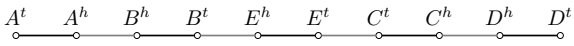
*They do not meet the biological definition of genes, but are more precisely "families of homologous sequences", containing from 0 to several genes in each genome.

A *Double Cut-and-Join* (DCJ) operation (sometimes called 2-break rearrangement) is defined for two adjacencies pq and rs of a genome. Telomeric adjacencies are also considered and any p, q, r or s could be a $\circ$ symbol. Moreover even the adjacency $\circ\circ$ is considered. Thus, the DCJ transforms two adjacencies pq and rs into either pr and qs, or ps and qr. The DCJ was introduced by Yancopoulos *et al*[16] to encompass all interesting types of genomic rearrangements. Indeed, an inversion, a chromosome fusion or fission, as well as a reciprocal translocation, is a particular case of a DCJ. A non-reciprocal translocation, meaning a chromosome arm is translocated to another chromosome, is also a DCJ. Transpositions and block interchanges can be mimicked by two consecutive DCJs, the intermediate genome containing a circular chromosome.

Example 2.2. Consider genome from Example 2.1 and the DCJ that transforms adjacencies A^hC^t, B^hE^t into A^hB^h, C^tE^t. The new genome is then

$$\Pi' = \{A^hB^h, E^hB^t, C^tE^t, C^hD^h\}$$

and its genome graph becomes



Given a genome Π, it is always possible to transform it into another arbitrary genome on the same set of genes applying a sequence of DCJ operations. This leads to the definition of DCJ distance.

The DCJ Sorting and Distance Problem. Given two genomes Π and Ψ defined on the same set of genes, find a shortest sequence of DCJ operations that transforms Π into Ψ. The length of such a sequence is called the *DCJ distance* between Π and Ψ, denoted by $d_{DCJ}(\Pi, \Psi)$.

As the DCJ distance is the main distance we consider here, d_{DCJ} is often abbreviated by d. When three genomes are considered, this yields the Median problem:

The Genome Median Problem. Given three genomes Π_1, Π_2, Π_3, find a *median* genome Π wich minimises $d_{DCJ}(\Pi_1, \Pi)+d_{DCJ}(\Pi_2, \Pi)+d_{DCJ}(\Pi_3, \Pi)$.

The *breakpoint graph* of a set of genomes on the same set of genes is the graph on vertex set the extremities of all genes, and edge set the set of adjacencies of all genomes. If an edge is an adjacency on a genome Π, it is called a Π-edge.

Each vertex of the breakpoint graph has degree at most the number of considered genomes. So for two genomes, it is a set of paths and cycles. The DCJ distance is easily computed from the breakpoint graph: for two genomes Π and Ψ on n genes, if C is the number of cycles and P the number of paths with an even number of edges (including trivial path with 0 edges),

$$d(\Pi, \Psi) = n - (C + P/2).$$

The DCJ sorting and distance problems thus have a linear time running solution.[3] The genome median problem, however, is NP-hard[13] under the DCJ distance, thought there are good algorithms to solve it.[11,15]

2.2. *Duplicated genes and genomes*

A *duplicated gene* is a couple of homologous oriented sequences of DNA nucleotides, identified by two tails A_1^t and A_2^t, and two heads A_1^h and A_2^h. An *all-duplicates* genome Δ is a set of adjacencies on a set of duplicated genes, where each gene extremity is contained in at most one adjacency.

For a genome Π on a gene set, a *doubled genome* $\Pi\oplus\Pi$ is an all-duplicates genome on the set of duplicated genes from the same gene set such that if A^xB^y $(x, y \in \{t, h\})$ is an (possibly telomeric) adjacency of Π (A^x or B^y may be $\circ$), either $A_1^xB_1^y$ and $A_2^xB_2^y$, or $A_1^xB_2^y$ and $A_2^xB_1^y$, are adjacencies in $\Pi \oplus \Pi$.

We note that on a doubled genome one finds two identical copies of each chromosome when we ignore the $1's$ and $2's$ in the names of genes. More precisely, it has two copies of each linear chromosome, and for each

circular chromosome, either two circular copies or one circular chromosome containing the two successive copies. For one genome Π there is an exponential number of possible doubled genome $\Pi \oplus \Pi$.

The DCJ distance and median problems generalize to the case of duplicated genomes in several ways.

The DCJ distance between two all-duplicates genomes is easily derived from the usual DCJ distance and has a polynomial time computation. But if we ignore the $1's$ and $2's$ in the names of genes, then then it calls for an re-assignement of $1's$ and $2's$ which minimizes the distance, and this problem is NP-complete.

The double distance problem. The *double distance* between an ordinary genome Π and an all-duplicates genome Δ is defined as $d(\Pi, \Delta) = min_{\Pi \oplus \Pi} d(\Pi \oplus \Pi, \Delta)$. It is also NP-hard.[13]

The Genome Halving Problem. Given an all-duplicates genome Δ on a set of duplicated genes, find a doubled genome $\Pi \oplus \Pi$ on the same set of genes such that the DCJ distance between Δ and $\Pi \oplus \Pi$ is minimal. The DCJ distance $\min_{\Pi \oplus \Pi} d(\Pi \oplus \Pi, \Delta)$ (the genome halving score) is denoted by $gh(\Delta)$.

The genome halving problem aims at constructing possible preduplication configuration of genomes which have undergone a whole duplication in the course of their histories. Computationnally, it is a generalization of the DCJ sorting problem (indeed if the two copies of the ancestral doubled genome evolve independently and no rearrangement concerns both, then it is equivalent to simple DCJ sorting). It was solved in the most complicated case where only linear chromosomes are allowed by El-Mabrouk and Sankoff,[5] resulting in a rather complicated algorithm. Alekseyev and Pevzner discuss and solved the same problem on unichromosomal genomes.[1] Recently, solutions with DCJ distance were presented by Warren and Sankoff[14] and Mixtacki.[6]

The solution relies on the *contracted breakpoint graph*[†] of the genome Δ: its vertex set is the set of pairs of homologous extremities of duplicates genes (two extremities (A_1^h, A_2^h) or (A_1^t, A_2^t) form a single vertex). Two vertices are connected by an edge if two extremities are adjacent in Δ. This graph is a set of cycles and paths. If we call n the number of genes (counting one for the two duplicates), EC the number of cycles of even length and EP the number of paths with an even number of edges (including trivial paths with no edges), then Mixtacki[6] proved that

$$gh(\Delta) = n - (EC + \lfloor \frac{EP}{2} \rfloor).$$

This formula yields a linear algorithm to solve the Genome Halving problem. The analysis of this algorithm shows the existence of a large, often exponential, number of optimal solutions. This fact makes it inappropriate for any practical, biologically significant computation. Seoighe and Wolfe[10] noted this extreme non-uniqueness of solution to genome halving problem and propose to use a reference genome, *i.e.* outgroup to reduce this number. Zheng *et al*[18] formalize this approach and propose the first computational method to solve it following with more recent[19] and more efficient method applied to find phylogenetic relationships among yeasts of the *Saccharomyces* complex. Here they also define the Genome Halving problem with two outgroups.

The Guided Genome Halving Problem. Given an all-duplicates genome Δ and an ordinary genome Γ defined on the same set of genes, find an ordinary genome Π which minimizes

$$ggh(\Delta, \Gamma) = d_{DCJ}(\Delta, \Pi) + d_{DCJ}(\Pi, \Gamma).$$

These problems have different variants depending of the possibility for a genome of having one or several chromosomes, only linear chromosomes or only circular chromosomes or a mix between the two. In what follows we consider the genomes with several, exclusively linear chromosomes and the distance is DCJ distance as defined here. The problem is a generalization of the median problem and is NP-hard as shown by Tannier et al. and by Zheng et al.[13,19]

[†]It is a generalization to linear chromosomes of the "contracted breakpoint graph" defined by Alexseyev and Pevzner,[1] and is the line-graph of the "natural graph" used by Mixtacki.[6]

3. A lower bound

For the median problem, there is a usual folklore lower bound that is used for very efficient branch and bound approaches.[15] Indeed, for three genomes Π_1, Π_2 and Π_3, and a median genome Π, it is trivial from the triangle inequality that

$$d_{DCJ}(\Pi_1,\Pi)+d_{DCJ}(\Pi_2,\Pi)+d_{DCJ}(\Pi_3,\Pi)$$
$$\geq \frac{d_{DCJ}(\Pi_1,\Pi_2)+d_{DCJ}(\Pi_1,\Pi_3)+d_{DCJ}(\Pi_3,\Pi_2)}{2}.$$

The Guided Genome Halving problem is a generalization of the median problem (indeed, if from the ancestor the two copies of the doubled genome evolve independenly and no rearrangement mixes the two, then it is equivalent to the median problem). But no such bound exists for this problem. We draw an equivalent, though less trivial and less computationally easy, which helps evaluating solutions.

Theorem 3.1. *Given an all-duplicates genome Δ and an ordinary genome Γ defined on the same set of genes,*

$$ggh(\Delta,\Gamma) \geq \frac{d_{DCJ}(\Gamma,\Delta)+gh(\Delta)}{2}.$$

Indeed, it is an easy exercise to show that the double distance verifies the triangle inequality. This yields, for any genome Π and doubled genome $\Pi\oplus\Pi$,

$$d(\Gamma\oplus\Gamma,\Delta) \leq d(\Gamma\oplus\Gamma,\Pi\oplus\Pi)+d(\Pi\oplus\Pi,\Delta).$$

Clearly, $\Gamma\oplus\Gamma$ can be chosen so that $d(\Gamma\oplus\Gamma,\Pi\oplus\Pi)\leq 2d(\Gamma,\Pi)$, which gives

$$d(\Gamma\oplus\Gamma,\Delta) \leq 2d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta).$$

Adding $d(\Pi\oplus\Pi,\Delta)$ to both sides, we get

$$d(\Gamma\oplus\Gamma,\Delta)+d(\Pi\oplus\Pi,\Delta) \leq 2(d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta)).$$

And finally, as by definition $gh(\Delta)\leq d(\Pi\oplus\Pi,\Delta)$ and $d(\Gamma,\Delta)\leq d(\Gamma\oplus\Gamma,\Delta)$,

$$\frac{d(\Gamma,\Delta)+gh(\Delta)}{2} \leq d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta).$$

Which, for a genome Π such that $ggh(\Gamma,\Delta)=d(\Gamma,\Pi)+d(\Pi\oplus\Pi,\Delta)$, yields the result

The bound may be reached only if the optimal solution of the guided halving problem is also an optimal solution to the halving problem for the all-duplicates genome.

It is based on the computation of the double distance, which is NP-complete.[13] So it is not immediately usable in a branch and bound algorithm as the one for the median problem.[15] A less tight bound may be used by replacing $d_{DCJ}(\Gamma,\Delta)$ by $d_{BP}(\Gamma,\Delta)/2$, where d_{BP} is the double breakpoint distance and is computed with a linear algorithm (see Tannier *et al*[13]). This more tractable bound is less often reached and never allows to prove optimality in our case. For the easiest instances on yeast data (see Section 5), we could compute the DCJ bound exactly and it allows to prove optimality of our Guided Genome Halving solutions given by the algorithm below.

4. The Algorithm for Guided Genome Halving

4.1. *Contracted breakpoint graphs*

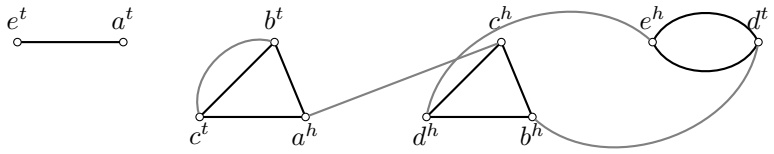
Previous work and experience on solving the problem computationally come from Sankoff's group.[18,19] The first approach[18] was to generate all possible genome halving solution and thereafter choose the subset of solutions that minimize the distance from the outgroup. At the end, the authors develop local improvement

heuristic searching in the neighbourhood of optimal halving solutions in order to find better solutions. Due to potentially huge number of halving solutions only numerical results for maize with 34 doubled blocks are reported. For any bigger instance one is obliged to choose heuristically a small subset of halving solutions and proceed with the method, therefore trade off the time and quality of solutions.

The inspiration for our algorithm is the idea from Zheng *et al*[19] where the authors combine information from both the all-duplicates genome and the outgroup early in the process of constructing the ancestor. We will use a small concrete genome with 5 duplicated genes to explain the main idea of the algorithm, and we will use the genome of Example 2.1 as an outgroup.

Definition 4.1. Let Δ be all-duplicates genome and let Γ be an ordinary genome defined on the same set of genes. The *contracted breakpoint graph* of Δ and Γ is the graph built on the contracted breakpoint graph of Δ (which edges are called the red edges), by adding an edge (called blue edge) for each adjacency of Γ. We note the obtained graph $B(\Delta, \Gamma)$.

Example 3: Let $\Delta = a_1 b_1 \bar{c_1} b_2 \bar{d_1} \bar{e_1} a_2 c_2 \bar{d_2} \bar{e_2}$ be an all-duplicated genome. Let $\Gamma = a\bar{c}bd\bar{e}$ be an ordinary linear unichromosomal genome. The associated contracted breakpoint graph follows (red edges are drawn bold, while blue ones are thin).

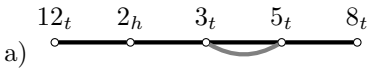


4.2. *The algorithm*

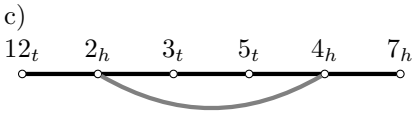
We used similar ideas as in Alekseyev and Pevzner[2] or Zhao and Bourque,[17] who find "reliable rearrangements" in a multiple breakpoint graph, as well as in Xu,[15] who searches for "adequate subgraphs" in a multiple breakpoint graph. The two are the dynamic and static versions of the same principle: indentifing some subgraphs in a breakpoint graph (whether it is a multiple or a contracted breakpoint graph changes only the details) for which there is a provably optimal local ancestral arrangement, and thus rearrangements. That is, for some patterns on the graph, it is possible to draw a realiable ancestor, and then to restrict the heuristic principles on the rest of the graph, which we expect much smaller.

A DCJ operation on genome Δ is immediately transposable on the contracted breakpoint graph: it consists in deleting two edges (or one edge and one telomere), and join the 4 pending vertices by two other edges. We chose the dynamic approach of Alekseyev and Pevzner[2] or Zhao and Bourque,[17] identifying reliable rearrangements: we start with the contracted breakpoint graph of Δ and Γ, and apply DCJ operations in sequence until all red edges are doubled, which means we reached a doubled genome, so a solution to the guided genome halving problem. We detect the following three configurations:

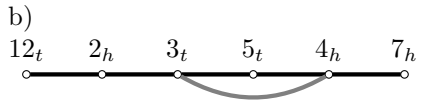
- We apply first all DCJs that directly lead to a red-blue cycle of length two;
- then we choose small sequences of DCJ leading to red-blue cycles of length two, at the condition that it is not destroying any other red-blue two cycle. We can recognize such a sequence in the contracted breakpoint graph: it is drawn from cycles consisting of some red edges and one blue edge, while other blue edges adjacent to cycle are adjacent to red connected components (see Figure 1). The sequence can be shuffled to diversify the final solution.
- finally choose sequences of DCJ leading to red-blue cycles of length four, at the condition that it is not destroying immediately any red-blue 2-cycle. We recognize such a sequence in the contracted breakpoint graph as a cycle consisting of some red edges and two blue edges while other blue edges adjacent to the cycle are adjacent to other red connected components of the contracted breakpoint graph (see Figure 1). The sequence can be shuffled to diversify the final solution.



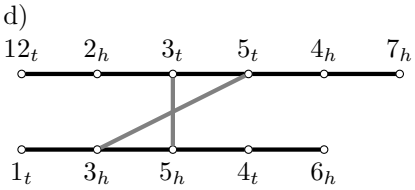
This is a straitforward case where a red edge is parallel to a blue edge. We choose the DCJ that transforms $(2_h, 3_t), (5_t, 8_t)$ into $(5_t, 3_t), (2_h, 8_t)$, constructing a double-red/blue cycle.



In this case, there is a red-blue cycle with adequated parity. A first DCJ connects 3_t and 5_t while the other connects 2_h and 4_h. Note that any order of these DCJs will construct one double-red/blue cycle.



Here there is a red-blue cycle but its parity does not allow a sequence of DCJs that construct a double-red/blue cycle.



This configuration forces the choice of two DCJs: one that connects 3_t and 5_t while the other connects 3_h and 5_h. These operations lead to a double-red/blue cycle of size 4 that is solved with only one DCJ.

Fig. 1. The detected DCJ rearrangements. Red edges are drawn black and blue edges are drawn grey.

When no such pattern is found, we apply series of DCJ which all lead to an optimal genome halving ancestor, with a randomized choice.

4.3. *Results on Sankoff's instances on yeast genomes*

Zheng *et al*[19] use their Guide Genome Halving Heuristic on several instances built from yeast genes (personnal communication). They choose a pool of genes from the Yeast Gene Order Browser,[4] namely those which are in one exemplar in the non duplicated species and in exactly two exemplars in the duplicated species. So a minority of the genes are covered, make it difficult to compare with the manual reconstructed ancestor of Gordon *et al.*[8] However they provide a good benchmark for comparison with our method, showing that we come to similar performances. We achive slighlty better solutions on the majority of instances. Results are presented on Table 1.

5. Results on high coverage yeast data

5.1. *Double conserved syntenies*

The instances of the Guided Genome Halving problem were constructed by a double conserved synteny method described in an earlier paper,[12] which roughly consists in, given orthologies between a duplicated genome and a non-duplicated genome (all orthologies are taken from the YGOB[4]) looking for a set S of gene families which have one gene in the first genome, and at least one in the second, and verify

(1) the genes of S are contiguous in Lk;
(2) the genes of S form two contiguous segments A_{Sc} and B_{Sc} in Sc, of at least 2 genes each;
(3) the two sets A_{Lk} and B_{Lk} of genes of Lk which have homologs respectively in A_{Sc} and B_{Sc} form two intersecting segments in Lk;
(4) At least one extremity of A_{Sc} (resp. B_{Sc}) is homologous to an extremity of A_{Lk} (resp. B_{Lk});
(5) S is maximal for these properties.

The first two conditions impose the presence of one segment in Lk and two orthologous segments in Sc, with a minimum size. It is the basis of the double synteny signal. The presence of at least two genes

Instance Γ - Δ	$2n$	$d(\Delta, X * X)$	$d(X, \Gamma)$	$d(X, \Gamma)$
AG-CG	538	186	153	144
AG-SC	1012	119	188	187
KL-CG	546	186	147	147
KL-SC	1026	122	197	197
KW-CG	542	188	215	205
KW-SC	994	121	323	317
A*-CG	600	199	84	71
A*-SC	1062	124	5	8
AG-V	576	61	148	149
KL-V	584	62	157	155
KW-V	582	62	212	210
A*-V	600	62	29	27

Table 1. Comparison of the results obtained in Zheng *et al*[19] and by our algorithm. The first column indicated the couple of compared species. The number of duplicated genes (in these instances they are the synteny blocks) is reported in the second column while in the third on we find the genome halving distance. All the best solutions we found are also genome halving solutions. The forth and the fifth column contain distances from the solution the outgroup as reported by Zheng *et al* and obtained in this study. AG stands for *E. gossypii*, CG for *C. glabrata*, SC for *S. scerevisiae*, KL for *K. lactis*, KW for *K. waltii*, A* for the pre-duplication ancestor of SC and CG, and V for the post-duplication last common ancestor of SC and CG. These are the notations used by Zheng *et al.*

avoids the possible fortuitous presence of one transposed or misannotated gene. The third condition avoids the ambiguous signal of two successive single syntenies. The fourth condition is used to orient the markers.

In this way, we were able to compare every pair of duplicated yeast (2 assembled species) and non-duplicated yeast (5 assembled species). The coverage of the genomes is always above 95%, which allows to reconstruct a large part of the history of the genomes.

5.2. *The alternative ancestors*

One first surprising thing already remarked in the earlier study[12] is that the solutions to the Guided Genome Halving on two species (*Saccharomyces cerevisiae* and *Lachancea kluyverii*) and the manually constructed ancestor of Gordon *et al*[8] from 11 species come very close. It is confirmed here, by the examination of several solutions.

First of all, on this instance, our program (as well as the one of Zheng *et al*[19]) finds an optimal solution: indeed, the value of the solution is 140, while the bound gives 139.5. This instance is one of the few for which the bound is tractable without involving deep algorithmics. So it is a good information that no arrangement can be strictly more parsimonious than the ones we find.

Our solutions are sequences of DCJs on the all-duplicates genome. 26 different sequences lead to an optimal solutions, and among the 26 solutions, only two are different. They vary by one reciprocal translocation and are both distant of one telomeric translocation from the solution of Gordon *et al*,[8] which is suboptimal for the number of DCJ (score 141).

The only point in which all three results vary is the position of a small part of Gordon *et al*[8]'s ancestral chromosome 1 (Anc1.1-Anc1.120), which is alternatively fusioned to ancestral chromosome 2 or 6.

5.3. *The rearrangements*

Gordon *et al*[8] have also manually infered all the rearrangements from the ancestral genome to *Saccharomyces cerevisiae*. They found in total 144 rearrangements, 73 being inversions, 66 reciprocal translocations and 5 telomeric translocations.

We find 115 DCS between our ancestor and the genome of *Saccharomyces cerevisiae*, and 116 between Gordon *et al*'s ancestor. The difference is probably partly due to our definition of Double Conserved Syntenies, which allow local rearrangements to a certain extant. Rearrangement distances are always difficult to compare at different levels of resolution. But interchromosomal rearrangements are comparable, since rearrangements inside the makers should be only inversions.

There is no unique scenario as pointed by Gordon *et al*,[8] and some types of rearrangements are difficult to assess with certainty. But for a pool of 26 scenarios, we found around 20 inversions, 7 couples of DCJs transposing or exchanging the positions of blocks, 70-71 reciprocal translocations, and 7-8 telomeric translocations. The number of reciprocal translocations and telomeric translocations hashas the same order than in the manually reconstructed scenario. Inversions are much less numerous, even if transposition couples of DCJs are counted as 3 inversions. So most inversions seem so be included local rearrangements within the syntenies. No current method can faithfully evaluate this number of local rearrangements which involve double conserved syntenies.

Gordon *et al*[8] only make their analyses on the *Saccharomyces cerevisiae* branch. Here, we are able to reconstruct also the whole distance matrix between all yeast species available in YGOB.[4] The results, in terms of numbers of DCJs, are reported in Table 2. The tendecies in the rearrangement rates on all species follow the "number of blocks" statistic of Gordon *et al*.[8] A multiple study would identify the shared rearrangements, but this is left for a future work.

	E.gossypii	K.lactis	L.thtolerans	L.waltii*	L.kluyveri	Z.rouxii
S.cerevisiae	117 + 163	116 + 183	111 + 67	114 + 84	115 + 25	119 + 118
S.bayanus*	156 + 164	140 + 175	168 +109	166 + 120	176 + 83	173 + 144
C.glabrata	251 + 177	258 + 189	266 + 108	256 + 124	266 + 75	273 + 136
N.castellii*	177 + 169	166 + 189	192 + 106	192 + 118	199 + 81	195 + 146
V.polyspora*	199 + 173	192 + 182	220 + 101	216 + 117	225 + 78	215 + 146

Table 2. DCJ distances between pairs of genomes. $A+B$ means: A is the distance from the ancestor to the duplicated species, and B is the distance from the ancestor to the non-duplicated species. So on one line, we expect the first number to be approximatively the same (up to a variation on the number of local rearrangements inside the markers, which vary for each comparison). Species which name is followed by an asterisk are not yet assembled, so probably the number of rearrangements is overestimated.

6. Conclusion

We presented an algorithm for the Guided Genome Halving problem, as well as a lower bound. The algorithm uses ideas similar to the ones of Zheng *et al*,[19] accompagnied by principles used for the median problem by Xu,[15] or for multiple comparisons by Alekseyev and Pevzner[2] or Zhao and Bourque,[17] which is natural as the guided halving generalizes the median. On some instances coming from yeast order data, the bound gives a proof that the obtained solution is optimal. Comparing a set of optimal solutions with a standard preduplication ancestor of *Saccharomyces cerevisiae*, we obtain two interesting conclusions : the standard arrangement is sub-optimal, at one operation from optimal solutions, and the latter vary only by the position of a single block.

This shows that on yeast data, it seems that the Guided Halving problem provides a very good modelization, perhaps better than on mammalian data, where automatic methods have diverged from standard manual studies for a while.

Future work will concern the efficient computation of the double distance problem, in order to provide a bound which is possible to use within an algorithm for the Guided Genome Halving. Zheng *et al*[19] have also generalized the Guided Halving to instances with two non-duplicated genomes. The data and study from Gordon *et al*[8] also calls for the possibility of reconstructing the full *Saccharomycetes* phylogeny on several duplicated as well as non duplicated species.

Ackownledgements

We thank Chunfang Zheng for kindly providing her program and the instances on which we could make a benchmark comparison. Thanks to Ken Wolfe for the informations on the unique locus which seems to be displaced in the guided genome halving solutions. This work was supported by the Centre National de la Recherche Scientifique, and by the Agence Nationale de la Recherche (ANR-08-GENM-036-01).

References

1. Alekseyev MA, Pevzner PA (2008) Colored de Bruijn graphs and the genome halving problem, IEEE/ACM Transactions on Computational Biology and Bioinformatics 4:98-107
2. Alekseyev MA, Pevzner PA (2009) Breakpoint graphs and Ancestral Genome Reconstruction, Genome Research 19(5):943-57
3. Bergeron A. Mixtacki J, Stove J (2006) A unifying view of genome rearrangments, Proceedings of WABI'06, Springer, Lecture Notes in Computer Science 4175:163-173
4. Byrne KP, Wolfe KH (2005) The Yeast Gene Order Browser: combining curated homology and syntenic context reveals gene fate in polyploid species. Genome Research 15(10):1456-61
5. El-Mabrouk N, Sankoff D (2003) The reconstruction of doubled genomes, SIAM Journal of Computing 32:754-792
6. Mixtacki J (2008) Genome Halving under DCJ Revisited Proceedings of COCOON'08, Springer, Lecture Notes in Computer Science 5092:276-286
7. Fertin G, Labarre A, Rusu I, Tannier E, Vialette S (2009) Combinatorics of Genome Rearrangements, MIT press
8. Gordon JL, Byrne KP, Wolfe KH (2009) Additions, Losses, and Rearrangements on the Evolutionary Route from a Reconstructed Ancestor to the Modern Saccharomyces cerevisiae Genome, PLoS Genetics 5(5): e1000485
9. Sankoff D (2009) Reconstructing the History of Yeast Genomes, PLoS Genet 5(5): e1000483
10. Seoighe C, Wolfe KH (1998) Extent of genomic rearrangement after genome duplication in yeast, Proc Natl Acad Sci U S A. 95(8):4447-52
11. Swenson KM, Arndt W, Tang J, Moret BME (2008) Phylogenetic reconstruction from complete gene orders of whole genomes, Proceedings of APBC'08, Imperial College Press, Advances in Bioinformatics and Computational Biology 6:241-250
12. Tannier E (2009) Yeast ancestral genome recontruction: the possibilities of automatic methods, Proceedings of RECOMB Comparative Genomics'09, Springer, Lecture Notes in Bioinformatics 5817:1-12
13. Tannier E, Zheng C, Sankoff D (2009) Multichromosomal Median and Halving Problems under Different Genomic Distances, BMC Bioinformatics 10:120
14. Warren R, Sankoff D (2009) Genome aliquoting with double cut and join, BMC Bioinformatics, 10(Suppl 1):S2
15. Xu AW (2008) A Fast and Exact Algorithm for the Median of Three Problem–A Graph Decomposition Approach, Proceedings of Recomb-Comparative Genomics'08, Springer, Lecture Notes in Bioinformatics 5267:184-197
16. Yancopoulos S, Attie O, Friedberg R (2005) Efficient sorting of genomic permutations by translocation, inversion and block interchange, Bioinformatics 21:3340-3346
17. Zhao H and Bourque G (2009) Recovering genome rearrangements in the mammalian phylogeny, Genome Research 19:934-942
18. Zheng C, Zhu Q, Sankoff D (2006) Genome halving with an outgroup, Evolutionary Bioinformatics 2:319-326
19. Zheng C, Zhu Q, Adam Z, Sankoff D (2008) Guided genome halving: hardness, heuristics and the history of the Hemiascomycetes, Bioinformatics 24(13):i96-104

A PRACTICAL ALGORITHM FOR ESTIMATION OF THE MAXIMUM LIKELIHOOD ANCESTRAL RECONSTRUCTION ERROR

GLENN HICKEY AND MATHIEU BLANCHETTE

McGill Centre for Bioinformatics and School of Computer Science, McGill University, 3480 University St., Montréal, Québec, H3A 2B4, Canada.

The ancestral sequence reconstruction problem asks to predict the DNA or protein sequence of an ancestral species, given the sequences of extant species. Such reconstructions are fundamental to comparative genomics, as they provide information about extant genomes and the process of evolution that gave rise to them. Arguably the best method for ancestral reconstruction is maximum likelihood estimation. Many effective algorithms for accurately computing the most likely ancestral sequence have been proposed. We consider the less-studied problem of computing the expected reconstruction error of a maximum likelihood reconstruction, given the phylogenetic tree and model of evolution, but not the extant sequences. This situation can arise, for example, when deciding which genomes to sequence for a reconstruction project given a gene-tree phylogeny (The Taxon Selection Problem). In most applications, the reconstruction error is necessarily very small, making Monte Carlo simulations very inefficient for accurate estimation. We present the first practical algorithm for this problem and demonstrate how it can be used to quickly and accurately estimate the reconstruction accuracy. We then use our method as a kernel in a heuristic algorithm for the taxon selection problem. The implementation is available at http://www.mcb.mcgill.ca/ blanchem/mlerror

Keywords: Ancestral Reconstruction Error; Maximum Likelihood; Ancestral Genomes; Error estimation; Sequence Evolution

1. Introduction

The rapid increase in DNA sequencing throughput over the last few years has greatly increased the number of species whose genome is completely or partially sequenced. This represents an extraordinary opportunity for comparative genomics and genome evolution studies. The availability of a few dozen mammalian genomes paves the way for efforts toward the computational inference of ancestral genomes, based on those of extant species. Ancestral sequence reconstructions have been undertaken at several levels. Ancestral protein sequences have been inferred and their function tested[1–3] . Blanchette et al.[4] have reconstructed large genomic regions of ancestral mammals and Paten et al.[5] have proposed reconstructions for whole ancestral mammalian genomes. Ancestral sequence reconstructions are key to understanding how genomes evolve and how sequences adapt. They further provide useful information toward the functional annotation of extant genomes, through the identification of evolutionary signatures.[6–8]

Much of the work in the ancestral sequence reconstruction community has focused on designing algorithms to infer ancestral states as accurately as possible. The process starts with the multiple alignment of orthologous sequences[9–12] and a model of sequence evolution, from which a phylogenetic tree is derived. Given this data, one seeks assignments of sequences to ancestral nodes that produce the most parsimonious or most likely evolutionary scenario. If only substitutions are allowed and sites evolve independently from each other, ancestral states can be inferred separately for each site, using the Fitch algorithm for parsimony,[13] Sankoff's algorithm for weighted parsimony,[14] or Felsenstein's algorithm for maximum likelihood,[15] all of which run in time $O(n \cdot L)$, where n is the number of extant species and L is the number of sites. When insertions and deletions are considered, finding the maximum parsimony solution becomes NP-hard[16] because sites cannot be treated independently. However, good heuristics have recently been developed.[7,17] The presence of genome rearrangements and duplications complicate ancestral inference further.[18–21]

While good algorithms or heuristics exist for most versions of the ancestral sequence reconstruction problem, the problem of assessing the accuracy with which the sequence at given ancestral node can be reconstructed has received less attention. Specifically, we are interested in the following problem:

ANCESTRAL RECONSTRUCTION EXPECTED ACCURACY PROBLEM
Given:

- A phylogenetic tree $\mathcal{T}$
- A stochastic model of sequence evolution along each branch of $\mathcal{T}$
- An ancestral sequence reconstruction algorithm M

Find: The expected accuracy with which algorithm M can reconstruct the sequence at each ancestral node, where the expectation is taken over all possible realizations (set of sequences at the leaves of $\mathcal{T}$) of the evolutionary process. Note that no sequences are part of the input to the AREA problem. Instead, sequences are random variables generated by the stochastic model of evolution: a random sequence is generated at the root of tree $\mathcal{T}$ and evolves randomly, according to the stochastic model provided for each branch, until leaf sequences are obtained. Let X_i be the random sequence generated at node i. An ancestral sequence reconstruction algorithm M computes a deterministic function of the random sequences generated at the leaves, $X_{l_1}, X_{l_2}, ..., X_{l_n}$ to predict an ancestral sequence $\hat{X}_{root}$ at the root of $\mathcal{T}$. Our goal is to estimate the expected value of the difference $d(X_{root}, \hat{X}_{root})$ between the generated and predicted ancestral sequence, for some appropriate edit-distance measure $d(\cdot)$.

Perhaps the simplest approach to assess the expected reconstruction accuracy of a given algorithm on a given tree is through simulations: Repeatedly generate sequences on T, apply the algorithm to the set of sequences generated at the leaves, and measure the distance between the generated and inferred sequences at the root of T, to eventually obtain an unbiased estimate of the expected error. Blanchette et al.[22] used this approach to show that, given the genomes of 20 well-chosen mammalian species, the genome of the Boreoeutherian ancestor (ancestor of all eutherian mammals except Xenarthrans (e.g. armadillo) and Afrotherians (e.g. elephant)) could be reconstructed with only approximately 1% error. However, simulations are very computationally expensive and provide little understanding of the fundamentals of the problem. A more efficient approach is sought because AREA lends itself to use as a kernel for more general algorithms. For instance, even a very basic heuristic for the taxon selection problem (defined below) can require up to $O(n!)$ reconstruction error estimates to be performed. Another example would be in cases where an aggregate of different site-by-site reconstruction errors across an entire genome must be computed. This can happen in the presence of a varying mutation rate or gaps, and may require millions of instances of AREA to be solved.

In this paper, we seek a more efficient approach to estimate the expected reconstruction error for maximum likelihood inference for substitutions. We start by reviewing related work, giving basic definitions and notation, introducing a straightforward random sampling algorithm. We then describe a faster, heuristic sampling approach and prove that it can provide an upper bound on the error. Finally, results on biological and simulation data are presented in Section 6, demonstrating that our approach is applicable to a wide range of trees. Equipped with efficient algorithm for reconstruction error estimation, in Section 7 we consider the problem of species selection: Given a large phylogenetic tree with a particular internal node N of interest, which subset of k leaves yields the maximum information about ancestor N. The ability of fast and accurate error estimation can thus open the door to a number of applications.

2. Previous work

An exact method for computing the accuracy of ancestral reconstruction using parsimony, given a tree topology and stochastic model of evolution, was presented by Maddison.[23] Although efficient, this dynamic programming algorithm cannot be extended to more general models of parsimony where, for example, different transitions can be associated with different scores. Parsimony also suffers the drawback that, depending on the model of evolution used, the reconstruction accuracy when using parsimony may actually decrease as more taxa are added to the tree.[24] Maximum Likelihood (ML) provides a statistically robust framework for performing ancestral reconstructions under general models of evolution. Yang et al.[25] adapted Felsenstein's pruning algorithm[15] to efficiently estimate the most likely ancestral states of protein sequences with the

greatest marginal maximum likelihood. Others have published similar results.[26–28] In most of these studies, the accuracy of the method is indirectly measured by the relative contribution of the reconstructed ancestor to the overall likelihood of the tree, including the observed character states at the leaves. Some work has also been done to determine how topology relates to reconstruction error.[24,29–31] However, there is no known analog for Maddison's algorithm in the ML setting that exactly computes the reconstruction error given only the tree topology and model, and we conjecture that the problem is NP-Hard. Ma and Zhang[32] have recently shown that the problem does admit a fully polynomial time approximation scheme (FPTAS), but the complexity of their algorithm is $O(\frac{n^{17}}{\epsilon^8})$ for a DNA alphabet, making their result primarily of theoretical, rather than practical, use.

3. Definitions

Let $\mathcal{T} = (V, E)$ be a rooted phylogenetic tree with n leaves. The length of edge e is denoted $\ell(e)$. Throughout this paper we use the Jukes Cantor model of evolution[33] on the alphabet $A = \{a_1, a_2, ..., a_{|A|}\}$. Under this model, all ancestral states have equal prior probabilities ($\pi = \frac{1}{|A|}$) and the probability of a mutation occurring on edge e is $p_e = \frac{|A|-1}{|A|}(1 - e^{\ell(e)})$. Let set $D = \{d_1, d_2, ..., d_{|A|^n}\}$ represent the set of all possible assignments of character states to the leaves of $\mathcal{T}$. $\Pr[d_i|\mathcal{T}]$ is the likelihood of the tree for the site given leaf configuration d_i, and can be computed in $O(n \cdot |A|^2)$ time using Felsenstein's dynamic programming algorithm.[15] The marginal maximum likelihood ancestral reconstruction for a site given a leaf configuration $d \in D$ is

$$R(\mathcal{T}, d) = \underset{a \in A}{\operatorname{argmax}} \Pr[d|r = a, \mathcal{T}] \tag{1}$$

where r is the ancestral state at the root of $\mathcal{T}$. Given that the prior probabilities are all equal, we assume throughout that the true ancestral state is a_1, and it follows that the reconstruction error can be expressed as

$$RE(\mathcal{T}) = \sum_{d \in D} \Pr[d|r = a_1] \cdot \begin{cases} 0 & \text{if } \mathrm{R}(\mathcal{T}, d) = \{a_1\} \\ 1 & \text{otherwise.} \end{cases} \tag{2}$$

Note that in this study, we consider the unambiguous reconstruction error, and a reconstruction is only considered correct of a_1 is the unique solution returned by $R(\mathcal{T}, d)$. It is clear that a naive implementation of Equation 2 would require time $O(|A|^{n+2} \cdot n)$, which is impractical for even moderately large trees. In this paper, we develop heuristics to efficiently estimate this summation, focusing on a small number of terms that contribute most to the total reconstruction error.

4. Monte Carlo Simulation

An estimate of the reconstruction error can be obtained by a simulation, as mentioned above, which runs as follows. The true ancestral state is selected using the prior (equilibrium) distribution. The desired substitution model is then used to simulate random substitutions downwards along the branches until a configuration of states at the leaves is obtained. This configuration of leaf states is then used to predict the ancestral state. If it does so incorrectly, the trial is counted as an error event. Let K be the random variable denoting the total number of errors encountered after N trials. Because the outcomes of each trial are independent, $K \sim \mathrm{Bin}(N, p)$. The reconstruction error can be estimated as $\hat{p} = \frac{K}{N}$ and the normal approximation can be used to estimate the binomial confidence interval:

$$p \in \left[\hat{p} - z_{1-\alpha/2} \sqrt{\frac{\hat{p}(1-\hat{p})}{N}}, \hat{p} + z_{1-\alpha/2} \sqrt{\frac{\hat{p}(1-\hat{p})}{N}} \right] \tag{3}$$

where $z_{1-\alpha/2}$ is the critical value of the two two-tailed normal distribution at level α (e.g. $z_{1-\alpha/2} = 1.96$ for 95% confidence). This approach is powerful in the sense that it can be used to accurately estimate the error with very few assumptions about the underlying model. However, as the true reconstruction

error becomes very small, it is possible that an intractable number of samples will be required to obtain a reasonable estimate. This is because the size of the confidence interval decreases proportional to $\sqrt{N}$. Furthermore, if $K = 0$ (as expected if $p < \frac{1}{N}$), then the confidence interval is undefined.

5. Prioritized Enumeration Algorithm

In this section, we describe an enumeration approach that will accurately estimate the reconstruction error in much fewer than $\frac{1}{RE(\mathcal{T})}$ trials. This method is based on the observation that a relatively small subset of leaf configurations often account for nearly all of the reconstruction error.

5.1. *Mutation Scenarios*

We analyze leaf configurations in relation to the mutation scenarios that can give rise to them. Define a *mutation scenario* $m \subseteq E$ as a set of edges of $\mathcal{T}$ where mutations occur. The reconstruction error can be rewritten in terms of the error of all $|\mathcal{P}E| = 2^{|E|}$ possible scenarios, where $\mathcal{P}E$ is the power set of E. We then have

$$RE(\mathcal{T}) = \sum_{m \in \mathcal{P}E} \Pr[m] \cdot RE(\mathcal{T}|m), \tag{4}$$

where

$$\Pr[m] = \prod_{e \in m} p_e \prod_{e \in E-m} (1 - p_e),$$

and the reconstruction error for an individual scenario can in principle be computed by analyzing all possible $(|A| - 1)^{|m|}$ leaf state configurations that it can give rise to:

$$RE(\mathcal{T}|m) = \sum_{d \in D} \Pr[d|m] \cdot \begin{cases} 0 & \text{if } R(\mathcal{T}, d) = a_1 \\ 1 & \text{otherwise.} \end{cases} \tag{5}$$

$\Pr[d|m]$ is the probability of a particular leaf configuration occurring on the tree given that mutations only occur on the edges of m. If d assigns a mutated state to a leaf that does not lie below m, then $\Pr[d|m] = 0$.

There are an exponential number of possible mutation scenarios, and the cost of computing the reconstruction error of a single scenario is exponential in the size of that scenario. As such, using (4) to exhaustively compute the error is no more efficient than the original definition (2). However, we hypothesize that only a small fraction of scenarios contribute significantly to the reconstruction error. Furthermore, we expect that the RE of these scenarios, $RE(\mathcal{T}|m)$ will tend to be relatively large, and therefore faster to estimate accurately using sampling. Intuitively, these significant scenarios will often contain mutations located closer to the root of the tree, where a smaller number of mutations can affect more leaves, causing a greater loss of signal. The two primary computational challenges of this approach are to rapidly determine the most relevant scenarios, and quickly estimate the reconstruction error of each. These tasks are equivalent to identifying largest terms in Equation (4) and computing the summation in Equation (5), respectively, and are explained in the following two subsections.

5.2. *Prioritization Strategy*

The ideal order in which to explore mutation scenarios is decreasing on $\Pr[m] \cdot RE(\mathcal{T}|m)$. This way, if only a k-subset of all scenarios are evaluated, the accuracy of the estimated error will be maximal. It is unknown how to exactly compute the k scenarios that contribute the most to the reconstruction error, but we use the next two lemmas below in our algorithm to efficiently estimate them. We say that a leaf x_i lies *below* a scenario m, which we denote $x_i \prec m$, if there exists an edge $e \in m$ on the path from x_i to the root of $\mathcal{T}$. We also use a similar notation to compare two scenarios: $m \preceq m'$ if, for all $e \in m$, there exists an edge $e' \in m'$ such that e lies on the path between e' and the root (including the possibility that $e = e'$).

Lemma 5.1. *Let m be a mutation scenario where no two mutations are on the same root-to-leaf path, and m' be any scenario obtainable by moving mutations in m away from the root in $\mathcal{T}$, then*

$$RE(\mathcal{T}|m') \leq RE(\mathcal{T}|m).$$

Proof. Let m' be obtained by moving edge $e_v \in m$ to one of its children, e_u. For any leaf configuration d such that $R(\mathcal{T}, d) = a_1$ and $\Pr[d|m] > 0$, there exists a unique configuration d' obtainable from d by assigning all leaves below e_v but not below e_u to state a_1. Furthermore, $\Pr[d'|m'] = \Pr[d|m]$ because $|m| = |m'|$ and no mutations occur on the same root-to-leaf path. From Lemma A.1 (see Appendix), $R(\mathcal{T}, d) = a_1 \rightarrow R(\mathcal{T}, d') = a_1$. $RE(\mathcal{T}|m') \leq RE(\mathcal{T}|m)$ follows directly from Equation (5), and the proof can be completed by induction on the number of edges moved. □

Lemma 5.2. *Let m and m' be two mutation scenarios such that $|m| \leq |m'|$ and let ℓ_{max} be the length of the longest edge in $\mathcal{T}$. Then*

$$\frac{\Pr[m']}{\Pr[m]} < \tau^{|m'|-|m|},$$

where $\tau = \frac{(|A|-1)(1-e^{-\ell_{max}})}{1+e^{-\ell_{max}}}$ and $\tau < 1$ whenever $\ell_{max} < -\ln \frac{|A|-2}{2(|A|-1)}$. (Proof follows directly from the definition of the Jukes Cantor model.)

For example, for $|A| = 4$, $\tau < 1$ when $\ell_{max} < 1.09$, which corresponds to an extremely long branch, never observed in trees that lend themselves to ancestral reconstruction. In the case of the mammalian tree used in Section 6, $\ell_{max} = 0.18$ and $\tau = 0.27$, which means that the probability of a scenario will tend to drop quickly with its size (but note also that the *number* of scenarios grows exponentially with their size).

Our algorithm enumerates mutation scenarios in increasing order of their size. Scenarios of the same size are evaluated in lexicographic order, where edges are ranked based on their depth in the tree (i.e. the edges closest to the root come first). Lemmas 5.1 and 5.2 show that this is generally a good approximation to a lexicographic ordering on $(\Pr[m], RE(\mathcal{T}|m))$, which we find to be an effective estimate of the ranking based on the true objective, $\Pr[m]RE(\mathcal{T}|m)$. Figure 1 provides an example of the enumeration procedure. The edges of the phylogenetic tree are first ranked (Figure 1(a)), then the scenarios are visited in the order of a breadth-first search of the tree shown in Figure 1 (b). It is important to note that this tree traversal visits each scenario once, and that all scenarios below m in the search tree are supersets of m. Also, as shown in the following section, mutation scenarios with multiple mutations on a single root-to-leaf path are never explicitly evaluated. Therefore the fact that Lemma 5.2 does not apply in these cases does not impact its applicability here.

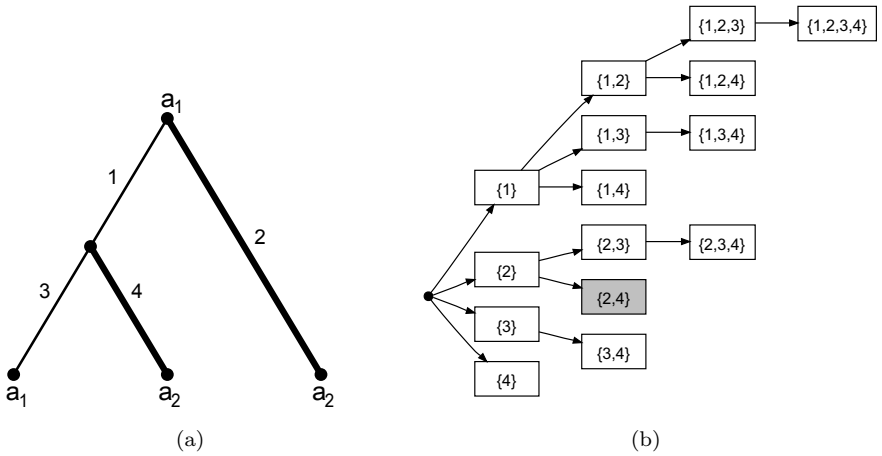
5.3. *Scenario Evaluation and Bounding the Search*

The algorithm must be able to quickly estimate $RE(\mathcal{T}|m)$ to effectively explore the search space. The cost of exactly computing this value, as described in Equation (5), is $O((|A|-1)^{|m|} \cdot n \cdot |A|^2)$, which is impractical for all but the smallest scenarios. We therefore propose a procedure to efficiently estimate $RE(\mathcal{T}|m)$ by considering many scenarios at once. For a given scenario m, consider the following "pessimistic" leaf configuration $pess(m) = x_1, x_2, ..., x_n$, where

$$x_i = \begin{cases} a_1 \text{ if } m \text{ contains no edge on the path from } i \text{ to the root} \\ a_2 \text{ if } m \text{ contains at least one edge the path from } i \text{ to the root} \end{cases}$$

The leaf configuration $pess(m)$ is pessimistic because all mutations lead to the same character a_2, which causes the maximal probability of reconstruction error. Note that it is possible that $\Pr[pess(m)|m] = 0$ (when there are two mutations in the same lineage). $RE^{pess}(\mathcal{T}|m)$ is used to denote the pessimistic estimate of $RE(\mathcal{T}|m)$ and is computed in $O(n \cdot |A|^2)$ as follows:

$$RE^{pess}(\mathcal{T}|m) = \begin{cases} 0 & \text{if } R(\mathcal{T}, pess(m)) = a_1 \\ 1 & \text{otherwise.} \end{cases}$$



(a) (b)

Fig. 1. (a) Phylogenetic tree where a mutation scenario $\{2, 4\}$ is highlighted. The ancestral state (a_1) mutates to a_2 along branches 2 and 4. (b) Corresponding search tree, traversed breadth first by the heuristic sampling algorithm. Scenario $\{2, 4\}$ is explored on the second level.

We now present a fast algorithm for estimating $RE^{pess}(\mathcal{T}|m)$, before relaxing it produce a more accurate estimator. Let $ext(m) = \{m' : m \subseteq m'\}$ be the set of scenarios that extend m and let $below(m) = \{m' : m' \preceq m\}$ be the set of scenarios below m. The following two properties, which directly follow from Lemma A.1, enable a branch and bound approach to further speed up the search.

Lemma 5.3. *If* $RE^{pess}(\mathcal{T}|m) = 1$*, then* $RE^{pess}(\mathcal{T}|m') = 1, \forall m' \in ext(m)$

Lemma 5.4. *If* $RE^{pess}(\mathcal{T}|m) = 0$*, then* $RE^{pess}(\mathcal{T}|m') = 0, \forall m' \in below(m)$

Lemma 5.3 allows us to prune the search tree at any scenario that does not yield a correct reconstruction under the pessimistic evaluation, since all scenarios below it in the search tree are supersets and will necessarily fail the pessimistic evaluation. In the example search shown in Figure 1(b), if $RE(\{2\}) = 1$, then the three nodes below $\{2\}$ in the search tree will also have $RE^{pess} = 1$ and need not be visited. The total probability of $ext(m)$ can be computed in linear time:

$$\Pr[m \cap ext(m)] = \prod_{e \in m} p_e \prod_{e \notin (m \cup ext(m))} (1 - p_e).$$

$\Pr[m \cap below(m)]$ can be computed in a similar manner. Lemma 5.4 allows for a similar optimization, except only extensions consisting of edges below m in the tree can be pruned. In the example, if $RE(\mathcal{T}|\{1\}) = 1$ then the search tree can be pruned at $\{1, 3\}$ and $\{1, 4\}$ but not $\{1, 2\}$.

Let $succ(m)$ be the set of all scenarios below m on the *search* tree. Pruning the search tree as described above does not take full advantage of Lemmas 5.3 and 5.4, because $ext(m)$ and $below(m)$ may contain scenarios that are not in $succ(m)$. These sets grow exponentially with $|m|$, however, so storing in memory all scenarios that have and have not been visited quickly becomes infeasible. Instead, a small cache is used to store the outcome of the most recently visited scenarios in the breadth-first search. As each new scenario is visited, the cache is checked before the scenario is evaluated. If it extends or lies below an appropriate scenario in the cache, its $RE(m|\mathcal{T})$ can be retrieved without an explicit evaluation. Let $RE^{pess}(\mathcal{T}) = \sum_{m \in \mathcal{P}_E} \Pr[m] \cdot RE^{pess}(\mathcal{T}|m)$. Lemmas 5.3 and 5.4 guarantee that when using the pessimistic evaluation of scenarios as described, the algorithm will exactly compute $RE^{pess}(\mathcal{T})$. Furthermore, his value is an upper bound for $RE(\mathcal{T})$ (Theorem A.1, see Appendix).

Assuming all mutations mutate to the same state is simplistic, as it becomes more likely that independent mutations give rise to two or more different states as the size of m increases. This situation is accounted for by estimating $RE(\mathcal{T}|m)$ using a small number of random samplings, whereby scenario m is used to generate a set

of r random leaf configurations. The error is estimated as $\frac{k}{r}$ where k is the number of failed reconstructions out of r random trials. Finally, Lemmas 5.3 and 5.4 are relaxed to apply when $RE(\mathcal{T}|m) \geq 1-\alpha$ and $RE(\mathcal{T}|m) < 1-\alpha$ respectively, where α is a chosen constant. Pseudocode for both the pessimist and relaxed sampling versions is provided in Algorithm 5.1.

Algorithm 5.1 Estimate $RE(\mathcal{T})$

```
TotalError ← 0;
ScenarioQueue q ← {};
while |q| > 0 do
  m ← q.pop()
  error ← { RE^pess(T|m)    if pessimist mode
          { RE^sample(T|m)  if sample mode
  if error ≥ 1 − α then
    TotalError ← TotalError + error · Pr[m ∪ succ(m)]
  else
    TotalError ← TotalError + error · Pr[m ∪ succ(m) ∩ below(m)]
    for e ∈ succ(m) − below(m) do
      q.push(m ∪ e)
return TotalError
```

6. Results

The prioritized enumeration algorithm was implemented in C++. Since we are unable to exactly compute the true error for non-trivial trees, the random Monte Carlo sampling approach was also implemented to use as a baseline for comparison. We carried out various experiments to determine the accuracy of the upper bound and estimates proposed here.

We first estimated the reconstruction error for the Boreoeutherian ancestor based on an actual mammalian phylogeny made of 32 species.[34,35] Figure 2 shows the average estimates obtained from 100 separate Monte Carlo simulations, as a function of the number of trials, N. This yields an unbiased estimator of the true reconstruction error, together with a mean confidence interval for that value. The mean estimates obtained from the prioritized enumeration, either using the pessimistic upper bound or the random sampling version [a] are plotted as well. The standard deviation of the estimates obtained with the sampling-based prioritize enumeration approach is too small to plot ($\sigma = 0.00043$). We observe that, as expected, the pessimistic version of the algorithm overestimates the reconstruction error. However, the sampling approach quickly converges to the correct value and, moreover, it does so with much less variance than the Monte Carlo simulations. In both cases, a fairly accurate estimate of the RE is obtained by our algorithm after fewer than 50 scenarios.

We then estimated the accuracy of our reconstruction errors on random trees. A set of 50 trees of size n and expected total number of substitutions μ were randomly generated using the Yule model of speciation.[36] Branch lengths were assigned using a uniform random distribution and scaled to total μ. For each tree, the 99% confidence interval $I_{99\%}$ for the RE was first estimated based on 5000 Monte Carlo simulations. Reconstruction errors were then estimated, evaluating a number k of leaf configuration ranging from 5 to 5000. Our first set of random trees have 50 leaves, with $\mu = 1$, and with an average 99% reconstruction error confidence interval of 0.007 ± 0.002. Figure 3(a) shows, for each value of k, the fraction of the trees for which the estimate obtained after k configurations lies within $I_{99\%}$. The second set of trees (Figure 3(b))

[a] $r = 2$ and relaxation parameter $\alpha = 0$ consistently gave the best performance and were used for all experiments.

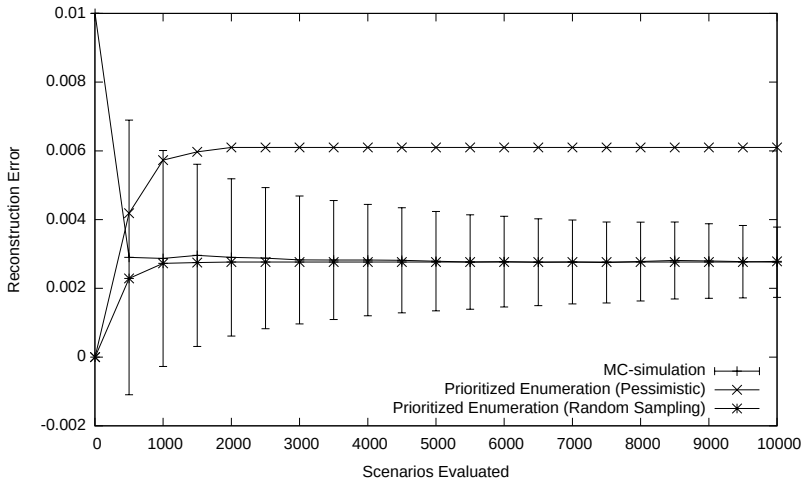


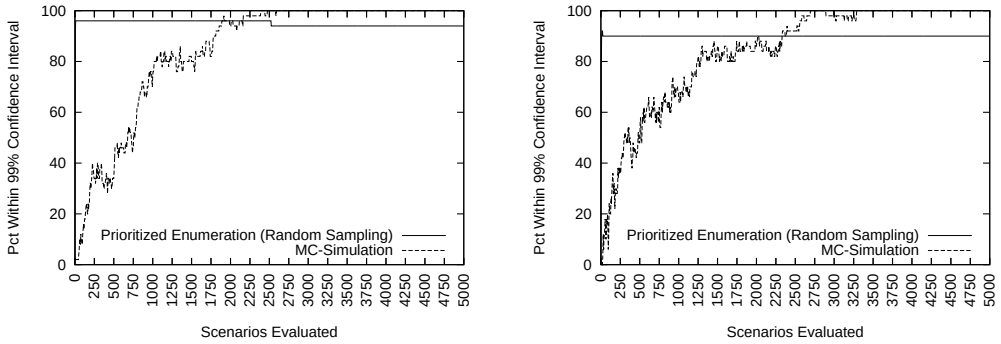
Fig. 2. Reconstruction Error of the Boreoeutheirian common ancestor. 99% confidence interval is shown for the Monte Carlo simulation.

was generated with $n = 100, \mu = 5$ and had an average 99% confidence interval for the reconstruction error of 0.020 ± 0.004. In terms of running time, the cost of evaluating each scenario is effectively identical to that of a single simulation trial. The graphs therefore show that the heuristic algorithm was able to accurately estimate 98% and 82% of the trees in the batches, respectively, an order of magnitude faster than running the full simulation. The same procedure was also applied to trees randomly selected from PANDIT 17.0,[37] a database of protein phylogenies. Two groups of 50 trees were created: the first with between 25 and 50 taxa (mean 99% C.I. $= 0.007 \pm 0.002$; Figure 3(c)) and the second with between 50 and 100 taxa (mean 99% C.I. $= 0.003 \pm 0.001$, Figure 3(d)). Clearly, the smaller the RE to be estimated, the more accurate and faster our approach is, as fewer scenarios need to be evaluated. The algorithm performed worse overall on the PANDIT trees, indicating that balance and branch-length distribution are factors as well.

Figure 4 reports the estimated reconstruction error obtained at each ancestral node of the mammalian phylogeny. Of course, these numbers do not reflect the accuracy of actual reconstructed ancestral sequences (as reported in Blanchette et al.[22]), because they only model substitutions and ignore alignment issues. Nonetheless, the estimated REs are informative about the ancestral nodes that can best be reconstructed. We first note that reconstruction errors vary significantly across the tree (from 0.11% to 3.5%). Excluding recent primate ancestors, the nodes that can best be reconstructed correspond to early ancestors that lived during the mammalian radiation. Indeed, those are the ones for which the largest number of nearly independent noisy copies exist. Interestingly, the RE of the ancestral primate is nearly two times higher than that of the Boreoeutherian ancestor, which predates it by approximately 25 Million years. However, all human ancestors, except the eutherian mammals ancestor, can be reconstructed with less than 0.7% error. On the other hand, nodes adjacent to three long branches (mouse-rat ancestor, rabbit-pika ancestor, hedgehog-shrew ancestor) have much higher reconstruction errors.

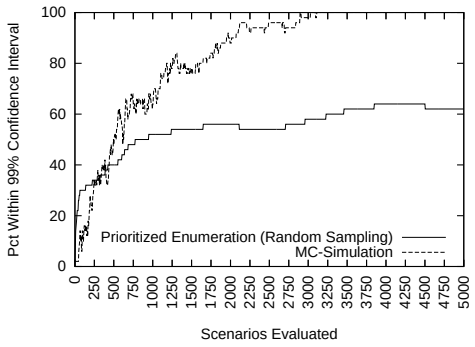
7. The Taxon Selection Problem

A natural application for an efficient algorithm to compute RE is the taxon selection for the ancestral reconstruction problem, introduced by Li et al.[39] This problem asks to select among the leaves of a large phylogenetic tree, a set of k taxa that will allow reconstructing a given ancestral node with minimum error, provided the tree and a model of evolution. Using the Maddison algorithm as a kernel, Li et al. showed

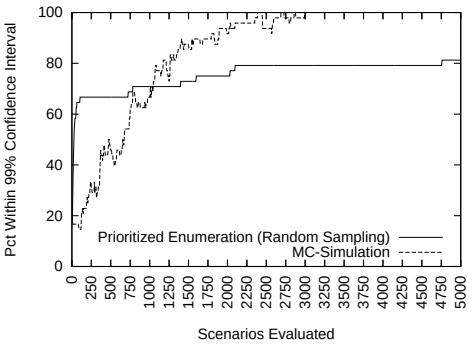


(a) Random Simulation ($n = 50$, $\mu = 1$)

(b) Random Simulation ($n = 100$, $\mu = 5$)



(c) PANDIT ($25 \leq n < 50$)



(d) PANDIT ($50 \leq n < 100$)

Fig. 3. Fraction of errors estimated within the 99% confidence interval by number of mutation scenarios evaluated for batches of simulated and real trees.

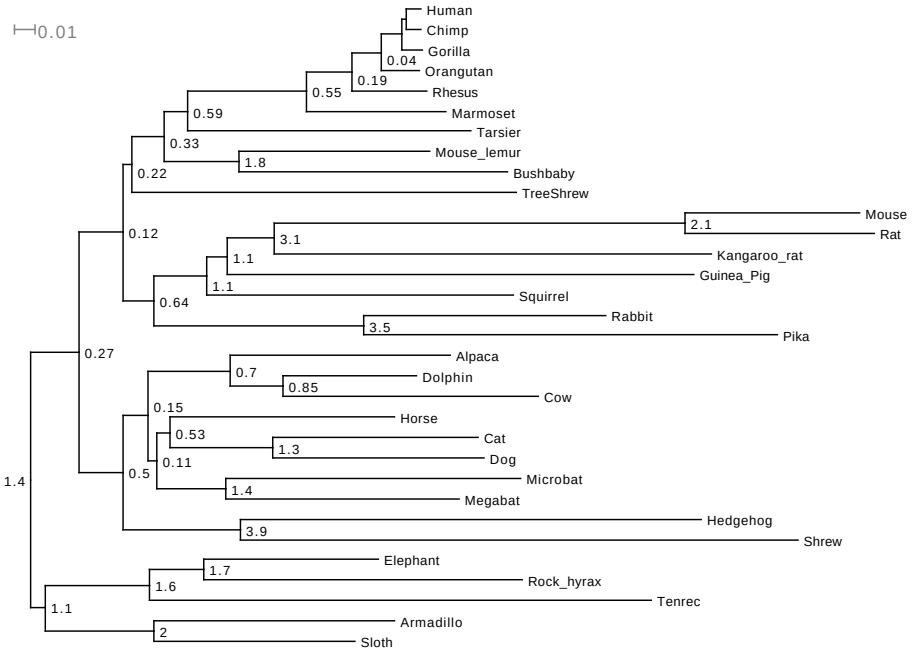


Fig. 4. Estimated reconstruction error (in percent) for all internal nodes of the eutherian mammals phylogeny (Human/Chimp and Human/Chimp/Gorilla ancestors both have RE=0.013%).

that a backward selection algorithm is an effective heuristic for genome selection under parsimony.[39] The algorithm starts from the complete set of species and greedily prunes the least valuable leaves from the tree until k remain.

We have investigated a similar algorithm to select an ideal set of species for ML-based ancestral reconstruction. The algorithm is similar to Li et al.'s. The only major difference is that when multiple species candidates for removal yield RE that cannot be distinguished based on our estimation algorithms, the species that is the furthest away from the target ancestral node is the one selected for pruning. Figure 5 reports the order in which species are removed, and the resulting RE for the Boreoeutherian ancestor. We note that a RE of less than 1% can be achieved by selecting only seven (armadillo through orangutan) slow-evolving species that sample the outgroup (Xenarthra, Afrotheria), and the main descendant phyla (primates, laurasia, etc.). This algorithm evaluates $O(n!)$ different topologies requiring an efficient reconstruction error estimate. The reduced variance of our approach, relative to MC simulation, is also desirable for this application: In a comparable amount of time, it will produce a much more stable solution.

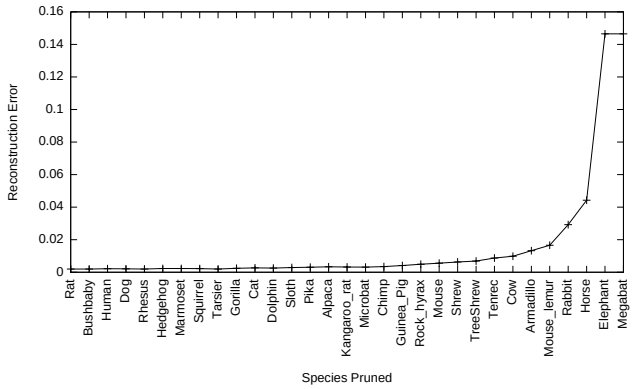


Fig. 5. Greedy species selection algorithm. The order in which species are pruned is presented along with the reconstruction error at each step. The final species is Sloth

8. Conclusion

We have presented a novel algorithm for estimating the error of maximum likelihood ancestral reconstruction. It is based upon quickly enumerating and evaluating the mutation scenarios that contribute the most to $RE(\mathcal{T})$ using a branch-and-bound search strategy. The assumption of a pessimistic leaf configuration is used to guarantee a non-trivial upper bound on the error. This assumption is then relaxed by performing a restricted sampling for each scenario, in order to more closely approximate the true error. We used our method to compute the RE of the Boreoeutherian mammal ancestor, the subject of several important ancestral reconstruction studies, and it quickly converged on an extremely accurate estimate. Benchmarks run on batches of simulated trees and those drawn at random from an online database showed that estimated reconstruction errors were within the 99% confidence interval of the unbiased estimator in the majority of cases. The best performance was seen in cases where the error is very small, a situation that we foresee in most real-life reconstruction applications.

The high speed and low variance of the prioritized enumeration algorithm make it a useful building block for solutions to complex problems. In this study, it was used to quickly determine all the errors of all internal nodes of the mammalian tree, and as an objective function in a greedy heuristic solution to the taxon selection problem. We envision more applications, such as large genomes that require multi-site analysis due

to hypothesized gaps or changes in substitution rate parameters, that could require efficient estimation of small errors. One avenue of future work that could lead to an increase of applicability of the algorithm is by exploring more complex models of DNA substitution. While not all properties stated in this paper hold for more general models, we conjecture that the overall strategy will be effective for any model where the probability of mutation is relatively low. Another area in which our approach can potentially be optimized is in the scenario enumeration. The current approach has a sound basis, but it may be possible to more directly sample the relevant mutation scenarios, using more information about the topology and branch lengths.

Acknowledgements

M.B. and G.H. are funded by NSERC. We thank Leonid Chindelevitch for initial discussions on this problem.

Appendix A. Upper Bound

Lemma A.1. *If $R(\mathcal{T}, d) = a_s$ then $R(\mathcal{T}, d') = a_s$ where d' is obtained from d by reassigning to a_s the state of any subset of leaves.*

Proof. We begin by showing that $\Pr[v = a_s|d'] \geq \Pr[v = a_s|d]$ for any node v on the tree. If v is a leaf, $\Pr[v = a_s|d'] = 1$ and $\Pr[v = a_s|d] = 0$ if v is reassigned and $\Pr[v = a_s|d'] = \Pr[v = a_s|d]$ otherwise. If v is an internal node, we make the inductive hypothesis that the inequality holds for v's children. That is, $\Pr[u = a_s|d'] = \Pr[v = a_s|d] + \delta_u$ where $\delta_u \geq 0$. To simplify the notation, we use p and q to respectively denote $\Pr[u = a_s|v = a_s]$ and $\Pr[u = a_s|v = a_t]$ for $a_t \neq a_s$. $\Pr[v = a_s|d]$ can be computed from its children as follows:[15]

$$\Pr[v = a_s|d] = \prod_{u \in \text{child}(v)} \left(p \cdot \Pr[u = a_s|d] + q \cdot \sum_{i \neq s} \Pr[u = a_i|d] \right)$$

By the inductive hypothesis,

$$\begin{aligned}
\Pr[v = a_s|d'] &= \prod_{u \in \text{child}(v)} \left(p \cdot (\Pr[u = a_s|d] + \delta_u) + q \cdot \left(\left(\sum_{i \neq s} \Pr[u = a_i|d] \right) - \delta_u \right) \right) \\
&= \prod_{u \in \text{child}(v)} \left(p \cdot \Pr[u = a_s|d'] + q \cdot \sum_{i \neq s} \Pr[u = a_i|d'] + \delta_u(p - q) \right) \\
&\geq \Pr[v = a_s|d]
\end{aligned}$$

since, under the Jukes Cantor model, $p \geq q$. The same procedure can be used to show that that $\Pr[v = a_t|d'] <= \Pr[v = a_t|d]$ for $t \neq s$. As before, the leaf case is trivial. If v is an internal node,

$$\Pr[v = a_t|d] = \prod_{u \in \text{child}(v)} \left(p \cdot \Pr[u = a_t|d] + q \cdot \sum_{i \neq t} \Pr[u = a_i|d] \right).$$

From the induction hypothesis that $\Pr[u = a_t|d'] = \Pr[u = a_t|d]\delta_u$ for $u \in child(v)$,

$$\begin{aligned}
&= \prod_{u \in \text{child}(v)} \left(p \cdot (\Pr[u = a_t|d] - \delta_u) + q \cdot \left(\left(\sum_{i \neq t} \Pr[u = a_i|d] \right) + \delta_u \right) \right) \\
&= \prod_{u \in \text{child}(v)} \left(p \cdot \Pr[u = a_t|d'] + q \cdot \sum_{i \neq t} \Pr[u = a_i|d'] + \delta_u(q - p) \right) \leq \Pr[v = a_s|d].
\end{aligned}$$

Since $\Pr[v = a_s|d'] \geq \Pr[v = a_s|d]$ and $\Pr[v = a_t|d'] \leq \Pr[v = a_t|d]$ for any $t \neq s$, then $\text{argmax}_{i \in A} \Pr[v = a_i|d] = a_s \rightarrow \text{argmax}_{i \in A} \Pr[v = a_i|d'] = a_s$. □

Theorem A.1. *Let $RE^{pess}(\mathcal{T})$ be the reconstruction error obtained by pessimistic scenario evaluation. Then $RE(\mathcal{T}) \leq RE^{pess}(\mathcal{T})$.*

Proof. It is sufficient to show that for any mutation scenario m, $RE(\mathcal{T}|m) \leq RE^{pess}(\mathcal{T}|m)$. The inequality trivially holds when $RE(\mathcal{T}|m) = 0$, so we need only consider the case where m is a scenario such that $RE(\mathcal{T}|m) > 0$. By definition, there exists a leaf configuration $d = \{d_1, d_2, ..., d_n\}$ such that $R(\mathcal{T}|d) = a_s \neq a_1$. Now consider the scenario $pess(m)$, where every leaf below a mutation has state a_s and all other leaves have the true ancestral state a_1. Both d and $pess(m)$ only contain non-a_1 states at leaves which are below mutations. It follows that $pess(m)$ can be constructed from d by changing the states of a subset of leaves to a_s. From Lemma A.1, $R(\mathcal{T}, pess(m)) = a_s$, giving $RE^{pess}(\mathcal{T}|m) = 1$, which implies that $RE(\mathcal{T}|m) \leq RE^{pess}(\mathcal{T}|m)$. □

References

1. B. Chang, K. Jonsson, M. Kazmi, M. Donoghue and T. Sakmar, *Molecular biology and evolution* **19**, 1483 (2002).
2. D. Kuang, Y. Yao, D. MacLean, M. Wang, D. Hampson and B. Chang, *Proceedings of the National Academy of Sciences* **103**, p. 14050 (2006).
3. J. Thornton, E. Need and D. Crews, *Science* **301**, 1714 (2003).
4. M. Blanchette, E. Green, W. Miller and D. Haussler, *Genome Res.* **14**, 2412 (2004).
5. B. Paten, J. Herrero, S. Fitzgerald, K. Beal, P. Flicek, I. Holmes and E. Birney, *Genome Res.* **18**, p. 1829 (2008).
6. J. Taylor, S. Tyekucheva, D. King, R. Hardison, W. Miller and F. Chiaromonte, *Genome Res.* **16**, p. 1596 (2006).
7. J. Kim and S. Sinha, *Bioinformatics* **23**, p. 289 (2007).
8. J. Kim, X. He and S. Sinha, *PLoS Genetics* **5** (2009).
9. M. Blanchette, W. Kent, C. Riemer, L. Elnitski, A. Smit, K. Roskin, R. Baertsch, K. Rosenbloom, H. Clawson, E. Green *et al.*, *Genome research* **14**, 708 (2004).
10. B. Paten, J. Herrero, K. Beal, S. Fitzgerald and E. Birney, *Genome Research* **18**, p. 1814 (2008).
11. M. Brudno, C. Do, G. Cooper, M. Kim, E. Davydov, N. Program, E. Green, A. Sidow and S. Batzoglou, *Genome research* **13**, 721 (2003).
12. N. Bray and L. Pachter, *Genome Research* **14**, 693 (2004).
13. W. Fitch, *Systematic zoology* **20**, 406 (1971).
14. D. Sankoff, *SIAM Journal on Applied Mathematics* , 35 (1975).
15. J. Felsenstein, *Journal of molecular evolution* **17**, 368 (1981).
16. L. Chindelevitch, Z. Li, E. Blais and M. Blanchette, *J. of Bioinformatics and Comp. Biol.* **4**, 721 (2006).
17. A. Diallo, V. Makarenkov and M. Blanchette, *Journal of Computational Biology* **14**, 446 (2007).
18. J. Ma, L. Zhang, B. Suh, B. Raney, R. Burhans, W. Kent, M. Blanchette, D. Haussler and W. Miller, *Genome Res.* **16**, p. 1557 (2006).
19. G. Bourque, P. Pevzner and G. Tesler, *Genome Res.* **14**, 507 (2004).
20. D. Sankoff and M. Blanchette, *Journal of Computational Biology* **5**, 555 (1998).
21. J. Ma, A. Ratan, B. Raney, B. Suh, W. Miller and D. Haussler, *PNAS* **105**, p. 14254 (2008).
22. M. Blanchette, E. Green, W. Miller and D. Haussler, *Genome Res.* **14**, 2412 (2004).
23. W. Maddison, *Systematic Biology* **44**, p. 474 (1995).
24. G. Li, M. Steel and L. Zhang, *Systematic Biology* **57**, 647 (2008).
25. Z. Yang, S. Kumar and M. Nei, *Genetics* **141**, 1641 (1995).
26. J. Koshi and R. Goldstein, *Journal of molecular evolution* **42**, 313 (1996).
27. M. Pagel, *Syst. Biol* **48**, 612 (1999).
28. D. Schluter, T. Price, A. Mooers and D. Ludwig, *Evolution* , 1699 (1997).
29. B. Lucena and D. Haussler, *Systematic biology* **54**, 693 (2005).
30. A. Mooers, *Systematic biology* **53**, 809 (2004).
31. J. Zhang and M. Nei, *Journal of molecular evolution* **44**, p. 139 (1997).
32. B. Ma and L. Zhang, *Journal of Combinatorial Optimization, IN PRESS* (2009).
33. T. Jukes and C. Cantor, *Mammalian protein metabolism* **3**, 21 (1969).
34. W. Murphy, E. Eizirik, S. O'Brien, O. Madsen *et al.*, *Science* **294**, 2348 (2001).
35. D. Karolchik, R. Kuhn, R. Baertsch, G. Barber *et al.*, *Nucleic Acids Research* **36**, p. D773 (2008).
36. G. Yule, *Phil. Trans. Royal Soc. of London. Series B, Containing Papers of a Biological Character* , 21 (1925).
37. S. Whelan, P. de Bakker and N. Goldman, *Bioinformatics* **19**, 1556 (2003).
38. D. Huson, D. Richter, C. Rausch, T. Dezulian, M. Franz and R. Rupp, *BMC bioinformatics* **8**, p. 460 (2007).
39. G. Li, J. Ma and L. Zhang, p. 110 (2007).

OPTIMIZATION METHODS FOR SELECTING FOUNDER INDIVIDUALS FOR CAPTIVE BREEDING OR REINTRODUCTION OF ENDANGERED SPECIES

WEBB MILLER†

Center for Comparative Genomics and Bioinformatics, Penn State University Park, PA 16802

STEPHEN J. WRIGHT

Computer Sciences Department, University of Wisconsin Madison, WI 53706

YU ZHANG

Department of Statistics, Penn State University Park, PA 16802

STEPHAN C. SCHUSTER

Center for Comparative Genomics and Bioinformatics, Penn State University Park, PA 16802

VANESSA M. HAYES

Children's Cancer Institute Australia, University of New South Wales Randwick, NSW 2031, Australia

Methods from genetics and genomics can be employed to help save endangered species. One potential use is to provide a rational strategy for selecting a population of founders for a captive breeding program. The hope is to capture most of the available genetic diversity that remains in the wild population, to provide a safe haven where representatives of the species can be bred, and eventually to release the progeny back into the wild. However, the founders are often selected based on a random-sampling strategy whose validity is based on unrealistic assumptions. Here we outline an approach that starts by using cutting-edge genome sequencing and genotyping technologies to objectively assess the available genetic diversity. We show how combinatorial optimization methods can be applied to these data to guide the selection of the founder population. In particular, we develop a mixed-integer linear programming technique that identifies a set of animals whose genetic profile is as close as possible to specified abundances of alleles (i.e., genetic variants), subject to constraints on the number of founders and their genders and ages.

1. Introduction

It is generally agreed that techniques from genetics and genomics can be useful for understanding, and ideally sometimes preventing, the process of extinction[1,2,3]. One facet of species-conservation efforts, when appropriate, is to breed animals of an endangered species in captivity, e.g., in zoos and wildlife parks, with the goal of releasing them back into the wild at some time in the future. In some cases, the number of animals has dropped so low that all members of a species have been captured, as was done in North America for the California condor, red wolf and black-footed ferret. A better approach may be to identify species whose numbers are declining sharply but where more animals exist than can be supported in captivity, and to select from among the wild animals a representative subset to serve as the founder population for the captive breeding program.

In such cases, there is broad agreement among wildlife management officials that a goal should be for the founder population to capture at least, say, 95% of the genetic diversity of the wild population. However, there is disagreement over precisely what this means and how to achieve it. Perhaps the most common reasoning goes as follows[4]. Suppose a genomic position has two alleles in the population, with frequencies p and $1–p$. A founder population consisting of a random sample of n animals has $2n$ instances of that position, and the probability that all of them contain the first allele is p^{2n}. Thus, the probability that the n animals have at least one copy of each allele (rather than $2n$ copies of the same allele) is:

$$1 – p^{2n} – (1–p)^{2n}.$$

† Correspondence: webb@bx.psu.edu

If we want to obtain, with 95% certainty, both alleles at each random locus that occurs with a frequency of at least 0.05 in the wild population, then 30 founders are adequate, as is shown by evaluating the formula with $p = 0.05$ and $n = 30$. To have a 95% chance of capturing an allele with a frequency of only 1% requires 150 founders; 30 founders will probably not contain it.

The deficiency in this reasoning is that alleles don't occur at random in a wild population. Instead, a given allele is likely to occur more frequently in one sub-population than another, e.g., because of geographical barriers to gene flow. Thus, for example, picking all of the founders from one sub-population can easily miss alleles that are common overall. In practice, without an accurate survey of the genetic profile of the species, the number, degree of differentiation, and geographical locations of the relevant sub-populations may not be known.

Modern methods for sequencing and genotyping make it possible to directly assess the genetic diversity of a species and identify its population structure; sequencing a small sample of geographically distinct individuals can identify a number of the available genetic variants, and genotyping methods can then efficiently determine the genetic make-up of a large number of individuals population-wide. This paper describes how the resulting data can be used to select an optimal subset of the genotyped individuals that best matches specified allele frequencies, while satisfying additional constraints such as the total number of individuals selected, the number of males, and a suitable distribution of ages. Also, we show that allowing a bit of flexibility in the choice of founders can simplify the computation. In another formulation, we know the average genetic characteristics in sub-populations of the species, but not the genotypes of the individual candidates for the founder population.

One way to select the target genetic profile for the founder population is to match what is observed in living animals. In that case, random sampling would be justified if there is no population-genetic structure to the sampled population. Using the genotyping data required by our approach, one could run a program like STRUCTURE[5] to see if such structure exists.

The goal of recapitulating the distribution of alleles found in the overall population is by no means the only approach. We believe that in some cases a preferable goal may be to restore the balance that existed before the species was affected by the onslaught of industrial pollution, pesticides, disease, etc. in the last century or two. (We assume here that the pollution has been cleaned up, since we wouldn't want to undo any progress the species has made in dealing with it.) To support this approach, we are developing improved methods for sequencing the DNA from museum specimens[6,7], which can be used to determine past allele frequencies and population structure, and thereby identify a more natural target allele distribution for a founder population.

Another option for the target profile is where all alleles have frequency 0.5. For each locus, this minimizes the probability that an allele will be lost due to random genetic drift in the captive population. (This observation can be rigorously proved by martingale theory.) Moreover, this choice maximizes the genetic diversity in the captive population, since the probability that two randomly selected copies of a locus with minor allele frequency p are different is $2p(1–p)$, which is maximized when $p = 0.5$.

2. Methods

2.1. *The Data*

In our approach, several individuals of the species (perhaps even just a single individual) are sequenced using next-generation sequencing technology, until an adequate number of genomic differences has been identified. This step is not as easy as it sounds, because we assume that at the start of this process we know nothing about the genomic characteristics of the target species. In contrast, next-generation sequencing instruments and the available software for analyzing the data they produce are primarily designed for re-sequencing, i.e., where the goal is to look for small differences from an available "reference" genome sequence. Often, and also in our case, the main quest is for single-nucleotide polymorphisms, abbreviated SNPs. (Experts might cringe at this use of "SNP", instead requiring that a nucleotide difference be observed at a certain frequency, say $\geq 1\%$, before deserving the title of a polymorphism.) We call this initial stage for our approach "identifying SNPs without a reference". In more detail, the problem is to start with a specification for the desired number of SNPs, and automatically process short fragments of unannotated sequence data, identifying putative SNPs until enough have been found. We have developed software to solve this problem, but that is not the focus of this report. The outcome of this stage is a list of

genomic positions, each identified by, say, the 50 nucleotides on each side, where we predict that two distinct alleles are present in the population.

Once enough SNPs have been predicted, we design a custom genotyping array — a fabricated device that uses hybridization of DNA samples to the sequences flanking the putative sequence difference to determine which variant of each SNP is possessed by each assayed individual. For now, let us assume that each animal has two copies (maternal and paternal) of that genomic position; they can be the same (in which case we say the individual is homozygous for the SNP) or different (heterozygous). For each SNP, arbitrarily pick one of the two variants as the "reference allele", and suppose there are L variant nucleotide positions and that K individuals have been genotyped. The outcome of the genotyping step is an array, $\boldsymbol{A}$, with L rows and K columns, whose value A_{ij} at the intersection of row i and column j is the number of copies (0-2) of the i^{th} SNP's reference allele that was observed in the j^{th} individual. (Be warned that in other contexts it may be common for the roles of rows and columns to be reversed.)

There are many ways to formulate the problem of using genotyping data to select an optimal founder population for a captive-breeding program. We next describe a few of the possibilities.

2.2. *Formulation 1: Approximating Specified Allele Frequencies using Mixed-Integer-Programming Tools*

Suppose that from a pool of K genotyped individuals we want to select N (a specified size of the founder population) whose combined allele frequencies are as close as possible to a given ideal, e.g., the species' allele frequencies in the year 1900 as estimated from museum specimens. Let L denote the number of SNP positions, which we assume are biallelic (two variants), and for each SNP pick one allele arbitrarily for reference. Denote the target frequency of the reference allele for SNP i (where $1 \le i \le L$) by f_i with $0 \le f_i \le 1$. Thus, in the founder population of N animals, there ideally will be $b_i = 2Nf_i$ occurrences of that allele. Selecting a founder population is then equivalent to determining N binary variables $x_j \in \{0,1\}$ for $1 \le j \le K$, where $x_j = 1$ if the j^{th} animal is in the founder population and $x_j = 0$ otherwise. The requirement that there be N founders can be stated as $\sum x_j = N$. The number of occurrences of reference allele i in the founder population is $\sum \{A_{i,j} x_j : 1 \le j \le K\}$, which we want to be very close to the chosen target value b_i.

We will typically want to place additional constraints on the set of selected individuals. For instance, suppose we want to select exactly 50 individuals, of which 20 are males, and 10 are two years old. We form a 3-by-K array, $\boldsymbol{C}$, where first row is all 1s, the second row has a 1 in column j if the j^{th} genotyped individual is a male (0 for a female), and the third row has a 1 in positions corresponding to 2-year-olds (0 otherwise). Our constraints then have the form $\boldsymbol{Cx} = \boldsymbol{d}$, where $\boldsymbol{d} = (50, 20, 10)^{\text{T}}$.

Thus, our focus is on the following general formulation: We are given the $L \times K$ matrix $\boldsymbol{A}$ of allele frequencies $A_{i,j}$, the vector $\boldsymbol{b} = (b_1, b_2, \ldots, b_L)^{\text{T}}$ that indicates the desired total abundance of each reference allele in the founder population, and the $M \times K$ matrix $\boldsymbol{C}$ and vector $\boldsymbol{d} = (d_1, d_2, \ldots, d_M)^{\text{T}}$ that represent other constraints on the population (such as total population size, number of individuals of each age, number of each sex, and so on). We want to determine a vector $\boldsymbol{x} = (x_1, \ldots, x_K)^{\text{T}}$ of binary variables that indicate whether individual j $(j=1,2,\ldots,K)$ is included in the selected population, such that the desired allele abundances are matched as closely as possible subject to the specified constraints. Our problem formulation is thus

$$\text{Minimize}_{\boldsymbol{x}}\ \rho(\boldsymbol{Ax}-\boldsymbol{b}) \text{ subject to } \boldsymbol{Cx} \le \boldsymbol{d},\quad \boldsymbol{x} \in \{0,1\}^{\text{K}},$$

where $\rho(.)$ denotes a loss function that measures goodness of fit between two vectors of length L. (Although the constraints are written as inequalities, our discussion generalizes immediately to the case in which some or all are equality constraints.) The most useful goodness-of-fit measures ρ are the sum-of-squares function, the l_1 (sum-of-absolute-values) loss function, or the l_∞ (maximum residual) loss function, leading to the following three specific formulations:

$$\text{Minimize}_{\boldsymbol{x}}\ (1/2L)\Sigma_i (\boldsymbol{A}_{i\cdot}\,\boldsymbol{x}-b_i)^2 \text{ subject to } \boldsymbol{Cx} \le \boldsymbol{d},\quad \boldsymbol{x} \in \{0,1\}^{\text{K}},$$

(where $\boldsymbol{A}_{i\cdot}$ denotes the i^{th} row of $\boldsymbol{A}$);

$$\text{Minimize}_{\boldsymbol{x}}\ \Sigma_{i=1,2,\ldots,L}\ |\boldsymbol{A}_{i\cdot}\,\boldsymbol{x}-b_i|/L \text{ subject to } \boldsymbol{Cx} \le \boldsymbol{d},\quad \boldsymbol{x} \in \{0,1\}^{\text{K}}$$

and:

$$\text{Minimize}_{\boldsymbol{x}} \ \max_{i=1,2,\ldots L} |\boldsymbol{A}_{i.}\,\boldsymbol{x}-b_i| \text{ subject to } \boldsymbol{Cx} \le \boldsymbol{d}, \ \ \boldsymbol{x} \in \{0,1\}^K$$

The objective in the least-squares formulation can be rewritten as $(1/2)\ \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{c}^T\boldsymbol{x}$ where $\boldsymbol{Q} = (1/L)\boldsymbol{A}^T\boldsymbol{A}$ and $\boldsymbol{c} = -(1/L)\boldsymbol{A}^T\boldsymbol{b}$, leading to a binary quadratic program. The l_1 and l_∞ formulations can be formulated as mixed-integer linear programs (MIP) by using some standard reformulation techniques. For the l_1 formulation, we introduce variables r_i and s_i to denote the positive and negative parts of $\boldsymbol{A}_{i.}\,\boldsymbol{x} - b_i$, respectively, and defining $\boldsymbol{r} = (r_1, r_2, \ldots, r_L)$ and $\boldsymbol{s} = (s_1, s_2, \ldots, s_L)$, we obtain the following reformulation:

$$\text{Minimize}_{\boldsymbol{x},\boldsymbol{r},\boldsymbol{s}} \ \boldsymbol{1}^T(\boldsymbol{r} + \boldsymbol{s}) \ \text{ subject to } \ \boldsymbol{r} - \boldsymbol{s} = \boldsymbol{Ax} - \boldsymbol{b}, \ \boldsymbol{Cx} \le \boldsymbol{d}, \ \boldsymbol{r} \ge \boldsymbol{0}, \ \boldsymbol{s} \ge \boldsymbol{0}, \ \boldsymbol{x} \in \{0,1\}^K,$$

where $\boldsymbol{1}$ denotes the vector of length L whose elements are all 1. In fact, this is a binary MIP, since the solution contains a combination of continuous variables ($\boldsymbol{r}$ and $\boldsymbol{s}$) and binary variables ($\boldsymbol{x}$). For the l_∞ formulation, we obtain:

$$\text{Minimize}_{\boldsymbol{x},\eta} \ \eta \ \text{ subject to } -\eta\boldsymbol{1} \le \boldsymbol{Ax} - \boldsymbol{b} \le \eta\boldsymbol{1}, \ \boldsymbol{Cx} \le \boldsymbol{d}, \ \boldsymbol{x} \in \{0,1\}^K,$$

where η is a scalar that captures the largest magnitude among elements in the vector $\boldsymbol{Ax} - \boldsymbol{b}$.

Commercial packages for solving MIP include CPLEX (www.ilog.com) and Xpress (www.dashoptimization.com), while CBC in the COIN-OR collection (https://projects.coin-or.org/Cbc) is a good open-source alternative. Many packages (including CPLEX and Xpress) now include support for integer quadratic programs as well as linear MIPs. Modeling languages are also available that interface to these solvers and allow users to specify their problem in an intuitive fashion. These include AMPL (www.ampl.com) and GAMS (www.gams.com). However, the problems in this paper have a simple enough form that modeling languages are not necessary unless one wishes to use the NEOS Solver, a web server for solving optimization problems at no cost (see www-neos.mcs.anl.gov).

MIPs are known to be extremely difficult to solve in general (they are NP-hard). However, for many practical problems, good solutions can be obtained in a reasonable amount of computing time. The algorithms underlying MIP codes are based on branch-and-bound strategies, cutting planes, and various other heuristics. When applied to the binary MIP above, branch-and-bound strategies construct a tree of relaxed problems, where in each node of the tree some of the binary variables x_j are fixed at 0 or 1 while the others are allowed to take on any value in the range [0,1]. This "relaxation" is a (continuous) linear program whose optimal value yields a lower bound on the optimal value of the original MIP. At each node of the tree, we form two child nodes by choosing one of the relaxed variables x_j and fixing it to 0 and 1, respectively. Note that the lower bound achieved at each child node must be at least as great as the lower bound at the current node. (At the root node of the tree, we relax all the variables and impose only the bounds $x_j \in [0,1]$, $j=1,2,\ldots,K$, along with the constraints $\boldsymbol{Cx} \le \boldsymbol{d}$.) The full tree would have 2^K nodes in total, but the hope is that we can avoid examining the vast majority of the tree by cutting off branches on which the lower bound is already greater than the value obtained at the "incumbent" – the feasible point with the best known objective value obtained to date.

Cutting planes are additional linear constraints added to the relaxed problem whose function is to exclude fractional solutions – those in which some components x_j are not either 0 or 1. Cuts can also be applied at lower nodes in the branch-and-bound tree. MIP software typically contains many heuristics for deciding on branching strategy (i.e. choosing the relaxed variable x_j on which to branch at each node), deciding what kinds of cutting planes to find and how often to look for them, and looking for candidate feasible solutions to use as incumbents. By setting options at input, users can control all these aspects of the codes.

The problems arising in this application have a property that makes them quite difficult to solve with standard MIP strategies. Because of the nature of the problem data, relaxation of the binary variables leads to a relaxed solution that has most of its components in the interior of the interval [0,1]. (This is true at most of the nodes of the branch-and-bound tree as well as at the root node.) Hence, the relaxed solutions are far from being feasible points for the original problem, and they produce only a weak lower bound on

the true objective. The upshot is that the branch-and-bound strategy is not able to exclude large parts of the tree from the search, and many nodes must be visited (and the relaxed problem at those nodes solved) before the lower bound increases to the point where a candidate solution can be declared to be nearly optimal. (This effect has also been observed in related contexts by D. Bienstock, in a personal communication.) It is likely that the heuristics for finding good candidate solutions in these codes do in fact generate near-optimal solutions after a fairly short time; the difficulty comes in verifying that indeed it is close to the best attainable. Generation of better lower bounds remains an open research question in computational mixed-integer linear programming.

When we use the mixed-integer quadratic programming formulation arising from the sum-of-squares loss function (see below), the same issue arises. However, in this case, in an unpublished manuscript dated 2009, D. Bienstock proposes to use lower bounds on the curvature of the quadratic objective to improve the quality of the lower bounds obtained at the nodes of the branch-and-bound tree. In our case, the matrix $\boldsymbol{A}$ (and $\boldsymbol{Q} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$) typically is well conditioned, so the bounds obtained by these means may be significantly stronger than the standard lower bound. We note that software frameworks for mixed-integer quadratic and nonlinear programs are much less prevalent than for linear MIPs, and this fact together with the large values of K and L for some data sets makes the logistics of developing a code for solving this problem quite daunting.

2.3. *Formulation 3: A Pure Integer-Programming Variation*

We have also explored simpler optimization problems related to selecting a set of individuals that satisfy requirements on their genotypes and other constraints. The hope is that the problem can be solved more efficiently than the problems discussed in the previous section, and that at least in some instances the solution will be adequate for the needs at hand.

For instance, formulating the computation as a linear programming problem whose only unknowns are the K binary variables x_{j} widens the domain of solvers that can be applied. One approach is as follows. Let $\boldsymbol{A}$, $\boldsymbol{b}$, $\boldsymbol{C}$ and $\boldsymbol{d}$ be as in Formulation 1. There, $\boldsymbol{A}$ and $\boldsymbol{b}$ have L rows and contribute to the objective function, while $\boldsymbol{C}$ and $\boldsymbol{d}$ have M rows and constitute the constraints. Let us assume that the constraints require that either $\boldsymbol{C}_{\mathrm{k}\cdot}\,\boldsymbol{x} = \boldsymbol{d}_{\mathrm{k}}$ or $\boldsymbol{C}_{\mathrm{k}\cdot}\,\boldsymbol{x} \geq \boldsymbol{d}_{\mathrm{k}}$ for $1 \leq k \leq M$ (where $\boldsymbol{C}_{\mathrm{k}\cdot}$ denotes the k^{th} row of $\boldsymbol{C}$). Form the $(2L+M-1) \times K$ array $\boldsymbol{A_2}$ and vector $\boldsymbol{b_2}$ as follows. For every row of $\boldsymbol{A}$, corresponding to the reference allele for a particular SNP, $\boldsymbol{A_2}$ contains that row plus a row for the other allele of that SNP. Thus the sum of the two rows has a 2 in every position. The corresponding two entries in $\boldsymbol{b_2}$ are the SNP's entry in $\boldsymbol{b}$ (giving the target abundance for the reference allele) and $2N$ minus that value. The other rows of $\boldsymbol{A_2}$ and $\boldsymbol{b_2}$ are taken from $\boldsymbol{C}$ and $\boldsymbol{d}$, omitting the constraint that the founder-population size is $\boldsymbol{N}$. The pure integer-programming approximation to Formulation 1 is:

$$\text{Minimize } \textstyle\sum x_{\mathrm{j}} \text{ subject to } \boldsymbol{A_2 x} \geq \boldsymbol{b_2}, \quad \boldsymbol{x} \in \{0,1\}^{\mathrm{K}}$$

This reformulation makes the following changes. First, the size of the founder population is now variable. For instance, if each of 5 age groups is required to have 10 representatives among the founders, it might require 55 founders to also satisfy the other constraints. Also, where we formerly required the abundance of the reference allele for the i^{th} SNP to approximate b_{i} (and by inference the abundance of the SNP's other allele to approximate $2N–b_{\mathrm{i}}$), we now require the two alleles' abundances to equal or exceed those values. Finally, equality constraints in Formulation 1 are relaxed to inequalities. For instance, an original requirement of exactly 20 males now requires at least 20 males. Compared to the l_1 version of Formulation 1, this gives a problem with many fewer unknowns (but the same number of binary unknowns), though there are far more constraints. In this respect it is similar to the l_∞ version of Formulation 1.

An advantage of looking at the problem this way is that several approaches for cutting corners are suggested. First, it may be possible in practice to simply discard most of the constraints. For instance, if the only concern is to retain rare alleles, then constraints where the required allele frequency exceeds an appropriate lower bound could be removed. Another strategy is to replace the function being optimized (the number of founders) by a constraint that the founder population be "small enough". For instance, if we want to require that there be at least 10 individuals in each of five age groups (requiring 50 founders), we can add the constraint $\sum x_{\mathrm{j}} \leq 55$. Strategies like these can substantially expand the range of problem that are readily solved, as illustrated in Section 3.3, below.

2.4. *Formulation 4: Selecting Animals That Aren't Genotyped*

Another variant is to again fix the number of animals to be chosen, say N, and optimally select from the already genotyped pool *and/or ungenotyped wild individuals*. For ungenotyped animals, we assume that the genetic profiles of animals from different populations or geographic regions are already characterized from the genotyping. The point is not that individuals who are not genotyped are being selected. Rather, the issue is that representative individuals are being sampled from populations whose general properties are assumed to be known, even if the specific individuals are unknown.

Suppose we have L SNPs characterized for K populations (or geographic locations). Fix a "reference" allele for each SNP (say, the more-common allele), and let $p_{i,j}$ denote that allele's frequency for SNP i in population j, where $1 \le i \le L$ and $1 \le j \le K$. We assume that $p_{i,j}$ is known. If we want to sample x_j animals from population j, an optimal sampling strategy that yields the desired genetic diversity can be determined by the following criteria:

- The expected abundance of the reference allele for the i^{th} SNP closely approximates the target value b_i.
- If population j has c_j individuals, then $0 \le x_j \le c_j$.
- The total sample size is N.

Let $\boldsymbol{P} = \{p_{i,j}\}$ denote the $L \times K$ matrix of allele frequencies, $\boldsymbol{p}_i$ denote the i^{th} row of $\boldsymbol{P}$, and $\boldsymbol{b} = (b_1, b_2, \ldots, b_L)^T$. Also let $\boldsymbol{x} = (x_1, \cdots, x_K)^T$ denote the vector of unknown animal-counts from the various sub-populations. For SNP i, the expected count for the reference allele is $2\boldsymbol{p}_i\boldsymbol{x}$. Thus, the above criteria can be expressed as:

$$\text{Minimize}_{\boldsymbol{x}}\ \rho(\boldsymbol{Px}-\boldsymbol{b}) \text{ subject to } 0 \le x_j \le c_j \text{ and } x_j \text{ an integer for } 1 \le j \le K \text{ and } \sum x_j = N.$$

Here ρ can be any of the loss functions mentioned above. To include both selection (from genotyped animals) and sampling (from the wild), we can treat each genotyped animal as a "one-individual population", for which $p_{i,j}$ equals 0 (homozygote wild type), 0.5 (heterozygote) or 1 (homozygote mutation), and $c_j = 1$. In cases where the sub-population sizes c_j are fairly large, the restriction to integer solutions might not be so onerous as the constraint to binary unknowns in Formulation 2; rounding the entries of a solution to the continuous-variable problem might be good enough in practice. That is, the problem is solved by allowing each variable x_j to assume arbitrary real values between 0 and c_j, and then the optimal value replaced by the closest integer.

3. Experience

In essence, the genotyping data considered in this paper consists of a matrix with L rows and K columns, where each row corresponds to a genomic position where different nucleotides have been observed within a species, and each column corresponds to an individual animal of that species. Each entry is 0, 1, or 2, depending how many copies of the (arbitrarily chosen) reference allele for that difference are present in that individual. (For now, we are considering autosomal nucleotide polymorphisms, but in Section 3.4 we sketch what needs to be changed for other kinds of genomic polymorphisms, such as microsatellites.) Small numbers of these so-called SNPs (single nucleotide polymorphisms) have been used for various purposes related to wildlife management[8], but we anticipate the day, not long off, when next-generation sequencing and large-scale genotyping methods will be employed. In particular, our goal here is to use such data to select a set of animals with optimal genetic diversity that meets certain additional constraints, and we have described combinatorial optimization methods for several formulations. We now recount our experience to date with applying some of those methods to realistic (though mostly artificial) sets of data.

3.1. *Test Data*

Large genotype data sets will soon be available for endangered species, including the orangutan (personal communication from C. Bustamante and D. Locke). Currently, data for 384 SNPs and 322 individuals is available for a bird, the collard flycatcher[9]. Much larger data sets are available for some domesticated species. For instance, for cattle there is data for 37,470 SNPs and 497 individuals[10]. However, the most

extensive data sets are for humans, and we have employed these data to begin evaluating our proposed methods.

We used human genotypes from a preliminary data release from the 1000 Genomes Project[11,12], which we downloaded from ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/release/2009_04/**.** We identified over 4 million SNPs having data from each of the population groups CEU, JPTCHB, and YRI, which yielded complete data for 172 individuals. To experiment with various solution techniques we created data sets by randomly selecting L rows and K columns from the derived array of genotypes, for various values of L and for $K \leq 172$. In our tests, and we anticipate also in practice, L (the number of SNPs) is typically larger than K (the number of genotyped animals). When genders and ages of the individuals were needed, we randomly assigned genders and age groups 1 to 5 with equal probabilities. When a vector of target allele frequencies was needed, we scaled the observed frequencies in the selected genotypes to the desired size of the founder population. That is, for each row (SNP), we summed the numbers (0-2) in that row, multiplied by the size of the founder population, and divided by the number of columns (genotyped individuals).

3.2. *Results Obtained with Mixed Integer Programming Formulations*

For the problem of fitting a desired distribution of allele frequencies, we first wanted to see how well one can do using freely available software, without any attempt to tune the software's behavior to the particularities of our data, and making only modest use of computational resources. To do this, we attempted to solve instances of the l_1 version of Formulation 1 using a MIP solver called CBC run at the NEOS[13] server (without an "Options File" to adjust CBC's behavior). For each of several combinations of L and K, we requested a founder population with 20 males and 10 animals of each age (1 to 5), for a total of 50 founders. For 200 SNPs and 100 animals, the problem was solved to optimality in just over 3 minutes. When the number of SNPs was raised from 200 to 300, the time needed to find an optimal solution shot up to almost an hour. For 500 SNPs and 100 animals, or 300 SNPs and 172 animals, the computation exceeded our 5000-second time limit, producing a solution that is probably reasonable, but quite possibly not optimality.

We next report on results obtained with the commercial MIP solver CPLEX on the l_1 and l_∞ formulations, using a data set with 5000 SNPs and 172 individuals. Again, it was required to select 10 individuals of each age 1 through 5, of whom 20 would be males, for a total of six constraints. The objective was to make the frequency of alleles in the selected population as close as possible to the frequency in the overall population, in the l_1 or l_∞ sense. The modeling language GAMS[14] (www.gams.com) was used to define the model. (Direct use of GAMS models has the advantages that they allow variants of the models to be tried quickly, and that GAMS files can be submitted to the NEOS server.)

CPLEX, like other MIP solvers, allows many options and parameters to be set to non-default values, to improve their efficiency. Insight into the nature of problems being solved, and some trial-and-error and parameter tuning, can lead to dramatic improvements in run time over default values. For these problems, however, we able to obtain only modest improvements over default behavior by choosing alternative values of these parameters. Techniques that can handle the special characteristics of these problems (noted above) and produce verified optimal solutions in a reasonable amount of time are not available in current MIP solvers and are a topic of ongoing research. We believe, however, that the code is finding close-to-optimal solutions in reasonable time. As noted above, the difficulty comes only in verifying that they are indeed near-optimal, as we need to examine a large fraction of the nodes on the search tree to increase the lower bound to a level close to the best feasible solution obtained so far.

We mention for the record that the following non-default parameter values were used in the CPLEX runs reported here: cliques=–1 and covers=–1 (to turn off certain kinds of cut generation), dpriind=3 (to use steepest-edge pricing in slack space of the simplex method used at each node), varsel=4 (to force the use of pseudo-reduced costs as a criterion for branching), symmetry=5 (to generate symmetry-breaking cuts aggressively during the early stages of the solution process), mipemphasis=4 (to emphasize the search for better candidate solutions rather than improvement of the lower bound). We make no claims that this combination gives the best performance overall, or even that it is better than the default settings on these problems. CPLEX Version 11.2 was executed on both formulations for 15,000 CPU seconds (a limit prescribed by us) on a PC with an Intel Xeon quadcore CPU at 2.66 GHz, running Red Hat Enterprise Linux Server release 5.3, with 4GB of DDR2 memory at 800 MHz.

For the l_∞ formulation, the code found a point with objective value 9.9770, with a lower bound of 1.3901, within a few hundred seconds of CPU time. The candidate solution was found by running default

heuristics and adding cuts to the relaxed problem at the root node of the search tree. During the remainder of the 15,000 seconds of execution time, 1460 nodes of the search tree were examined and four progressively better candidate solutions were found. At termination, the best solution found had objective 9.4070, while the lower bound had increased only to 1.8060.

For the l_1 formulation, the code quickly identified a candidate solution with objective 2.2173, with a lower bound of .3931, again without performing any branching. During the remainder of the execution time, three more candidate solutions with progressively better objective values were identified, with the final solution having objective 2.2150, while the lower bound had increased to 0.4717. Only 201 nodes of the branch-and-bound search tree were evaluated.

We note that the solutions obtained with these two objectives were quite different. In fact, of the 50 individuals in each solution, only 19 were selected in both.

We conclude by reporting results obtained with the binary quadratic programming formulation. Note that the only variables in this formulation are the original 172 binary variables. (By forming the Hessian matrix $\boldsymbol{Q}$ and linear term $\boldsymbol{c}$ explicitly, we avoid the need to introduce any additional continuous variables.) We found that the results were improved slightly when we performed a simple transformation of $\boldsymbol{A}$ and $\boldsymbol{b}$, before forming $\boldsymbol{Q}$ and $\boldsymbol{c}$. Since a total of 50 animals are to be selected (that is, 50 of the components of $\boldsymbol{x}$ are 1 at the optimum, while the remainder are zero), we can subtract 1 from all elements of $\boldsymbol{A}$, while subtracting 50 from each element of $\boldsymbol{b}$. After this transformation, $\boldsymbol{A}$ contains elements –1, 0, and +1, and $\boldsymbol{Q}=(1/L)\boldsymbol{A}^T\boldsymbol{A}$ can have slightly better conditioning. We use this shifting procedure for the experiment reported below.

For this formulation, we wrote a C code to set up the problem and call CPLEX. Except for the termination criteria (which enforced a limit of 15,000 seconds on CPU time), default parameter settings for CPLEX's MIQP solver were used. Within a few seconds, the code finds an incumbent (the best feasible point encountered to date) with objective 3.9701. A total of 22 incumbents are found during the run. The last of these (which is the solution reported by the code) has an objective of 3.8113. After about a minute, the lower bound is approximately 1.71; it increases steadily but slowly thereafter to a final value of 1.8878. About 13,500,000 nodes of the branch-and-bound tree are examined during the run. CPLEX's mixed-integer QP solver makes much less use of cuts, and thus places much more reliance on branching as a means of solving the problem.

To compare the optimization methodology with a random strategy as a means for selecting a set of individuals that fits the target allele frequencies while satisfying the constraints, we programmed in MATLAB a code that generates 3414 feasible points for this test set at random. The mean of the objective values obtained from this process is 6.2729, with a standard deviation of 1.1798. The best objective found was 4.8353, which is significantly worse than the solution obtained by the optimization code after just a few seconds of run time.

3.3. *The Pure Integer-Programming Variation*

For a given set of genotyping data, finding a "sweet spot" for the many combinations of potential LP shortcuts, available solvers, and choices of solver options will require some effort. As might be expected of a computational problem with exponential time complexity, seemingly minor changes in the approach often mean the difference between a quick solution and failure of the computation to terminate in a reasonable time. However, we now report one simple experiment that hints at what can be achieved.

In Section 3.2 we noted failure of a straightforward attempt to run a NEOS MIP solver on random data for 300 SNPs and 172 animals. To experiment with reformulations of the problem, we generated random data for 2000 SNPs and 172 animals and applied the following shortcuts to the approach outlined in Section 2.3. First, we permitted up to 55 founders (while continuing to require at least 10 in each of the five age groups). Second, we retained constraints on the allele abundance only when the lower bound was 5 or less, leaving 146 of the original 4000 allele-abundance constraints. Using C-language programs that we wrote to convert genotyping data to MPS format, we submitted the data to the SCIP[15] optimizer at the NEOS server (with no "Parameter file"). In around 30 seconds, SCIP found a feasible solution containing 55 founders. We discovered that 16 of the original alleles whose constraints had been removed had less than 5 occurrences in the computed founder-set. All of those had at least 3 occurrences, which might be considered adequate under certain conditions. An alternative is to add the requirement that each of those 16 alleles occur at least 5 times to the 146 allele-abundance constraints, and re-solve the problem.

We also experimented with an even simpler integer-programming formulation, using a real-world data set. The objective was simply to guarantee a minimum number of occurrences of each allele, using genotyping data from a 96-SNP array that we designed for the Tasmanian devil (*Sarcophilus harrisii*), a species that is gravely threatened by extinction from a transmissible tumor[16]. We used the array to genotype 85 animals, and for the following analysis used the 69 SNPs that appeared in more than one animal. The problem we investigated was to select a set of animals that contains each allele at least twice, which corresponds to a binary linear programming problem of minimizing Σx_j subject to $\boldsymbol{Ax} \geq \boldsymbol{2}$, $\boldsymbol{x} \in \{0,1\}^K$, where $\boldsymbol{2}$ denotes the vector containing 2s. For our 69-by-85 array, $\boldsymbol{A}$, linear programming selected four animals. Interestingly, a simple greedy algorithm[17] that repeatedly picks the animal that adds the most "uncovered" alleles, selected six animals. With four randomly selected animals, an average of only 22.7 of the 69 alleles appeared at least twice. Looked at another way, random selection picked an average of 31.9 animals before reaching 2-fold representation of every allele. What makes the problem difficult for random sampling is that a few of the alleles appeared only a few times among the 85 animals. (Extensive comparisons of the greedy and random selection strategies for a very similar problem have been published by others[18].)

3.4. *Handling Other Types of Polymorphisms*

Our approach has been described under the assumption that each genetic marker has two possible states, and that each individual has two copies. Minor adjustments are required for other cases, such as microsatellites with multiple states, or sex-chromosome or mitochondrial markers where an individual may have fewer than two copies. For instance, for a microsatellite with three observed lengths, we could devote three rows of the genotypes array, one for each length. For each column (individual) the values in those rows will each be 0, 1, or 2, and the sum of the three values will be 2 (except perhaps for microsatellites on a sex chromosome). This works for both the linear-programming and the quadratic-programming formulations.

4. Conclusion

We have developed and evaluated several methods that can use high-throughput genotyping data to select a set of animals that best approximates a specified allele profile, subject to additional constraints; the selected animals might be used as founders for a captive breeding program or reintroduced into a former range for that species, for example. We anticipate that the costs of sequencing and genotyping will continue to plummet, making it feasible to gather genome-wide population-genetic data from hundreds of animals from each of many species. One can easily imagine having genotypes for many thousand SNPs in hundreds of individuals from an endangered species, and using that information to select founders. The number of SNPs required to reliably represent genome-wide variation is liable to depend heavily on the particular species (not to mention how one interprets "reliably represent"). For humans it has been estimated[19] that, among all of the 6 million common (minor-allele frequency ≥ 0.1) European SNPs in the human genome, about 80% could be ascertained at $\rho^2 > 0.8$ through pairwise linkage disequilibrium (LD) by genotyping 0.8 million European common and non-redundant variants in the database dbSNP. To cover a similar percentage of common SNPs in African-Americans, the number increases to 1.1 million SNPs. For a highly bottlenecked species with more extensive LD, the required number of SNPs could be much less.

In any case, even with the available technologies, affordable approaches are possible. For a few tens of thousands of dollars it is possible to sequence transcripts (cDNA) from several individuals, from which polymorphic amino-acid positions can be identified. If the number of those differences exceeds the desired array capacity, computational analysis could reduce them to the differences that show signs of being functionally important[20]. For an additional few tens of thousands of dollars, a custom genotyping array can be purchased and applied to genotype those differences on several hundred animals. Techniques described in this paper, such as the mixed-integer-programming technique, can then perform an optimal, or at least near-optimal, selection of founders, in the sense of capturing desired abundances of the putatively functional protein variants.

Departure of the target allele distribution from the distribution in the overall population increases the value of using an approach like ours, compared to simple random selection. For instance, we might want to avoid hybridized alleles (e.g., cattle genes in American bison). More generally, the belief that the goal of conservation efforts is to restore ecosystems to their "natural state", which to some writers means their state before any human intervention, has been widely discussed[21]. Several publications[22,23] have used museum

specimens to determine earlier genetic profiles. However, this has typically been done using only tiny amounts of hyper-variable sequence data, such as 500 bp from the mitochondrial control region or a small number of microsatellites. Another scenario with differing allele profiles might be importing animals to bring a struggling population up to a viable size when we want to retain its unique genetic make-up and/or reduce the threat of outbreeding depression.

Other decisions related to species conservation have been approached with computational methods. One such problem is selecting a set of species to be preserved. The problem has been formalized as optimally selecting a specified number of species, given a phylogenetic tree relating a more inclusive set of species and with branch lengths that measure inter-species differences. A natural objective is to maximize the sum of branch lengths in the induced subtree[24]. A simple greedy algorithm guarantees an optimal solution[25], and dynamic-programming algorithms solve several generalizations[26.27].

A set of individuals within a species is less well represented by a weighted tree, and other ideas have been applied to selecting intervals, with "maximum diversity" a frequent goal. For instance, given a set of genetic markers (SNPs, microsatellites, allozymes, etc.) one could try to maximize the number of observed alleles represented in the selected set of individuals[28]. This is essentially just the infamous Minimum Set Cover problem[29], which is NP-complete and hence solved in biodiversity practice[17] and studied in computer science theory[30] using approximation methods (including linear programming). The problem studied here, with the added complexity of an arbitrary target distribution and constraints on gender and age, is correspondingly more challenging.

A somewhat different class of optimization problems arises for management of a captive breeding program, where in addition to genotypes one has access to genealogies. As with selection of founders, there is tension between the potential goals of maximizing diversity and maintaining allele frequencies[31]. Combinatorial optimization methods have been proposed for designing breeding programs[32,33].

To keep pace with plummeting costs for sequencing and genotyping, more work is needed to extend the computational approaches described in this paper. Cutting-edge research in optimization algorithms that exploits the particular characteristics of this family of optimization problems may be needed to best utilize cutting-edge data-producing technologies.

Acknowledgments

We thank Belinda Giardine for preparing the 1000-Genomes data set. Oleg Shylo and the reviewers provided a number of helpful suggestions. W.M. and Y.Z. are supported by grant HG-002238 from the National Human Genome Research Institute. S.J.W. has support from NSF grants DMS-0427689, CCF-0430504, DMS-0914524, and DOE grant DE-FG02-04ER25627. S.C.S. is supported by the Gordon and Betty Moore Foundatation, and V.M.H. is a Cancer Institute of New South Wales Fellow.

References

1. R. Frankham, J. D. Ballou and D. A. Briscoe, *Introduction to Conservation Genetics,* Cambridge University Press (2002).
2. J. Höglund, *Evolutionary Conservation Genetics,* Oxford University Press (2009).
3. G. Bertorelle, M.W. Bruford, H. C. Hauffe, A. Rizzoli and C. Vernise, *Population Genetics for Animal Conservation*, Oxford University Press (2009).
4. D. R. Marshall and A. D. H. Brown, in *Crop Genetics Resources for Today and Tomorrow* (O. H. Frankel and J. G. Hawkes, eds.), Cambridge University Press, 53-80 (1975).
5. J. K. Pritchard, M. Stephens and P. Donnelly, *Genetics* **155**, 945-959 (2000).
6. W. Miller, D. I. Drautz, A. Ratan, et al., *Nature* **456**, 387-390 (2008).
7. W. Miller, D. I. Drautz, J. Janecka. et al., *Genome Research* **19**, 213-220 (2009).
8. J. Slate, J.Gratten, D. Beraldi, et al., *Genetica* **136**, 97-107 (2009).
9. N. Backström, N. Karaiskou, E. H. Leder et al., *Genetics* **179**, 1479-1495 (2008).
10. The Bovine HapMap Consortium, *Science* **324**, 528-532 (2009).
11. B. M. Kuehn, *JAMA* **300**, 2715 (2008).
12. N. Siva, *Nat. Biotech.* **26**, 256 (2008).
13. J. Czyzyk, M. Mesnier and J. J. Moré, *IEEE Comp. Sci. Eng.* **5**, 68-75 (1998).
14. A. Brooke, D. Kendrick and A. Meeraus, *GAMS: A User's Guide*, The Scientific Press (1988).

15. T. Achterberg, *Constraint Integer Programming*, Ph. D. dissertation, TU Berlin (2007). Available at http://scip.zib.de/documentation.shtml.
16. H. V. Siddle, A. Kreiss, M. D. Eldridge et al., *Proc. Natl. Acad. Sci. USA* **104**, 16221-16226 (2007).
17. B. Gouesnard, T. M. Bataillon, G. Decoux et al., *J. Hered.* **92**, 93-94 (2001).
18. H. I. McKhann, C. Camilleri, A. Bérard, et al., *Plant J.* **38**, 193-202 (2004).
19. C. S. Carlson, M. A. Eberly, M. J. Rieder et al., *Nat. Genet.* **33**, 518-521 (2003).
20. P. Ng, S. Levy, J. Huang et al., *PLoS Genetics* **4**, e1000160. doi:10.1371/journal.pgen.1000160 (2008).
21. P. I. Angermeier, *Conserv. Biol.***14**, 373-381 (2000).
22. H. C. Rosenbaum, M. G. Egan, P. J. Clapham et al., *Conserv. Biol.* **14**, 1837-1842 (2000).
23. A. Koblmüller, M. Nord, R. K. Wayne and J. A. Leonard, *Mol. Ecol.* **18**, 2313-2326 (2009).
24. D. P. Faith, *Biol. Conserv.* **61**, 1-10 (1992).
25. M. Steel, *Syst. Biol.* **54**, 527-529 (2005).
26. F. Pardi and N. Goldman, *Syst. Biol.* **56**, 431-444 (2007).
27. B. Q. Minh, F. Pardi, S. Klaere and A. von Haeseler, *IEEE/ACM Trans.Comp, Biol. Bioinf.* **6**, 22-29 (2009).
28. D. J. Schoen and A. H. D. Brown, *Proc. Natl. Acad. Sci. USA* **90**, 10623-10627 (1993).
29. R. Karp, *Complexity of Computer Computations* (R. E. Miller and J. W. Thatcher, eds), Plenum Press, 85-103 (1972).
30. D. Hochbaum, *Approximation Algorithms for NP-Hard Problems.* PWS Publishing, Boston (1996).
31. M. Saura, A. Pérez-Figueroa, J. Fernándes et al., *Conserv. Biol.* **22**, 1277-1287 (2008).
32. J. Vales-Alonso, J. Fernándes, F. J. Gonzáles-Castaño and A. Caballero, *Math. Biosci* **183**, 161-173 (2003).
33. T. Y. Berger-Wolf, C. Moore and J. Saia, *J. Theor. Biol.* **244**, 433-439 (2007).

COMPUTATIONAL STUDIES OF NON-CODING RNAS

ROLF BACKOFEN
Institute of Computer Science, Albert-Ludwigs-University Freiburg, Germany
backofen@informatik.uni-freiburg.de

HAMIDREZA CHITSAZ
School of Computing Science, Simon Fraser University, Canada
hrc4@cs.sfu.ca

IVO HOFACKER
Institute for Theoretical Chemistry, University of Vienna, Austria
ivo@tbi.univie.ac.at

S. CENK SAHINALP
School of Computing Science, Simon Fraser University, Canada
cenk@cs.sfu.ca

PETER F. STADLER
University of Leipzig, Germany
studla@bioinf.uni-leipzig.de

1. Introduction

Until recently, RNA has been viewed as a simple "working copy" of the genomic DNA, simply transporting information from the genome into the proteins. In the 1980s, this picture changed, to certain extent, with the discovery of ribozymes and the realization that the ribosome is essentially an "RNA machine". Since the turn of the millenium, however, RNA has moved from a fringe topic to a central research topic following the discovery of RNA interference (RNAi), the post transcriptional silencing of gene expression via interactions between mRNAs and their regulatory RNAs.

More recent studies[1,2] have revealed that a large fraction of the genome sequences give rise to RNA transcripts that do not code for proteins. Those RNAs that do not code for proteins are called non-coding RNAs (ncRNAs).

A recent computational screen estimated the number of small regulatory RNAs, which form an important class of non-coding RNAs, in *Arabidopsis thaliana* to be in the order of 75,000.[3] Among small RNAs, two subclasses form the bulk of all regulatory RNAs: microRNAs (miRNAs) and small interfering RNAs (siRNAs) — which are of similar length (21 to 25 nt) and composition but different by origin. It is predicted that these two subclasses regulate at least one-third of all human genes. There are many other classes of non-coding RNAs with functionalities beyond simple regulation of gene expression: examples include snoRNAs, snRNAs, gRNAs, and stRNAs, which respectively perform ribosomal RNA (rRNA) modification, RNA editing, mRNA splicing and developmental regulation.[4] Even for these well-studied RNAs, their precise mode of function remains poorly understood.

In addition to such endogenous ncRNAs, antisense oligonucleotides have been used as exogenous inhibitors of gene expression; antisense technology is now commonly used for therapeutic purposes and as a research tool. The therapeutic objective of antisense technology is to block the production of disease-causing proteins. In principle, these artificial regulatory RNA molecules could be employed as drugs for the treatment of a variety of human diseases including various types of cancer, rheumatoid arthritis, brain diseases, and viral infections.[5] As a research tool, antisense nucleic acids may be used to study metabolic networks by controling or interfering with the dynamics and function of various modules in the network. Furthermore,

synthetic nucleic acid systems have been engineered to self-assemble into complex structures performing various dynamic mechanical motions.[6] Despite advances in computational studies of non-coding RNA, there are still many open areas and unresolved issues particularly for high-throughput applications based on the new genome sequencing technologies.

The main objective of this session is to discuss new algorithms, software tools and their applications in non-coding RNA bioinformatics. In particular, the papers in this session exemplify recent progress in computational methods that help non-coding RNA sequence prediction and identification, structure prediction and determination, and function determination. Specific problems in computational studies of ncRNAs include:

- Algorithms for modelling interactions between RNAs and other molecules, particularly RNA-RNA and RNA-protein interactions
- Learning thermodynamic parameters that are involved in the prediction of secondary and tertiary structure of non-coding RNAs
- Novel approaches to single or joint RNA structure prediction and determination
- Algorithms for exploring RNA folding pathways and kinetic traps on the energy landscape
- Alignment and comparative analysis of multiple non-coding RNA sequences
- RNA evolution
- Functional classification of non-coding RNAs
- Modelling classes of single or joint non-coding RNAs through stochastic context free grammars and their variations
- Tools that detect structural motifs in a genome sequence, especially those that could be potentially involved in the regulation of target mRNAs
- Combinatorial and heuristic tools for de novo non-coding RNA identification in a genome sequence
- Efficient algorithms for searching RNAs in a data collection with specific sequence and structural motifs.

2. Session papers

Recent improvements in sequencing methods have introduced high-throughput, low-cost, and cloning-free (thus less labor-intensive) technologies. The revolution in DNA sequencing will shortly result in an enormous collection of sequence data pertaining to the genomes and transcriptomes of various human individuals from different populations and also various species. Several papers in this session try to address the increased demand for this type of data analysis.

RNA secondary structure and folding kinetics have always been a central research topic in computational studies of RNA. In the first paper of this session, Thachuk *et al.* make two new contributions to the problem of calculating pseudoknot-free folding pathways with minimum energy barrier between pairs (A, B) of RNA secondary structures. Their first contribution is an exact algorithm to find a minimum barrier direct folding pathway for a simple energy model in which each base pair contributes equally to the structures stability. In a direct (minimum length) folding pathway, intermediate structures contain only base pairs in A and B and are of length $|A| + |B|$. The problem is NP-hard, therefore their algorithm requires exponential time in the worst case. Their second contribution proves that for the simple energy model, repeatedly adding or removing a base pair from A or B along a pathway does not lower the energy barrier.

Dotu *et al.* describe dynamic programming segmentation algorithms to segment RNA secondary and tertiary structures into distinct domains in the second paper. A possible application is to determine the boundaries of predicted ncRNAs. Under the assumption that microRNA precursors are less than 100nt long, their method predicts the precursors embedded in a genomic context of up to 1000nt with an accuracy around 90%. They also compare their algorithm to the manual segmentation of 16S rRNAs reported in the literature.

As a solution to a fundamental problem in computational studies of ncRNAs, `RNAz` has been used for de novo prediction of structured non-coding RNAs in comparative genomics data. In the third paper, `RNAz`

2.0 is presented which improves the previous version in several aspects: It uses a dinucleotide background models to increase accuracy, and an entropy measure to represent sequence similarities. In addition, it has been trained on a larger data set, using either sequence-based or structural alignments. As a result `RNAz` 2.0 has a significantly lower false positive rate than the previous version.

Recent advances in high-throughput sequencing for the first time provide the opportunity to study the entire transcriptome sequences and concentrations. In many cases, a significant part of the transcriptome consists of non-coding RNAs. Some of these ncRNAs are processed by the cell post-transcriptional machinery to yield shorter RNA products. It is believed that these splicing patterns depend on the secondary structure. This leads to specific patterns of short reads that can be detected after mapping the read sequences to the reference genome. In the fourth paper, Langenberger et al. suggest that these read patterns are characteristic for the spliced RNA transcripts. Therefore, Langenberger *et al.* explore the potential of short read sequence data in the classification and identification of non-coding RNAs.

Okada *et al.* report on an improved version of the Structure Conservation Index, used in RNAz. based on centroid estimators rather than minimum free energy structures. Poolsap *et al.* present a dynamic programming approach to compute RNA-RNA interactions for the case where the sequences motifs involved in the binding are known in advance.

Acknowledgements

We thank all the authors who submitted papers to the session, and we gratefully acknowledge the reviewers who contributed their time and expertise to the peer review process.

References

1. The FANTOM Consortium, *Science* **309**, 1159 (2005).
2. ENCODE Project Consortium, *Nature* **447**, 799 (2007).
3. M. W. Vaughn and R. Martienssen, *Science* **309**, 1525 (2005).
4. R. F. Gesteland, T. R. Cech and J. F. Atkins (eds.), *The RNA World*, 3rd edn. (Cold Spring Harbor Laboratory Press, Plainview, NY, 2006).
5. A. Fjose and O. Drivenes, *Birth Defects Res C Embryo Today* **78**, 150 (2006).
6. P. Guo, *J Nanosci Nanotechnol* **5**, 1964 (2005).

RNA STRUCTURAL SEGMENTATION

IVAN DOTÚ

Department of Computer Science, Brown University, Box 1910, Providence, RI 02912, and Department of Biology, Boston College, Chestnut Hill, MA 02467, USA.

WILLIAM A. LORENZ

Department of Biology, Boston College, Chestnut Hill, MA 02467, USA.

PASCAL VAN HENTENRYCK

Department of Computer Science, Brown University, Box 1910, Providence, RI 02912.

PETER CLOTE

Department of Biology, Boston College, Chestnut Hill, MA 02467, USA.
(∗) Corresponding author with email: clote@bc.edu

We describe several dynamic programming segmentation algorithms to segment RNA secondary and tertiary structures into distinct *domains*. For this purpose, we consider fitness functions that variously depend on *(i)* base pairing probabilities in the Boltzmann low energy ensemble of structures, *(ii)* contact maps inferred from 3-dimensional structures, and *(iii)* Voronoi tessellation computed from 3-dimensional structures. Segmentation algorithms include a direct dynamic programming method, previously discovered by Bellman and by Finkelstein and Roytberg, as well as two novel algorithms – a parametric algorithm to compute the optimal segmentation into k classes, for each value k, and an algorithm that simultaneously computes the optimal segmentation of all subsegments.

Since many non-coding RNA gene finders scan the genome by a moving window method, reporting high-scoring windows, we apply structural segmentation to determine the most likely $5'$ and $3'$ boundaries of precursor microRNAs. When tested on all precursor microRNAs of length at most 100 nt from the Rfam database, benchmarking studies indicate that segmentation determines the $5'$ boundary with discrepancy (absolute value of difference between predicted and real boundaries) having mean -0.640 (stdev 15.196) and the $3'$ boundary with discrepancy having mean -0.266 (stdev. 17.415). This yields a sensitivity of 0.911 and positive predictive value of 0.906 for determination of exact boundaries of precursor microRNAs within a window of approximately 900 nt. Additionally, by comparing the manual segmentation of Jaeger et al. with our optimal structural segmentation of 16S and 16S-like rRNA of *E. coli*, rat mitochondria, *Halobacterium volcanii*, and *Chlamydomonas reinhardii* chloroplast into 4 segments, we establish the usefulness of (automated) structural segmentation in decomposing large RNA structures into distinct domains.

Availability: Source code for all algorithms is available at `http://bioinformatics.bc.edu/clotelab/`.

Keywords: non-coding RNA gene finder, segmentation algorithm,secondary structure, tertiary structure, RNA domain

1. Introduction

Several groups, such as Benaola-Galván et al.,[3] Román-Roldán et al.,[16] and Li et al.,[10–12] have developed recursive segmentation algorithms with the goal of segmenting chromosomal regions in order to detect isochores, CpG islands and other broad genomic features. The underlying idea of such divide-and-conquer recursive segmentation algorithms is similar to that of C4.5 decision trees, cf. Quinlan,[15] and depends on repeatedly splitting a segment into left and right halves in order to maximize the Jensen-Shannon divergence[a]

$$JS(L,R) = H(W) - \frac{m}{n}H(L) - \frac{n-m}{n}H(R). \quad (1)$$

[a]See Lin[13] for more on the Jensen-Shannon divergence, Kullback-Liebler distance, etc.

Here L, R are the left and right segments of the whole segment W, the lengths of L, R, W are respectively $m, n-m, n$, and $H(X)$ denotes the Shannon entropy of segment X.[b] Figure 1 depicts the pseudocode corresponding to this approach, where it should be noted that the main weakness of the recursive segmentation method is to determine when to discontinue segmentation. See Clote and Backofen[4] for detailed explanation and full pseudocode of this algorithm.

While this method has been applied to the detection of broad features of chromosomal DNA,[3,10–12,16] other segmentation algorithms in the literature have been introduced by Finkelstein and Roytberg[5] (dynamic programming), and Schmidler et al.[17] (Bayesian a posterior method).[c] Applications of the dynamic programming segmentation algorithm of Finkelstein and Roytberg[5] have been given by Sunyaev et al.[18] for multiple alignments of proteins, while applications of the Bayesian a posterior method have been presented by Schmidler et al.[17] to predict protein secondary structure α-helices and β-sheets given the primary sequence information.

In this paper, we describe a dynamic programming segmentation algorithm, previously discovered by Bellman[2] and by Finkelstein and Roytberg,[5] as well as several novel algorithmic extensions. The segmentation algorithms are applied to segment 3-dimensional RNA structures into *domains*, and use to detect the boundaries of certain non-coding RNA genes within high scoring windows, as determined by many moving-window genome scanning algorithms.

```
1   void segment( int i, int j, double s) {
2      max=0
3      for k=i to j-1{
4         L = w_i ··· w_k
5         R = w_{k+1} ··· w_j
6         if statistical significance of splitting L, R exceeds s then
7            if JS(L, R) > max then
8               max = JS(L, R)
9               x = k
10        }
11     print x
12     segment(i,x,s)
13     segment(x+1,j,s)
14  }
```

Fig. 1. Pseudocode for recursive segmentation algorithm of Román-Roldán et al.[16] Note that one of the difficulties of this approach consists in determining a minimum threshold s, below which segmentation is discontinued.

2. Methods

The problem we consider consists in segmenting a sequence $S = \langle s_1, \ldots, s_n \rangle$ into a number of consecutive subsequences (called segments) $S_1, \ldots, S_k$. (The sequence S is thus the concatenation of $S_1, \ldots, S_k$.) Each segment S_i is associated with a *base fitness function* value $f(S_i, S)$ which only depends on the elements in S_i and those in $S \setminus S_i$, not on the segmentation itself. In this paper, such a function will be expressed in terms of two functions g and h as follows:

$$f(S_i, S) = \frac{\sum_{x,y \in S_i, x \neq y} g(x, y) - \sum_{x \in S_i, y \in S \setminus S_i} h(x, y)}{|S_i|} \tag{2}$$

[b] In general, if X is a sequence in the k-letter alphabet Σ, then $H(X)$ equals $-\sum_{i=1}^{k} p_i \cdot \ln p_i$, where p_i is the relative frequency of the ith letter of Σ. Typical applications of entropy in genomic segmentation consider the 2-letter alphabet $\{R, Y\}$ of purines (A,G) and pyrimidines (C,T), the 2-letter alphabet $\{S, W\}$ of strong (C,G) and weak (A,T) nucleotides, etc.

[c] An anonymous referee kindly pointed out the pertinence of the much earlier paper by R. Bellman.[2]

where $|S_i|$ denotes the number of elements in S_i. This contrasts with previous methods[16–18] where base fitness function $f(S_i, S)$ depends only on S_i, but not on $S \setminus S_i$. Our goal is to find a segmentation $S_1, \ldots, S_k$ that maximizes the sum of the fitness values, i.e.,

$$\sum_{i=1}^{k} f(S_i, S). \tag{3}$$

Observe that the number of segments k is not fixed and is chosen to maximize the overall fitness. In the following, we also use $f_{i,j}$ to denote $f(\langle s_i, \ldots, s_j \rangle, S)$.

2.1. *Dynamic programming using quadratic time and quadratic space*

We now present an $O(n^2)$ algorithm to solve this problem. The key idea underlying the algorithm is to reason about partial segmentations which cover prefixes $(s_1, \ldots, s_k)$ but whose fitness values are computed with respect to the entire sequence S. Obviously, when $k = n$, we obtain a solution to the original problem.

The algorithm is based on a recurrence relation on the starting positions and lengths of the last segment in an optimal (partial) segmentation. More precisely, $V(\ell, x)$ denotes the fitness value of the best partial segmentation whose last segment has length ℓ and starting position x; i.e. $V(\ell, x)$ equals the fitness of $\langle s_1, \ldots, s_{x+\ell-1} \rangle$ when segmented into $S_1, \ldots, S_k$ for arbitrary k, where the rightmost segment $S_k = \langle s_x, \ldots, s_{x+\ell-1}$. The base case corresponds to $x = 1$ and is given by

$$V(\ell, 1) = f_{1,\ell} \tag{4}$$

for $1 \leq \ell \leq n$. The recursive case for $1 < x$ is given by the formula

$$V(\ell, x) = f_{x,x+\ell-1} + \max\{V(i, j) : j + i = x\} \tag{5}$$

The left part of the sum is the fitness value of the last segment. The right part is the fitness value of the best partial segmentation that ends at $x-1$. It is obtained by considering the fitness values of all the partial segmentations of $\langle s_1, ..., s_{x-1} \rangle$. By induction, these fitness values are associated with their last segments, i.e., segments that start at some position j, have some length i, and end at position $x - 1$. The fitness value of the optimal segmentation of S is then given by

$$\max\{V(\ell, x) : \ell + x = n + 1, \ell > 1, x \geq 1\}. \tag{6}$$

Given the entry (ℓ^*, x^*) with maximal fitness value $V(\ell^*, x^*)$, the set of starting positions $st[\ell^*, x^*]$ of the segments in the best segmentation can be traced backwards from using the following recurrence

$$\begin{aligned} st[\ell, x] &= \{x\} \cup st[x - p, p] \\ &\qquad \text{where } p = \max\{p' : V(x - p', p') = \max\{V(i, j) : j + i = s\}\}; \\ st[\ell, 1] &= \emptyset \end{aligned} \tag{7}$$

which, at each step, retrieves the last segment $\langle s_{x-p}, \ldots, s_{x-1} \rangle$ of the optimal partial segmentation.

We now argue that these recurrence relations can be computed by an $O(n^2)$ dynamic programming algorithm. First observe that the expression

$$\max\{V(i, j) : j + i = x\} \tag{8}$$

must only consider $x-1$ segments since $j \geq 1$ and $i \geq 1$, i.e., there are only $O(x)$ pairs to consider. Moreover, observe that this expression does not depend on ℓ in the recurrence relation and hence can be computed once for all entries $V(1, x), \ldots, V(n - x, x)$. As a result, the dynamic programming algorithm runs in $O(n^2)$ provided that the expression is computed once at the beginning of each column. Note also that the index p in the recurrence for st can be computed during the forward computation, so that the backward computation takes only $O(n)$ time.

Note that this algorithm can yield the maximum, minimum, and average fitness of all segments; however, the space required is quadratic. In the next section, we describe a linear space algorithm.

```
1   int[n][n] parametricSegmentation(rna,f,numSegments){
2     /*------------------------------
3      rna is RNA sequence, f is base fitness function.
4     ------------------------------------------------*/
5     n = len(rna); SplitPoints = ∅
6     for d = LOWER to n
7       for num = 2 to min(numSegments, d/LOWER)
8         for m = (num − 1) · LOWER + 1 to d − LOWER − 1
9           val = PF(m, num − 1) + f_{m+1,d}
10          if val > max
11            splitPoint = m
12            max = val
13        PF(d, num) = max
14        SplitPoints[d,num] = splitPoint
15    return SplitPoints  //Using SplitPoints array, one can perform traceback
16    }
```

Fig. 2. Pseudocode for parametric segmentation algorithm to compute optimal *parametric fitness* $PF(d,k)$ over all segmentations of $[1,d]$ into k segments. Note how bounds for minimum segment size (LOWER) and maximum segment size (UPPER) can easily be accommodated within such segmentation algorithms.

2.2. *Dynamic programming using quadratic time and linear space*

Given the complete segment $S = s_1, \ldots, s_n$, let $F(i)$ designate the maximum fitness over all segmentations of $s_1, \ldots, s_i$. Straightforwardly,

$$F(i) = \begin{cases} 0 & \text{if } i = 0 \\ \max(f_{1,i}, \max_{1 \le k < i} F(k) + f_{k+1,i}) & \text{else.} \end{cases} \tag{9}$$

It can be seen how the maximum fitness of S is given by $F(n)$, and by means of tracebacks, we obtain the obtimal segmentation. Computation time is obviously quadratic in n, while space is linear in n. This latter version of the segmentation algorithm turned out to be equivalent to that of Finkelstein and Roytberg,[5] displayed in equation (9).

We can extend equation (9) by computing the *partition function* over all segmentations, defined by $Z = Z(n)$, where by induction on i we define

$$Z(i) = \begin{cases} 1 & \text{if } i = 0 \\ \exp(f_{1,i}) + \sum_{1 \le k < i} Z(k) \cdot \exp(f_{k+1,i}/RT) & \text{else} \end{cases} \tag{10}$$

where R is the universal gas constant and T absolute temperature.[d] Using the partition function, one can *sample* high fitness (suboptimal) segmentations to determine the *maximum expected accurate* segmentation, in analogy to the maximum expected accurate RNA secondary structure, denoted McCaskill-MEA, as described in Kiryu et al.[7] For reasons of space, we do not further describe the partition function, sampling, or MEA segmentations in this article.

2.3. *Parametric dynamic programming method*

In this section, we describe a new algorithm that computes, given an RNA sequence (structure) and integer K, the optimal segmentation into k segments, for each $1 \le k \le K$. This algorithm runs in time $O(n^2K)$ and space nK.

The underlying idea of the algorithm described in this section is to maintain separately indexed tables $PF(m,i)$ for the *parametric* optimal fitness over all segmentations of $[1,m]$ into i segments; i.e. we inductively define $PF(m,i) = \max_{1 \le k < m} PF(k, i-1) + f_{k+1,m}$. (See Figure 2 for pseudocode of algorithm.) Clearly the

[d] In this setting, RT is simply a constant and can be taken to be equal to 1.

```
1   int[n][n] segmentation(rna){
2     // rna is RNA sequence, f is pre-computed base fitness function.
3     n = len(rna); SplitPoints = ∅
4     for d = 1 to n − 1
5       for i = 1 to n
6         j = i + d
7         if (j > n) then break
8         max = f_{i,j}
9         for k = i to j − 1
10          val = F(i, k) + f_{k+1,j}
11          if val > max then
12            max = val
13            splitPoint = k
14        F(i, j) = max
15        SplitPoints[i,j] = splitPoint
16    return SplitPoints
17    }
```

Fig. 3. Algorithm to determine optimal segmentation of each subsequence $[i, j]$, with run time $O(n^3)$ and space $O(n^2)$. This algorithm is inspired by the Nussinov-Jacobson algorithm,[14] which determines the secondary structure having maximum number of base pairs. Assuming the base fitness function f has been precomputed, this algorithm computes the fitness $F(i, j)$ for the optimal segmentation of each subsequence $[i, j]$. The optimal segmentation can be computed by traceback using the information from $SplitPoints$.

run time of parametric segmentation is $O(n^2 \cdot K)$ and the space requirement is $O(n \cdot K)$, when computing optimal segmentations of $[1, n]$ into k segments, for all $k \leq K$.

2.4. *Optimal fitness of all segmentations of subwords*

In this section, we describe a cubic time algorithm to compute the optimal segmentation, simultaneously for all subwords $[i, j]$, where $1 \leq i \leq j \leq n$. This algorithm is inspired by the Nussinov-Jacobson algorithm,[14] which determines the secondary structure having a maximum number of base pairs. (See Figure 3 for the pseudocode of this algorithm.) By using this algorithm, where the base fitness function f is defined from the contact map obtained by RNAview,[19] one could produce segmentations where low scoring initial portions $[1, i-1]$ and low scoring terminal portions $[j+1, n]$ are dropped, thus leaving a segmentation of subword $[i, j]$. The manual segmentations of Jaeger et al.[6] described in the Results section appears to be of this type.

2.5. *Fitness Functions*

We have considered different base fitness functions for RNA secondary structure, all of them fitting in the following scheme:

$$f_{i,j} = \frac{\sum_{i \leq x < y \leq j} w_1 \cdot p_{x,y} - \sum_{x \in [i,j]} \sum_{y \notin [i,j]} w_2 \cdot p_{x,y}}{j - i + 1} \tag{11}$$

where $f_{i,j}$ is the fitness function of segment $[i, j]$ and $p_{x,y}$ can be the following:

- The base pair probability between nucleotides x and y as computed by RNAfold -p (or by RNAplfold -p).
- The existence (or not) of a base pair between nucleotides x and y as computed by RNAview.

The pseudocode to compute the fitness function for base pairing probabilities (and equivalently for contact maps) is depicted in Figure 4.

We have also considered a 3D fitness function (which can also be used for proteins or other molecules) which consists on minimizing the normalized volume (by computing a tessellation with Qhull[1]). The fitness

```
1   void fitness(rna){
2     using RNAfold -p determine base pairing probabilities p_{i,j}
3     n = len(rna)
4     for d = 0 to n
5       for i = 1 to n
6         j = i + d
7         if j>n then break
8         if i==j
9           sum = 0.0
10        else //i < j
11          sum = f_{i,j-1}
12        for k = 1 to n
13          if i ≤ k < j
14            sum += (w_1 + w_2) · p_{k,j}
15          else if k < i
16            sum -= w_2 · p_{k,j}
17          else if k > j
18            sum -= w_2 · p_{j,k}
19        f_{i,j} = sum/(j-i+1)
20    return f
21  }
```

Fig. 4. The base fitness $f_{i,j}$ of segment $[i,j]$ is defined by normalizing the sum $\sum_{i\leq x<y\leq j} w_1 \cdot p_{x,y} - \sum_{x\in[i,j]} \sum_{y\notin[i,j]} w_2 \cdot p_{x,y}$ by segment length, where base pairing probabilities $p_{x,y}$ are computed by RNAfold -p. Straightforward implementation of the formula for $f_{i,j}$ requires $O(n^4)$ time. In contrast, this figure depicts pseudocode to compute base fitness function f in time $O(n^3)$.

function of segment $[i,j]$ is thus the negative normalized volume as calculated by Qhull, i.e. $f_{i,j} = -\frac{vol}{diam}$, where *vol* and *diam* respectively denote the volume and diameter of the Voronoi polyhedra of residues $i,\ldots,j$.

3. Results

3.1. *Finding Precursor microRNAs*

As previously mentioned, we applied our segmentation to help determine non-coding RNA genes within a window of flanking nucleotides. Many non-coding RNA gene finders use a moving window strategy, where the likelihood that the fixed-size window contents contain a non-coding RNA gene is represented by a numerical score. To that end, we tested our segmentation algorithm to detect precursor microRNA within a window of flanking nucleotides, where the flanking nucleotides were extracted from the EMBL genomic file. Our experiment can be summarized as follows.

- Download all the accession codes for precursor micro RNA, riboswitches and SECIS (only results for precursor microRNA are reported here).
- Download the EMBL data for each of the above with 500 flanking nucleotides on each side (when possible). In some cases, there were fewer than 500 nucleotides to the left, or less than 500 nucleotides to the right, in which case the sequence was skipped.
- Run segmentation algorithm by varying the following parameters:
  - flanking nts (50, 100, 200, 400)
  - max segment size(100, 1000 which translate to not having a maximum size in practice)
  - weight combinations $w_1 \quad w_2(10, 01, 11, 21, 12, 51, 15)$
  - base pairing probabilities, obtained by RNAfold -p
- Report histograms and measures of accuracy.
- Run segmentation with flanking nucleotides replaced by random combination (permutation)

Table 1. Boundary prediction: precursor microRNA from Rfam of size ≤ 100 nt.

	Left Border		Right Border		Stats	
Parameters	**Mean**	**St Dev**	**Mean**	**St Dev**	**Sensitivity**	**PPV**
$w_1 = 1, w_2 = 0 - 50$	9.984	16.193	-9.486	17.115	0.774	0.990
$w_1 = 1, w_2 = 0 - 100$	10.032	15.814	-10.035	17.441	0.770	0.992
$w_1 = 1, w_2 = 0 - 200$	9.691	15.059	-10.887	17.450	0.765	0.993
$w_1 = 1, w_2 = 0 - 400$	10.206	15.899	-11.038	18.063	0.761	0.992
$w_1 = 0, w_2 = 1 - 50$	-2.453	14.132	1.891	13.149	0.927	0.888
$w_1 = 0, w_2 = 1 - 100$	-1.807	7.102	1.379	11.489	0.969	0.936
$w_1 = 0, w_2 = 1 - 200$	-3.331	11.541	4.199	10.462	0.963	0.887
$w_1 = 0, w_2 = 1 - 400$	-4.113	11.803	3.351	12.122	0.949	0.876
$w_1 = 1, w_2 = 1 - 50$	-0.598	15.612	1.235	14.142	0.922	0.903
$w_1 = 1, w_2 = 1 - 100$	-0.701	9.917	1.428	10.856	0.956	0.935
$w_1 = 1, w_2 = 1 - 200$	-1.492	11.211	1.624	10.344	0.945	0.916
$w_1 = 1, w_2 = 1 - 400$	-1.322	12.006	1.483	12.571	0.935	0.909
$w_1 = 2, w_2 = 1 - 50$	-0.524	15.994	0.125	15.654	0.913	0.905
$w_1 = 2, w_2 = 1 - 100$	0.376	12.380	1.096	13.667	0.934	0.927
$w_1 = 2, w_2 = 1 - 200$	-0.299	15.326	-0.132	14.447	0.920	0.918
$w_1 = 2, w_2 = 1 - 400$	-0.640	15.196	-0.266	17.415	0.911	0.906
$w_1 = 1, w_2 = 2 - 50$	-1.958	12.122	0.846	11.772	0.933	0.908
$w_1 = 1, w_2 = 2 - 100$	-0.846	9.419	1.547	8.970	0.964	0.939
$w_1 = 1, w_2 = 2 - 200$	-2.251	10.180	2.080	9.906	0.953	0.911
$w_1 = 1, w_2 = 2 - 400$	-2.955	11.547	2.168	11.022	0.944	0.894
$w_1 = 5, w_2 = 1 - 50$	0.740	15.887	-0.723	17.444	0.901	0.913
$w_1 = 5, w_2 = 1 - 100$	1.968	15.787	-1.572	17.908	0.886	0.921
$w_1 = 5, w_2 = 1 - 200$	2.524	16.453	-1.482	16.025	0.886	0.927
$w_1 = 5, w_2 = 1 - 400$	2.392	17.011	-2.727	18.053	0.868	0.920
$w_1 = 1, w_2 = 5 - 50$	-2.408	13.061	1.129	12.430	0.931	0.899
$w_1 = 1, w_2 = 5 - 100$	-1.431	8.053	1.203	10.187	0.966	0.939
$w_1 = 1, w_2 = 5 - 200$	-2.997	10.655	3.569	10.482	0.960	0.894
$w_1 = 1, w_2 = 5 - 400$	-3.843	11.888	3.297	11.626	0.949	0.878

Tables 1 and 3 show, respectively, the results of our segmentation with and without maximum segment size limit. The main conclusions that can be drawn are the following:

- Certain weight combinations yield very poor results, specially in the case of $w_1 = 0$, $w_2 = 1$ and $w_1 = 1$, $w_2 = 0$ which means that both characteristics of inside and cross-segments are necessary.
- Giving a higher weight to cross-segment characteristics does not yield the best results which indicates that the local structure of the precursor micro RNA is stronger than its lack of potentially base pair with other regions in other suboptimal configurations.
- Overall, the weight combination $w_1 = 2$, $w_2 = 1$ achieves the best results.
- The algorithm is robust to the size of the flanking nucleotides.
- Limiting the maximum size of the segment does impact efficiency. Interestingly, the weight combination $w_1 = 5$, $w_2 = 1$ performs better in this case. This seems to indicate that a higher weight to inside base pairings is necessary for larger instances since it reinforces its locality, i.e., if there are more nucleotides there are potentially more possibilities of cross-segment base pairings which (in this case), for nucleotides farther away in the primary sequence might not be very significant.

A very useful tool to visualize the quality of the results is to plot the distributions of both left and right end segments of the calculated precursor micro RNA. This information is depicted in Figure 5. Note that both distributions are very similar and they clearly show a higher concentration of segmentations in which the distance from the actual end segment and the calculated one are very close to 0.

It is conjectured that precursor micro RNAs have a very strong local structure with which the flanking nucleotides cannot compete. To prove that our algorithm is sensitive to that local structure (which is consistent with the fact that a higher weight for inside segment yields better results) we have carried out a set of experiments in which we permuted the flanking nucleotides before performing the segmentation. Results of

Table 3. Boundary prediction: precursor microRNA from Rfam, no size limit.

	Left Border		Right Border		Stats	
Parameters	**Mean**	**St Dev**	**Mean**	**St Dev**	**Sensitivity**	**PPV**
$w_1 = 1, w_2 = 0 - 50$	9.113	16.578	-8.624	17.574	0.782	0.984
$w_1 = 1, w_2 = 0 - 100$	9.325	16.005	-9.341	17.723	0.777	0.989
$w_1 = 1, w_2 = 0 - 200$	9.016	15.235	-10.222	17.661	0.772	0.990
$w_1 = 1, w_2 = 0 - 400$	9.479	16.141	-10.322	18.314	0.769	0.988
$w_1 = 0, w_2 = 1 - 50$	-48.997	0.057	49.994	0.113	1.000	0.467
$w_1 = 0, w_2 = 1 - 100$	-98.990	0.098	100.000	0.000	1.000	0.304
$w_1 = 0, w_2 = 1 - 200$	-199.000	0.000	200.000	0.000	1.000	0.179
$w_1 = 0, w_2 = 1 - 400$	-399.000	0.000	400.000	0.000	1.000	0.099
$w_1 = 1, w_2 = 1 - 50$	-37.814	17.105	39.601	17.305	0.993	0.552
$w_1 = 1, w_2 = 1 - 100$	-58.402	42.203	62.003	42.980	0.978	0.507
$w_1 = 1, w_2 = 1 - 200$	-58.929	72.924	57.685	72.334	0.966	0.604
$w_1 = 1, w_2 = 1 - 400$	-60.941	99.448	65.014	99.134	0.957	0.614
$w_1 = 2, w_2 = 1 - 50$	-28.331	21.168	28.611	22.686	0.980	0.649
$w_1 = 2, w_2 = 1 - 100$	-34.624	39.918	33.177	40.922	0.956	0.666
$w_1 = 2, w_2 = 1 - 200$	-22.457	34.832	25.801	39.003	0.960	0.725
$w_1 = 2, w_2 = 1 - 400$	-25.801	43.108	30.119	45.335	0.948	0.702
$w_1 = 1, w_2 = 2 - 50$	-42.132	13.798	43.605	13.462	0.997	0.514
$w_1 = 1, w_2 = 2 - 100$	-76.897	34.990	80.740	33.833	0.990	0.395
$w_1 = 1, w_2 = 2 - 200$	-123.775	82.962	120.785	84.169	0.989	0.385
$w_1 = 1, w_2 = 2 - 400$	-214.867	169.619	211.993	169.320	0.979	0.343
$w_1 = 5, w_2 = 1 - 50$	-11.170	21.748	10.871	21.846	0.943	0.802
$w_1 = 5, w_2 = 1 - 100$	-9.219	26.431	9.563	26.815	0.913	0.821
$w_1 = 5, w_2 = 1 - 200$	-7.762	22.604	7.460	22.576	0.923	0.836
$w_1 = 5, w_2 = 1 - 400$	-6.720	25.015	5.871	24.055	0.905	0.840
$w_1 = 1, w_2 = 5 - 50$	-45.601	10.258	46.775	9.366	0.998	0.488
$w_1 = 1, w_2 = 5 - 100$	-87.682	25.800	91.077	23.983	0.998	0.341
$w_1 = 1, w_2 = 5 - 200$	-174.678	53.153	167.801	62.362	0.999	0.226
$w_1 = 1, w_2 = 5 - 400$	-339.252	118.600	352.521	104.862	0.995	0.136

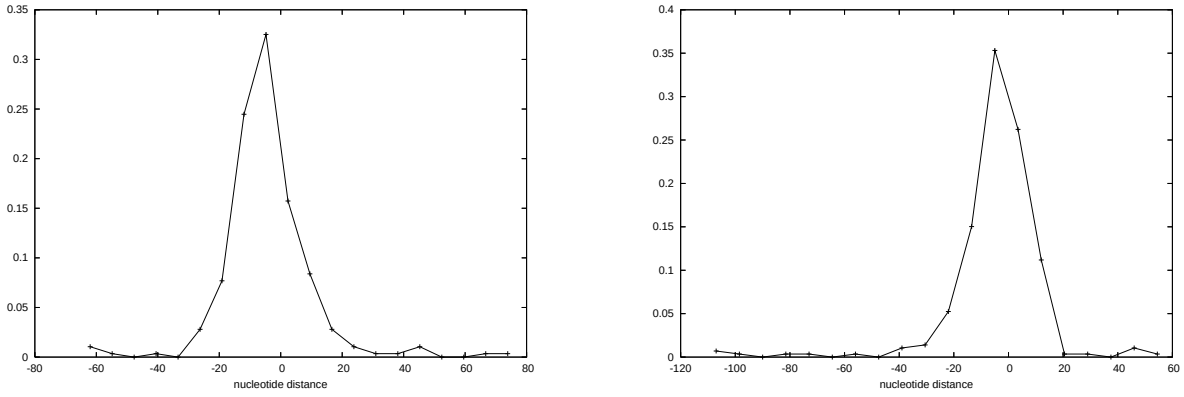


Fig. 5. Distribution of boundary discrepancy for 5′ end (left panel) and 3′ end (right panel) of precursor microRNAs within window having 400 nt flanking both on left and right of microRNA. Analysis performed over all precursor microRNAs from Rfam 9.1 (January 2009, 454 subfamilies). Here, discrepancy is defined as the absolute value of the difference between predicted boundary and real boundary.

this are shown in Table 5 (where we compare them against the *normal* seuquence, i. e., that with the actual flanking nucelotides), and the distributions are depicted in Figure 6. These results are for weight combination $w_1 = 2$, $w_2 = 1$ with 400 flanking nucleotides and with no maximum segment size limit. As it can be seen, results are very similar to those for the actual sequence which proves the robustness of our approach.

Table 5. Boundary prediction: permuted versus unpermuted tails of precursor miRNA

	Left Border		Right Border		Stats	
Parameters	**Mean**	**St Dev**	**Mean**	**St Dev**	**Sensitivity**	**PPV**
$Normal - 50$	-0.524	15.994	0.125	15.654	0.913	0.905
$Normal - 100$	0.376	12.380	1.096	13.667	0.934	0.927
$Normal - 200$	-0.299	15.326	-0.132	14.447	0.920	0.918
$Normal - 400$	-0.640	15.196	-0.266	17.415	0.920	0.918
$Permuted - 50$	1.129	14.073	-2.334	15.948	0.899	0.939
$Permuted - 100$	0.251	11.629	-1.074	14.183	0.928	0.944
$Permuted - 200$	1.180	14.354	0.113	12.329	0.918	0.933
$Permuted - 400$	0.287	14.406	-0.955	15.669	0.910	0.924

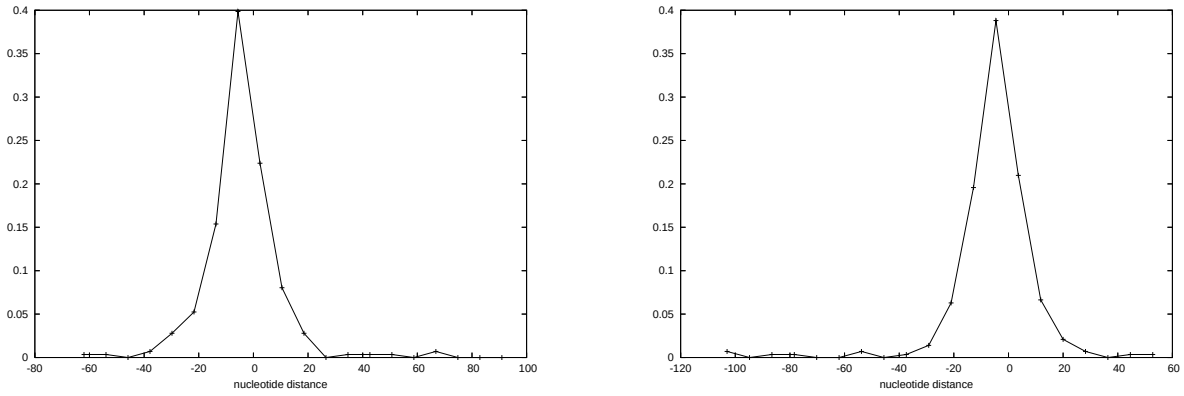


Fig. 6. Distribution of boundary discrepancy for 5′ end (left panel) and 3′ end (right panel) of precursor microRNAs within window having 400 nt flanking both on left and right of microRNA. Analysis performed over all precursor microRNAs from Rfam 9.1 (January 2009, 454 subfamilies) with permuted flanking nucleotides.

Table 7. Manual and computed segmentations of 16S rRNA.

Organism & method	seg 1	seg 2	seg 3	seg 4	fit 1	fit 2	fit 3	fit 4
E. coli (manual)	$27 - 509$	$515 - 857$	$866 - 1326$	$1329 - 1476$	0.628	0.623	0.462	0.658
E. coli (computed)	$1 - 338$	$339 - 350$	$351 - 1132$	$1133 - 1542$	0.399	0.635	0.573	0.570
rat mitochondrial (manual)	$20 - 279$	$279 - 509$	$526 - 829$	$829 - 953$	0.550	0.459	0.559	0.323
rat mitochondrial (computed)	$1 - 459$	$460 - 484$	$485 - 928$	$929 - 953$	0.551	0.760	0.785	0.6221
H. volcanii (manual)	$21 - 495$	$501 - 857$	$865 - 1342$	$1342 - 1474$	0.600	0.618	0.597	0.617
H. volcanii (computed)	$1 - 84$	$85 - 405$	$406 - 433$	$434 - 1476$	0.551	0.760	0.785	0.622
C. reinhardii (manual)	$27 - 509$	$515 - 857$	$866 - 1326$	$1329 - 1476$	0.632	0.622	0.596	0.647
C. reinhardii chloroplast (computed)	$1 - 754$	$755 - 1350$	$1351 - 1413$	$1414 - 1476$	0.480	0.466	0.673	.563

3.2. *Finding RNA domains*

Our initial motivation for developing a segmentation algorithm was to determine an automated method to decompose large X-ray structures of RNA, such as PDB code 1FFK, into coherent units, or domains. Also, to segment RNA sequences in which secondary structure is available.

With the intent of benchmarking the accuracy of MFOLD, Jaeger et al.[6] performed a manual segmentation of *E. coli* 16S rRNA, as well as the 16S-like rRNA domains of rat mitochondria, *Halobacterium volcanii*, and *Chlamydomonas reinhardii* chloroplast into 4 segments.

In Table 7 we present results from the manual and optimal segmentation of 16S rRNA into four segments. Optimal segmentation is calculated using base pairing probabilities with weights $w_1 = 2$, $w_2 = 1$ (these weights were determined by previous benchmarking experiments). In that table, column headings, *seg* abbreviates segment, while *fit* abbreviates fitness. The manual segmentation was created by Jaeger et al.,[6] while the computed segmentation used the parametric algorithm described in Figure 2. Note that we could have modified (but did not) the parametric segmentation to discard with no penalty a small initial

Table 8. Average, min, max fitness over all segments in manual segmentation of 16S rRNA.

Organism & method	avg	min	max	fit 1	fit 2	fit 3	fit 4
E. coli	0.308	−1.000	0.857	0.628	0.623	0.462	0.658
rat mitochondrial	0.190	−1.000	0.857	0.550	0.459	0.559	0.323
H. volcanii	0.292	−1.000	0.857	0.600	0.618	0.597	0.617
C. reinhardii	0.299	−1.000	0.916	0.632	0.622	0.596	0.647

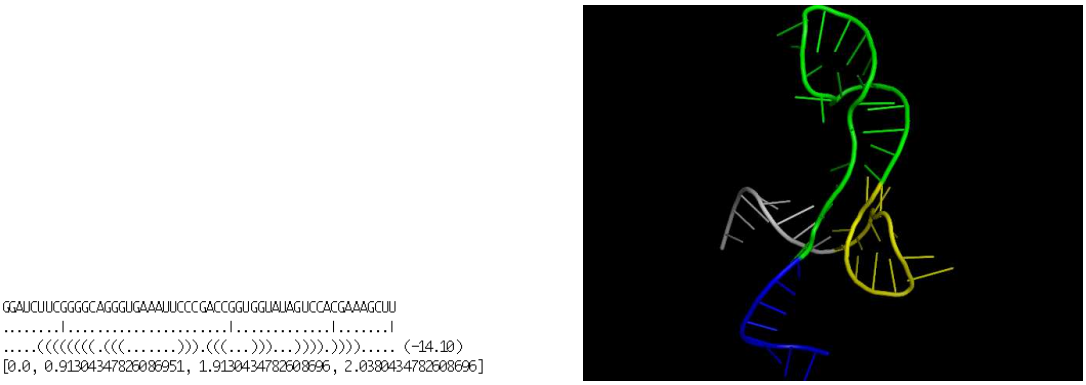


Fig. 7. (Left) Segmentation of the riboswitch with PDB code 3F4H:X. This optimal segmentation has 4 segments, respectively of weights 0.0, 0.913, 1.913, 2.038. Segmentation produced by applying software RNAview,[19] which annotates all hydrogen bonds (canonical base pairs, non-canonical base pairs, single nucleotide stacking). Using the resulting contact map, we determined an optimal segmentation, where the fitness function used involved a weight of 2 for contacts within the same segment and a penalty of 1 for contacts between segments. (Right) Three-dimensional display of the same segmentation, where segments of PDB file 3F4H:X are demarcated in different colors, using Pymol.

and final segment. Since this was not done, all computed segmentations begin at nucleotide 1 and end at the last nucleotide, unlike that from the manual segmentation. This explains how a manual segmentation can paradoxically have higher fitness than the computed *optimal* segmentation.

Even though our optimal segmentation does not always resemble the manual segmentation, from Table 8 (which shows average, minimum and maximum fitness for all segments) it can be seen how all manually calculated segments have fitnesses higher than the average. This seems to indicate that our fitness function correlates with reality but that possibly more specific information needs to be added to boost efficiency.

Figure 7 presents two alternative representations of the optimal segmentation of FMN riboswitch (3F4H:X) with respect to the base fitness function defined from the contact map (base pairing) output from RNAview. The left panel of Figure 7 depicts the segmentation in text format while the right panel displays the segmentation as a Pymol image in which different segments appear in different colors. This latter image shows more clearly the division in domains, which appear to be reasonable in light of its 3D representation.

Alternatively, the base fitness function can be defined using Voronoi tessellation computed by Qhull. Segmentations obtained in this manner are applicable to both RNA and protein 3-dimensional structures; indeed, Figure 8 displays optimal segmentations of the secretin protein with PDB code 1Y9L and of the metabotropic glutamate receptor protein (mGluR) with PDB code 1EWT. Note that segments determined by structural segmentation are not simply α-helices or β-strands.

4. Conclusions

In this paper, we present a dynamic programming algorithm that produces an optimal segmentation for RNA, given either an RNA sequence, or secondary structure, or tertiary structure. Given 3-dimensional RNA structures, the fitness function can be defined using Voronoi tessellation obtained by Qhull or alternatively using contact maps produced by RNAview. Given an RNA sequence, the fitness function can be defined

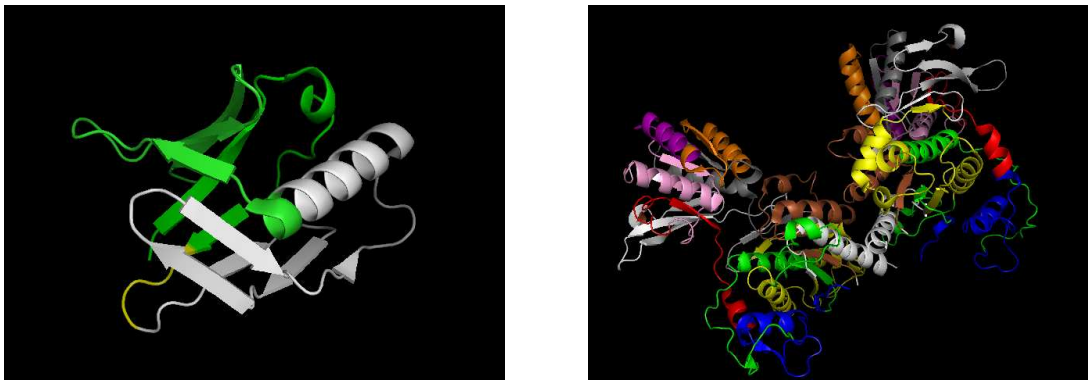


Fig. 8. (Left) Segmentation of the secretin protein with PDB code 1Y9L. Gram-negative pathogens such as Shigella, Salmonella, Yersinia and Pseudomonas use a type III secretion apparatus to translocate virulence proteins into host cells. X-ray structure determined by Lario et al.[9] (Right) Segmentation of the metabotropic glutamate receptor (mGluR) with PDB code 1EWT. X-ray structure determined by Kunishima et al.[8] Each segment in the optimal segmentation is displayed in a different color. Images produced by Pymol.

from base pairing probabilities computed by McCaskill's algorithm.[e]

Optimal parametric segmentation, described in Figure 2, as well as simultaneous optimal segmentation of all intervals, described in Figure 3, both appear to be new. In future work, we plan to describe the dynamic programming computation of the partition function for segmentations, as suggested in equation (10), and to stochastically sample (suboptimal) segmentations. Applications of segmentation in the context of RNA include *(i)* an automated method to decompose large RNA 3-dimensional structures into domains suitable for estimating knowledge-based potentials or instead for benchmarking secondary structure algorithms, as done manually by Jaeger et al.,[6] *(ii)* a method to determine the possible $5'$ and $3'$ boundary of non-coding RNA gene found within a window of a genome scanning algorithm. As future work we would like to add other metrics to our fitness function as well as to perform exhaustive benchmarking on 3D segmentation using Qhull. Preliminary results on trans-membrane proteins (Figure 8) show the potential of this fitness function whenever X-ray structures are available.

Acknowledgments

We would like to thank Y. Ding for some assistance with some scripts, D.H. Mathews for the reference to the manual segmentations of Jaeger et al.,[6] M.A. Roytberg for kindly sending a reprint of Finkelstein and Roytberg[5] (dynamic programming), and three anonymous referees for helpful suggestions.

Research of I. Dotú was supported by a grant from Fundacion Caja Madrid, while that of W.A. Lorentz was supported by NSF grants DBI-0543506. Research of P. Van Hentenryck was supported by NSF DMI-0600384. Research of P. Clote was supported by the Foundation Digiteo - Triangle de la Physique, as well as by NSF grants DBI-0543506 and DMS-0817971. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. Additional thanks to the Deutscher Akademischer Austauschdienst for funding a visit of P. Clote to Martin Vingron's group in the Max Planck Institute of Molecular Genetics.

References

1. C.B. Barber, D.P. Dobkin, and H.T. Huhdanpaa. The Quickhull algorithm for convex hulls. *ACM Trans. on Mathematical Software*, 22(4):469–483, 1996.
2. R. Bellman. On the approximation of curves by line segments using dynamic programming. *Communications of the ACM*, 4(6):284, 1961.

[e]Of course, given an RNA secondary structure S, one can alternatively consider the base pairing probabilities $p_{i,j} = 1$ exactly when $(i, j) \in S$.

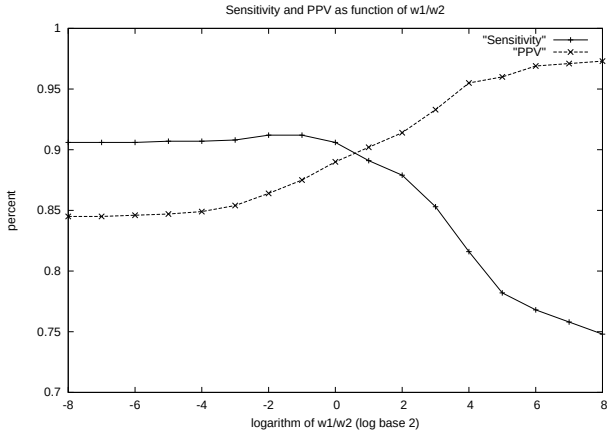


Fig. 9. Sensitivity and positive predictive values (PPV) for gene detection of precursor microRNAs from Rfam with a variable number of flanking nucleotides on both left and right sides, ranging from 20 to 100 nt. Sensitivity (TP/P) and PPV (TP/PP), defined earlier in the caption of Table 3.1, are computed as a function of the logarithm of the ratio w_1/w_2 of weights w_1, w_2. Values of $\log_2(w_1/w_2)$ range from -8 to 8; i.e. w_1/w_2 ranges from 0.004 to 256. Note the decrease in sensitivity and increase in PPV as weight ratio increases. (All logarithms are with respect to base 2.)

3. P. Benaola-Galván, R. Román-Roldán, and J. L. Oliver. Compositional segmentation and long-range fractal correlations in DNA sequences. *Physical Review E*, 53:5181–5189, 1996.
4. P. Clote and R. Backofen. *Computational Molecular Biology: An Introduction*. John Wiley & Sons, 2000. 279 pages.
5. A. V. Finkelstein and M. A. Roytberg. Computation of biopolymers: a general approach to different problems. *Biosystems*, 30(1-3):1–19, 1993.
6. J. A. Jaeger, D. H. Turner, and M. Zuker. Improved predictions of secondary structures for RNA. *Proc. Natl. Acad. Sci. U.S.A.*, 86(20):7706–7710, October 1989.
7. H. Kiryu, T. Kin, and K. Asai. Robust prediction of consensus secondary structures using averaged base pairing probability matrices. *Bioinformatics*, 23(4):434–441, February 2007.
8. N. Kunishima, Y. Shimada, Y. Tsuji, T. Sato, M. Yamamoto, T. Kumasaka, S. Nakanishi, H. Jingami, and K. Morikawa. Structural basis of glutamate recognition by a dimeric metabotropic glutamate receptor. *Nature*, 407(6807):971–977, October 2000.
9. P. I. Lario, R. A. Pfuetzner, E. A. Frey, L. Creagh, C. Haynes, A. T. Maurelli, and N. C. Strynadka. Structure and biochemical analysis of a secretin pilot protein. *EMBO J.*, 24(6):1111–1121, March 2005.
10. W. Li, P. Bernaola-Galvan, P. Carpena, and J. L. Oliver. Isochores merit the prefix 'iso'. *Comput. Biol. Chem.*, 27(1):5–10, February 2003.
11. W. Li, P. Bernaola-Galvan, F. Haghighi, and I. Grosse. Applications of recursive segmentation to the analysis of DNA sequences. *Comput. Chem.*, 26(5):491–510, July 2002.
12. W. Li, G. Stolovitzky, P. Bernaola-Galvan, and J. L. Oliver. Compositional heterogeneity within, and uniformity between, DNA sequences of yeast chromosomes. *Genome Res.*, 8(9):916–928, September 1998.
13. Jianhua Lin. Dvergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.
14. R. Nussinov and A. B. Jacobson. Fast algorithm for predicting the secondary structure of single stranded RNA. *Proceedings of the National Academy of Sciences, USA*, 77(11):6309–6313, 1980.
15. J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.
16. R. Román-Roldán, P. Benaola-Galván, and J. L. Oliver. Sequence compositional complexity of DNA through an entropic segmentation method. *Physical Review Letters*, 80(6):1344–1347, February 1998.
17. S. C. Schmidler, J. S. Liu, and D. L. Brutlag. Bayesian segmentation of protein secondary structure. *J. Comput. Biol.*, 7(1-2):233–248, Feb-Apr 2000.
18. S. R. Sunyaev, G. A. Bogopolsky, N. V. Oleynikova, P. K. Vlasov, A. V. Finkelstein, and M. A. Roytberg. From analysis of protein structural alignments toward a novel approach to align protein sequences. *Proteins.*, 54(3):569–582, February 2004.
19. H. Yang, F. Jossinet, N. Leontis, L. Chen, J. Westbrook, H.M. Berman, and E. Westhof. Tools for the automatic identification and classification of RNA base pairs. *Nucleic Acids Res.*, 31(13):3450–3560, 2003.

RNAZ 2.0: IMPROVED NONCODING RNA DETECTION

ANDREAS R. GRUBER[1,2], SVEN FINDEIß[1], STEFAN WASHIETL[2,3],
IVO L. HOFACKER[2] AND PETER F. STADLER[1,2]

[1] *Bioinformatics Group, Department of Computer Science, and Interdisciplinary Center for Bioinformatics, University of Leipzig*
Härtelstrasse 16-18, D-04107 Leipzig, Germany

[2] *Institute for Theoretical Chemistry, University of Vienna*
Währingerstrasse 17, A-1090 Wien, Austria.

[3] *European Molecular Biology Laboratory – European Bioinformatics Institute*
Wellcome Trust Genome Campus, Hinxton, Cambridgeshire, CB10 1SD, UK

`RNAz` is a widely used software package for *de novo* detection of structured noncoding RNAs in comparative genomics data. Four years of experience have not only demonstrated the applicability of the approach, but also helped us to identify limitations of the current implementation. `RNAz 2.0` provides significant improvements in two respects: (1) The accuracy is increased by the systematic use of dinucleotide models. (2) Technical limitations of the previous version, such as the inability to handle alignments with more than six sequences, are overcome by increased training data and the usage of an entropy measure to represent sequence similarities. `RNAz 2.0` shows a significantly lower false discovery rate on a dinucleotide background model than the previous version. Separate models for structural alignments provide an additional way to increase the predictive power. `RNAz` is open source software and can be obtained free of charge at: `http://www.tbi.univie.ac.at/~wash/RNAz/`

Keywords: RNA structure; noncoding RNA; structure conservation; comparative genomics; gene prediction

1. Introduction

Noncoding RNAs (ncRNAs) are transcripts that are not translated to proteins but function directly on the RNA level. During the past few years it has become evident that such "RNA genes" are more common than previously thought. MicroRNAs, for instance, have profoundly changed our view of gene regulation, and several completely new classes of ncRNAs were discovered recently.[1] They have been found to be involved in such diverse processes as transcriptional regulation,[2–4] post-transcriptional regulation,[5] chromatin modification and epigenetics,[6,7] and development.[8] Non-coding RNAs thus are key players in cellular regulation, a realization that has also moved the computational analysis and the annotation of ncRNAs at genome-wide scales into the focus of attention.

With the rapidly increasing availability of genomic sequence data, the *de novo* prediction of ncRNAs is of particular interest. While protein gene prediction is a classical problem in computational biology and has been studied for more than 15 years, RNA gene prediction is still in its infancy. Nevertheless, significant progress has been made regarding the prediction of "structured ncRNAs". This class of ncRNAs is characterized by evolutionary conserved secondary structures which appear to be important for their function. Most of the well-characterized ncRNAs belong to this class. Leading software tools developed for *de novo* RNA gene finding therefore use evolutionary conservation of functional secondary structures as the main signal to detect these ncRNAs.[9–13]

`RNAz` also detects structural ncRNAs by means of a comparative approach. In addition to measuring evolutionary conservation, however, it also explicitly evaluates the thermodynamic stability of the secondary structure.[14] A support vector machine (SVM) is then used to evaluate both criteria. `RNAz 1.0` has been used successfully to map structural ncRNAs in a wide variety of genomes.[15–20] A large number of these predictions have also been verified experimentally.[21–23] Moreover, the generic approach and many algorithmic details developed for `RNAz 1.0` have been re-used, extended, and adapted to other problems in the field of RNA gene-finding.[11,24–30]

The wide-spread use of `RNAz 1.0` also helped to identify some of its limitations and to point our directions

for improvements. In this contribution, we describe a major update of the RNAz program. It is based on the results of two follow-up studies,[31,32] on our experiences gained during many real-life applications, in particular the ENCODE pilot project,[33,34] and last but not least, on the user feedback we received over the past four years.

One major improvement is that RNAz 2.0 now allows to calculate thermodynamic stability scores based on a dinucleotide background model. It has been noted early-on that folding algorithms utilizing stacking energies of adjacent base-pairs in their energy model are sensitive to the dinucleotide content.[35] In the context of genome-wide ncRNA predictions, this effect can lead to an increased number of false positive calls as pointed out several times.[32,33,36] The new dinucleotide model in RNAz 2.0 now avoids this source of potential false positives and increases the accuracy of the program.

Another major limitation of RNAz 1.0 was the fact that only alignments with at most six sequences could be scored. This rather arbitrary restriction was the result of the limited amount of comparative data sets that were available at the time. During the past few years, however, comparative data sets have grown massively and therefore we adapted the algorithm to allow flexible analysis of alignments of any size.

2. Methods

2.1. *Overview of the RNAz algorithm*

RNAz predicts functional RNA structures on two independent criteria: (i) thermodynamic stability and (ii) structural conservation.

A common way to express thermodynamic stability is in terms of a z-score. This is simply the number of standard deviations by which the minimum free energy (MFE) deviates from the mean MFE of a set of randomized sequences with the same length and base composition. A negative z-score thus indicates that a sequence is more stable than expected by chance. As this procedure involves energy evaluation of a large set of random sequences it is not applicable for large-scale genomic screens. RNAz instead uses support vector regression (SVR) to estimate the mean and the standard deviation based on the nucleotide composition of a sequence.

RNAz evaluates evolutionary conservation of RNA structures in terms of the structure conservation index (SCI). A consensus secondary structure is predicted using the RNAalifold algorithm,[42] which is an extension of standard minimum free energy folding algorithms with the constraint that all sequences have to fold into a common structure. Compensatory mutations, i.e. mutations that preserve a certain base pair, yield bonus energies, while inconsistent mutations add penalty energies. RNAz measures structural conservation by calculating the ratio of the consensus folding energy to the unconstrained folding energies of the single sequences.

Both criteria are combined by another support vector machine model that classifies the input alignment as "structural RNA" or "other". A graphical overview of the RNAz algorithm is depicted in Fig. 1. In the following, we describe independent refinements of these steps that improve the overall prediction accuracy of the RNAz approach.

2.2. *z-score regression for dinucleotide shuffled sequences*

As in RNAz 1.0, we use support vector regression to compute z-scores for folding energies because the direct approach via repeated shuffling and folding is too costly for genome-wide applications.

In order to efficiently train the regression engine of RNAz 2.0, we used the following grid-like procedure: We first generated synthetic sequences of length 50 with G+C content, A/(A+U) ratio, and C/(C+G) ratio ranging from 0.20 to 0.80 in steps of 0.05. For each of these start sequences we then generated 500,000 mononucleotide shuffled sequences and discarded those sequences where the relative difference between the observed dinucleotide frequency and the expected frequency exceeded the threshold of 1.5. Evaluation on human ENCODE sequences showed that only a small fraction of approximately 1% of the sequences have a higher value and it was hence considered to be a reasonable threshold. Sequences of length 100, 150 and

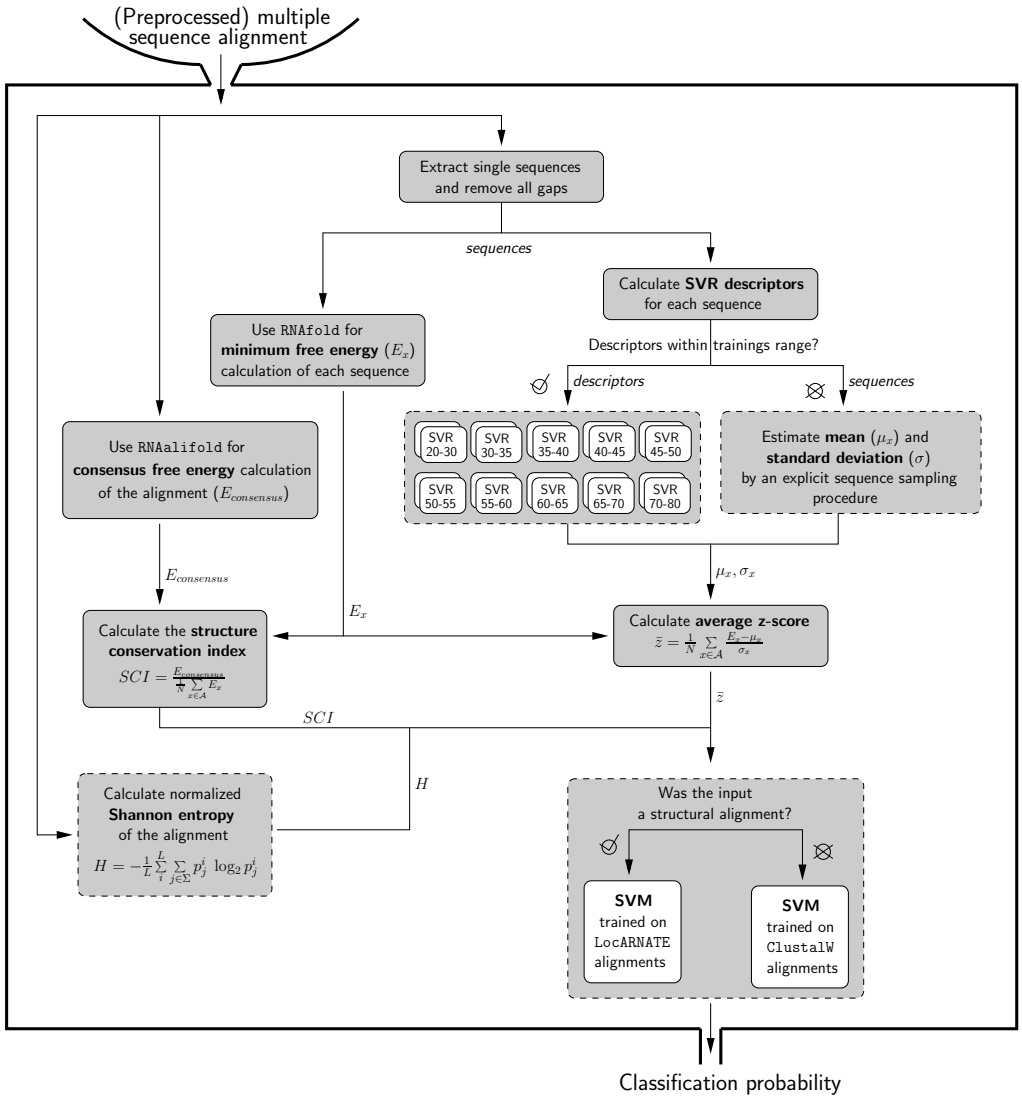


Fig. 1. Outline of the RNAz 2.0 work-flow and algorithm. In a first step large genomic multiple alignments are processed using rnazWindow.pl into smaller alignments. This filtering procedure involves several steps: (i) overlapping windows given a fixed window and step size are created, (ii) sequences that contain too many gaps are removed and (iii) from the remaining sequences only those sequences are kept that meet a predefined average pairwise identity threshold. The resulting alignments are then separately processed by RNAz. First, structure and energy predictions are performed for both the single sequences and the alignment. These results can be immediately combined to calculate the SCI as the measure of the evolutionary conservation of the RNA sequences in the alignment. In a second step, the mean free energy and the standard deviation used for the calculation of the z-score are estimated. For this purpose descriptors based on the nucleotide composition (G+C content, A/(A+U) ratio, C/(C+G) ratio, all 16 dinucleotide frequencies and the length of the sequence) are calculated for each sequence. If descriptors are within the training boundaries they are passed to the corresponding support vector regression (SVR) based on the G+C content. Otherwise, the mean and the standard deviation are evaluated explicitly by folding of 1,000 randomized sequences with the same dinucleotide composition. In a final step the average z-score of the sequences, the SCI and the normalized Shannon entropy of the alignment are passed to the classification SVM, which returns a probability estimate that the given alignment harbours thermodynamically stable and/or evolutionary conserved RNA secondary structures. Parts that are highlighted in dashed boxes are new or modified components of RNAz algorithm. RNAfold and RNAalifold are part of the Vienna RNA Package. Numbers in the SVR boxes indicate the G+C content the particular SVR is trained on. For a detailed explanation of the formulas we refer to section 2.3.

200 where then generated by concatenating the initial set of sequences 2 to 4 times. This initial set can be generated very quickly and served as the basis for the selection of a much smaller, approximately evenly

spaced, training set with representative dinucleotide frequencies. A sequence from the initial set was only added to the representative training set if the Euclidean distance of the dinucleotide frequencies to any sequence already present in the representative set was above a certain threshold (0.075 for a G+C content of 0.20 and 0.80, 0.100 for a G+C content of 0.25, 0.30, 0.70 and 0.75, and 0.125 for the remaining range). For the final training set we also added sequences of length 75, 125 and 175, which were generated as described above, resulting in a total of 1,155,737 training instances.

For each of these instances, we generated 1,000 randomized sequences by the Altschul-Erikson algorithm[37] with the same dinucleotide composition and used `RNAfold`[38] with parameter `-d2` to evaluate their folding free energy. More than 1 million training instances are by far too many to be used in SVM training procedures in reasonable time. For this reason we split the training instances into smaller subsets according to their G+C content. In total we have 10 subsets with at most 150,000 training instances. We used the SVM library `LIBSVM` to train regression models for the mean and the standard deviation for each of the ten subsets. As input features we used the G+C content, the A/(A+U) ratio, the C/(C+G) ratio, all 16 dinucleotide frequencies and the length of the sequence scaled to the interval [0,1]. The regression for estimating the mean free energy was trained to learn energy per nucleotide, while the standard deviation was not scaled. We chose the ν variant of regression and a radial basis function kernel. The standard grid search approach was used to find optimal combinations for SVM parameters. Regression accuracy was monitored on an independent test set compiled from randomly selected sequences of variable length from 50 to 200 nt from the human ENCODE regions. The average number of support vectors for the mean and the standard deviation regression models are 8,763 and 8,607, respectively.

2.3. *Training data generation and training of the SVM classifier*

Training and test sets are based on the data available in the `Rfam` 9.1 database.[39] 93 RNA families were selected based on their signals for thermodynamic stability and structural conservation. The `RNAz 2.0` training set covers a broad range of different RNA families including major classes such as tRNAs, snoRNAs, microRNAs, riboswitches, and bacterial regulatory RNAs.

For each RNA family, a set of alignments with varying numbers of sequences and average pairwise identities was generated using the following strategy: `Rfam` full alignments were used if they contained less than 300 sequences, otherwise we used the seed alignments. For our purpose the use of at most 300 sequences proofed well to generate a set of alignments over the desired range of average pairwise identities. `Rfam` alignments were utilized only as a source to retrieve family members of a particular ncRNA class and only extracted, ungapped RNA sequences were used for subsequent analyses.

First, `Rfam` alignments were filtered to remove nearly identical sequences, so that the training alignments contained sequences with at most 98% identity. The sequences were then re-aligned using `ClustalW`. For each of these ncRNA family alignments we then proceeded as follows: for each number of sequences from 2 to 15 we generated at most 10 alignments with a randomly chosen average pairwise identity between 50 and 98% and with a maximum relative difference in sequence lengths of 65% using `rnazWindow.pl` which is part of the `RNAz` analysis pipeline.[44]

To ensure that this set of positive training examples contained only instances with good structural conservation signals we filtered alignments by using tree editing distances between the structures of the sequences in the alignment as a quality measure of structural conservation. Ordered, rooted trees can be deduced from the dot-bracket notation of RNA secondary structures. Tree editing defines a metric in the space of trees by a set of operations (deletions, insertion and relabeling of nodes) and hence can be used to calculate distances between RNA secondary structures.[31] For each alignment we extracted sequences, removed gaps and calculated the averaged pairwise tree editing distance using `RNAdistance` with options `-d2 -Dh` to enable dangling ends and to use the HIT representation for RNA secondary structures. We repeated this for a set of 100 randomized alignments and calculated an empirical p-value as a measure of structural conservation. Alignments with a p-value higher than 0.05 were removed from the training set. Alignments retained after this filtering procedure were realigned with `ClustalW` with standard options for

application to sequence-based alignments.

For the generation of structural alignments for the training set we chose to use `LocARNATE`,[40] which is a structural alignment program based on the Sankoff algorithm for the simultaneous solution of the RNA folding and the alignment problem. `LocARNATE` uses `RNAfold` for structure predictions and hence the same energy parameters as `RNAz` does. `LocARNATE` was called with options `--no-seq --no-struc` to generate global, structural alignments.

Negative instances of the training set were generated by shuffling using `multiperm`[41] v. 0.9.3 if the normalized Shannon entropy of the alignment[31] was less than 0.50. Otherwise, alignments were simulated using `SISSIz`[32] to ensure full randomization for the more diverse alignments where shuffling can become inefficient. The final training set was composed of 10,538 alignments for each the positive and the negative class.

The `RNAz 2.0` SVM classifier uses three features to detect structured noncoding RNAs: (i) the average minimum free energy z-score $\bar{z}$ estimated from a dinucleotide shuffled background, (ii) the SCI and (iii) the normalized Shannon entropy H of the alignment as a measure for the content of evolutionary information.

Consider an alignment $\mathcal{A}$ consisting of N sequences. Let E_x denote the minimum free energy of sequence x, and let μ_x and σ_x be the mean and standard deviation, respectively, of the folding energies of a large number of random sequences of the same length and same dinucleotide composition as x. The averaged z-score of the alignment $\mathcal{A}$ is defined as

$$\bar{z} = \frac{1}{N} \sum_{x \in \mathcal{A}} \frac{E_x - \mu_x}{\sigma_x}$$

The SCI of an alignment is given as the fraction of the consensus folding free energy ($E_{consensus}$) to the average of the folding free energies of the single sequences:

$$SCI = \frac{E_{consensus}}{\frac{1}{N} \sum\limits_{x \in \mathcal{A}} E_x}$$

The normalized Shannon entropy H of an alignment $\mathcal{A}$ of RNA sequences over the alphabet $\Sigma = \{$A, C, G, U, -$\}$ is defined as the sum of the Shannon entropies of the individual columns divided by the length of the alignment denoted by L:

$$H = -\frac{1}{L} \sum_{i}^{L} \sum_{\alpha \in \Sigma} p_\alpha^i \ \log_2 \ p_\alpha^i$$

The probability p_α^i is approximated by the observed frequency of character α in alignment column i (normalized by the number N of sequences in the alignment). All features were scaled to a range of [-1,1]. Standard grid search combined with a 10-fold cross validation was applied to find optimized SVM parameters. Among the models with the best cross-validation accuracy (top 20) we chose the model that showed best performance on an independent test set created the same way as the training set. The output of the final classification SVM is a probability estimate that the input alignment contains thermodynamically stable and/or structurally conserved RNA sequences.

A second, independent, SVM classifier was trained on sequence/structure-based alignments generated by `LocARNATE` using the same procedure.

3. Results

3.1. *Dinucleotide based z-scores*

To estimate the mean and standard deviation of folding energies for mononucleotide shuffled sequences it is feasible to sample uniformly simply by varying variables describing the four mononucleotide frequencies and the length of the sequence on a grid. This approach cannot, however, be extended that easily for dinucleotide shuffled sequences. One has to consider the much larger space of dinucleotide compositions that is occupied by

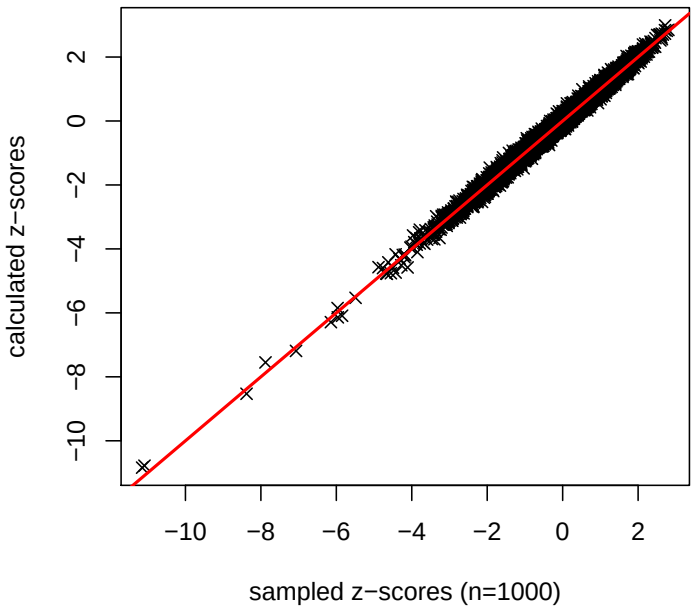


Fig. 2. z-scores calculated by support vector regression in comparison with z-scores determined from 1,000 random samples preserving dinucleotide frequencies for 10,000 randomly drawn sequences from the human ENCODE regions. Correlation of z-scores is 0.996 and the mean absolute error is 0.076.

sequences of practical interest. In this work we use a grid-like approach, where we first apply uniform sampling to cover the mononucleotide space and then choose, for each data point in the grid, a representative set of sequences that covers the dinucleotide space for that particular base composition. However, this procedure still gave more than one million training instances. The training data was split into different ranges of the G+C content to guarantee efficient training and fast prediction. This comes at the price of increased memory consumption but keeps the number of support vectors comparable to the approach used in `RNAz 1.0`. Accuracy of the z-score regression for dinucleotide shuffled sequences was evaluated on 10,000 randomly chosen sequences of variable length from 50 to 200 nt from the human ENCODE regions[34] (Fig. 2) and genomic sequences of *D. melanogaster* and *E. coli*. The mean absolute error (MAE) and the correlation (R) of z-scores calculated by SVM regression compared to z-scores determined from 1,000 random samples is 0.0748 and 0.996, respectively ($n = 30,000$; genomic sequence from ENCODE regions, *D. melanogaster*, and *E. coli*). Comparisons of z-scores determined from 1,000 dinucleotide shuffled sequences to 100 dinucleotide shuffled sequences (MAE= 0.107, $R = 0.992$) and to 1,000 mononucleotide shuffled samples (MAE= 0.420, $R = 0.916$) clearly demonstrate that our method is a suitable approach for fast and efficient estimation of dinucleotide controlled z-scores. `RNAz 1.0` also showed restrictions on the base composition because of the training range of the SVR. This limitation is now overcome by explicit generation of shuffled sequences once the base composition of a sequence is out of the training range. Since boundaries have been chosen broadly (e.g. G+C content from 20 to 80%) this will only apply in a small minority of cases.

3.2. *New training sets and improved classification model*

Since the postulation of the SCI, it has been a major point of criticism that the SCI evaluates structural conservation on the energy level rather than on the RNA structures themselves. However, in previous study[31] it has been shown that the SCI is on average the most powerful method and that it is only outperformed by

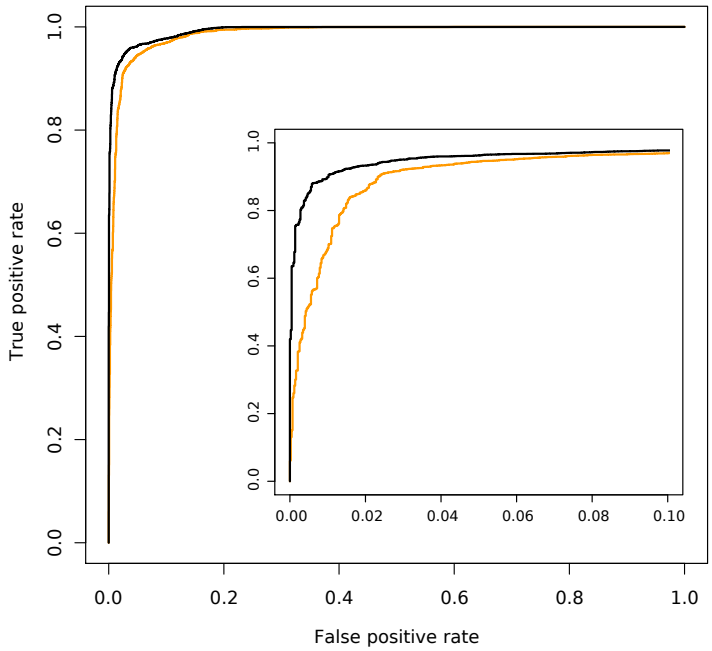


Fig. 3. Accuracy of `RNAz 2.0` classification (black) vs. `RNAz 1.0` classification (orange) on a previously published data set for the evaluation of noncoding RNA gene finders.[32] The positive instance data set consists of 4,303 alignments of structural RNA families (5S ribosomal RNA, U2 spliceosomal RNA, tRNA, Hammerhead ribozyme, U3 snoRNA, U5 spliceosomal RNA, Group II catalytic intron, and Mir-10 microRNA) with two to six sequences per alignment. The negative instance data set consists of 4,303 alignments taken from random genomic location, which resemble approximately the same dinucleotide composition and conservation degree as the positive set. The inset shows the region of high specificity were `RNAz 2.0` clearly outperforms the old version.

other approaches in the high sequence identity range. Attempts to use other conservation measure methods than the SCI, however, failed to give results of comparable quality (data not shown).

To use the SCI for efficient classification one has to take into account the average pairwise identity and the number of sequences as well. Due to the lack of comparative data at the time of training of the initial `RNAz` algorithm limits on these two descriptors were rather arbitrarily chosen. In this work we generated a new training set covering a broader range of RNA families and evaluate sequence variation in terms of the normalized Shannon entropy which has been shown to combine both sequence variation and the number of sequences into one measure.[31] This does not only result in dimensionality reduction of the final classification model, but also overcomes the need to set an upper boundary to the number of sequences in an alignment.

The new `RNAz 2.0` algorithm now uses the average z-score of the sequences in the alignment based on a dinucleotide background model, the SCI and the normalized Shannon entropy as features in the final classification model. To evaluate the predictive power of `RNAz 2.0` we chose a test set used in a previous study.[32] This test set is especially well suited as it contains randomly chosen genomic regions from vertebrate alignments as negative controls. The background dinucleotide content in vertebrate genomes is known to be the main reason for false positive calls in `RNAz 1.0`.[33] Although both versions perform well on this test set, `RNAz 2.0` clearly outperforms version `1.0` in the high specificity range (Fig. 3). For example, at the generally used 0.01 false-positive cutoff, `RNAz 2.0` shows 0.899 sensitivity compared to 0.688 in the old version.

It is a well known fact that sequence-based alignment methods fail to give high quality alignments regarding RNA secondary structures in low average pairwise identity ranges. By using structural alignments one can expect an improvement in discrimination capability of the SCI for alignments with low sequence

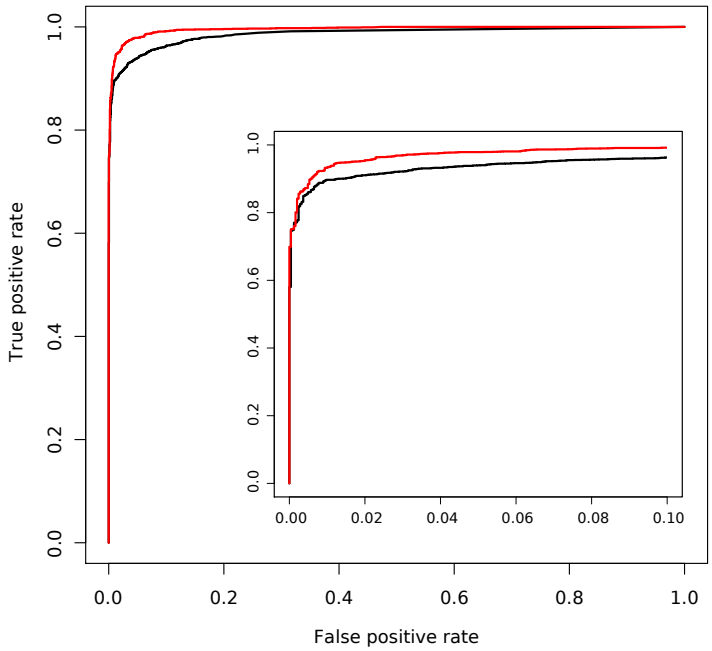


Fig. 4. ROC curves for the RNAz 2.0 prediction accuracy on sequence-based alignments (black) vs. structural alignments (red). A significant improve of the overall predictive power of RNAz 2.0 is achieved by use of structural alignments. The test set is composed of 2,455 alignments of various ncRNA families with an average pairwise identity between 30 and 70%, as well as a negative set consisting of 2,455 alignments derived by randomization of reference alignments with multiperm or SISSIz as described in section 2.2. Sequence-based alignments were generated with ClustalW, while structural alignments were generated with LocARNATE.

similarity.[11] Therefore, we trained a separate SVM decision model based on sequence/structure alignments, similar to the approach used in RSSVM.[30] Structural alignments were generated using LocARNATE, a multiple alignment variant of LocARNA.[46] As depicted in Fig. 4 structural alignments improve the overall predictive power of RNAz.

Recent studies (e.g. Washietl *et al.*[33]) have shown that RNAz suffers from a high false discovery rate (FDR). We therefore evaluated the performance of both versions for the human ENCODE regions. 17-way MAF alignments based on the human genome assembly hg.17 were downloaded from the UCSC genome browser. In total we screened 193,634 MAF alignments derived by pre-filtering with rnazWindow.pl with standard options (window length is 120 nt, step size is 40 nt, average pairwise identity the resulting alignment is optimized to is 80%, and at most six sequences are allowed). Both reading directions were considered in our analysis. A dinucleotide background model was generated with SISSIz[32] and all hits detected by RNAz on this data set were considered to be false positives. Results are summarized in Tab. 1. While RNAz 1.0 shows a very high FDR of around 80%, the FDR of RNAz 2.0 is much lower being around 54% for high confident hits (classification probability > 0.9). It seems noteworthy, that in a previous study[33] the FDR for RNAz 1.0 on ENCODE data was estimated to be around 50%. This estimate was based on a rather simplistic *ad hoc* method to correct for the dinucleotide bias. The new results are based on the more accurate SISSIz null model and demonstrate that RNAz 1.0 is even more affected by the dinucleotide bias than previously assumed. The new version, however, reduces this source of false positives significantly.

To investigate a potential G+C bias of RNAz that was observed for version 1.0,[33] we also trained a classification model that included the G+C content as fourth feature. This additional feature, however, had little impact on the predictions. In particular, the distribution of the G+C content of the positive predictions

Table 1. Comparison of the false discovery rate (FDR) based on ENCODE regions and a dinucleotide background model for low (P > 0.5) and high (P > 0.9) confidence hits. A hit corresponds to a single alignment derived from pre-filtering of ENCODE MAF alignments with `rnazWindow.pl`.

	`RNAz 1.0`		`RNAz 2.0`	
	# low conf.	# high conf.	# low conf.	# high conf.
ENCODE regions	17,814	6,854	6,880	2,259
background	14,489	5,596	4,090	1,219
estimated FDR	81%	82%	59%	54%

remained nearly unchanged (data not shown). This suggests that the elevated G+C content of `RNAz` hits is not an artificial bias, but rather reflects the G+C content of true functional RNAs. Consistent with this observation, the G+C bias of structured RNAs has been used successfully for *de novo* prediction of RNA genes.[43] Preliminary analysis of the ENCODE data showed that the effect is smaller for `RNAz 2.0` than in the earlier version.

3.3. *Computational speed*

The performance of `RNAz 2.0` in comparison to `RNAz 1.0` was benchmarked on 50,000 randomly chosen MAF alignments from the ENCODE data set. Alignment length was 120 nucleotides and alignments contained at most six sequences. Experiments were conducted on an Intel Xeon 2.40GHz CPU. For each alignment both reading directions were examined, resulting in a total of 100,000 alignments that had to be scored. The execution time required by `RNAz 1.0` was 202 min, `RNAz 2.0` with explicit shuffling switched off was 252 min and `RNAz 2.0` using explicit shuffling was 1,230 min. Although explicit shuffling had to be used for only 1% of the sequences (5,524 out of 549,210), it comes with an tremendous overhead increasing the run time of `RNAz 2.0` almost 5-fold. We extracted those alignments where explicit shuffling was used and compared the classification probability to the one derived from calling `RNAz` with option `--no-shuffle` to avoid explicit shuffling. For the vast majority of cases (96%) the change in classification probability was less than 1%. For this data set the maximal observed difference was 0.21. In general, we observed larger differences in the range from 0.2 to 0.8 than in the regions close to 0 or 1.

With option `--no-shuffle`, `RNAz 2.0` has an execution time that is increased by about 25% compared to `RNAz 1.0`.

4. Future directions

In this work we present a major update of the `RNAz` algorithm. Evaluation of thermodynamic stability has been improved by considering a dinucleotide background model. This directly translates into a significantly lower false discovery rate. In addition to the dinucleotide z-score, the overall prediction accuracy is improved by a combination of the use of a new training set and the normalized Shannon entropy as a measure of sequence variation. Furthermore, the updated version is not any more restricted to limitations concerning the base composition or number of sequences in the input alignment.

The generation of structural alignments is computationally expensive but we showed that they can improve the `RNAz` classification power. This is true in particular for alignments of low average pairwise identity. Given that the overall computational complexity of `LocARNATE` is $O(n^4)$, the routine use of structural alignments on a genome-wide scale is still out of reach, at least when off-the shelf hardware is used. In general, it has to be questioned if ncRNA gene finding would benefit from realigning genomic alignments available to date with a structural aligner. These alignments have been generated by means of sequence-only based methods and therefore are not likely to contain homologous RNA sequences that evolve fast on nucleotide level but retain structural conservation. A feasible strategy, however, is the pre-selection of

syntenic regions based on better-conserved flanking regions.[13] Such an approach could be employed for the detection of conserved local structures in the untranslated regions of protein-coding mRNAs, where orthology is established based on similarities of the much better conserved coding sequences. The re-scoring of positively scored hits of a sequence-based `RNAz` screen after re-aligning them with a structural aligner may help to increase the overall accuracy, in particular for relatively poorly conserved alignment slices. One could also use `RNALfold`[45] augmented with the z-score prediction engine of `RNAz` to screen for loci that show signature of increased thermodynamic stability then re-evaluate these loci using structural alignments with `RNAz 2.0` to also account for structural conservation.

An open question, not covered by this work, is how to address the growing number of species in genomic alignments. The use of the normalized Shannon entropy helped us to remove the upper limit on the number of sequences in the alignment. Preliminary analysis of `RNAz 2.0` on multiz 44-way, 28-way and 17-way alignments shows, however, that the simple use of more sequences does not necessarily correlated with improved classification power. To a large extent the increased conservation signal is counteracted by increasing levels of alignment errors. Structural variation of the ncRNAs themselves also poses technical challenges. To-date, an algorithm that addresses both possible misalignments and structural variation is still missing.

RNA secondary structure prediction is sensitive to the exact ends of the input sequence. The use of arbitrarily determined alignment windows of fixed width thus introduces noise. This issue will be alleviated in a forthcoming update of `RNAz` that addresses the pre-processing of long genomic alignments. Here, the sliding window approach will be replaced by the systematic use of `RNALalifold`,[47] an algorithm that computes locally stable consensus RNA secondary structures. These are then used to extract alignments of self-contained (sub)structures for `RNAz` scoring.

`RNAz 2.0` was trained on two particular alignment methods, `ClustalW` for sequence-based alignments and `LocARNATE` for structure-based alignments. As `RNAz` uses a machine learning approach, we have to expect some influence of the alignment algorithm since the features passed to the SVM implicitly also incorporate properties of the alignment algorithms themselves. It may thus become necessary to either re-align the input data or to train decision models for alternative alignment methods.

Supplementary material

An Electronic Supplement located at `www.bioinf.uni-leipzig.de/Publications/SUPPLEMENTS/09-026/` compiles a supplemental figure and data sets used in this work.

Acknowledgments

This work has been funded, in part, by the Austrian GEN-AU projects "bioinformatics integration network III" and "noncoding RNA II", the University of Vienna and the German Research Foundation (grants STA 850/7-1 under the auspices of SPP-1258 "Sensory and Regulatory RNAs in Prokaryotes").

References

1. P. P. Amaral *et al.*, *Science* **319**, 5871 (2008).
2. T. Kuwabara *et al.*, *Cell* **116**, 6 (2004).
3. J. Feng *et al.*, *Genes Dev* **20**, 11 (2006).
4. C. A. Espinoza *et al.*, *RNA* **13**, 4 (2007).
5. A. Pagano *et al.*, *PLoS Genet* **3**, 2 (2007).
6. A. Wutz, *Trends Genet* **23**, 9 (2007).
7. J. L. Rinn *et al.*, *Cell* **129**, 7 (2007).
8. P. P. Amaral and J. S. Mattick, *Mamm Genome* **19**, 7-8 (2008).
9. E. Rivas and S. R. Eddy, *BMC Bioinformatics* **2**, (2001).
10. J. S. Pedersen *et al.*, *PLoS Comput Biol* **2**, 4 (2006).
11. A. V. Uzilov *et al.*, *BMC Bioinformatics* **7**, (2006).
12. Z. Yao *et al.*, *Bioinformatics* **22**, 4 (2006).

13. E. Torarinsson *et al.*, *Genome Res* **16**, 7 (2006).
14. S. Washietl *et al.*, *Proc Natl Acad Sci USA* **102**, 7 (2005).
15. S. Washietl *et al.*, *Nat Biotechnol* **23**, 11 (2005).
16. K. Missal *et al.*, *Bioinformatics* **21**, (2005).
17. K. Missal *et al.*, *J Exp Zoolog B Mol Dev Evol* **306**, 4 (2006).
18. D. Rose *et al.*, *BMC Genomics* **8**, (2007).
19. D. Rose *et al.*, *J Bioinform Comput Biol* **6**, 6 (2008).
20. A. M. McGuire and J. E. Galagan, *PLoS One* **7**, 3 (2008).
21. C. Weile *et al.*, *BMC Genomics* **8**, (2007).
22. T. Mourier *et al.*, *Genome Res* **18**, 2 (2008).
23. C. del Val *et al.*, *Mol Microbiol* **66**, 5 (2007).
24. J. Hertel *et al.*, *Bioinformatics* **24**, 2 (2008).
25. J. Hertel *et al.*, *Bioinformatics* **22**, 14 (2006).
26. K. Reiche *et al.*, *Algorithms Mol Biol* **2**, (2007).
27. P. P. Gardner *et al.*, *Nucleic Acids Res* **33**, 8 (2005).
28. P. W. Hsu *et al.*, *Nucleic Acids Res* **33**, (2006).
29. T. Sandmann and S. M. Cohen, *PLoS ONE* **2**, 11 (2007).
30. X. Xu *et al.*, *PLoS Comput. Biol.*, **5**, (2009).
31. A. R. Gruber *et al.*, *BMC Bioinformatics* **9**, (2008).
32. T. Gesell and S. Washietl, *BMC Bioinformatics* **9**, (2008).
33. S. Washietl *et al.*, *Genome Res* **17**, 6 (2007).
34. ENCODE Project Consortium, *Nature* **447**, 7146 (2007).
35. C. Workman and A. Krogh, *Nucleic Acids Res* **27**, 24 (1999).
36. T.Babak *et al.*, *BMC Bioinformatics* **8**, (2007).
37. S. F. Altschul and B. Erickson, *Mol Biol Evol.* **2**, 6 (1985).
38. I. L. Hofacker *et al.*, *Monatsh. Chem.* **125**, (1994).
39. P. P. Gardner *et al.*, *Nucleic Acids Res.* **37**, (2008).
40. W. Otto *et al.*, *Proceedings of the German Conference on Bioinformatics* **P-136**, (2008).
41. P. Anandam *et al.*, *Bioinformatics* **25**, (2009).
42. I. L. Hofacker *et al.*, *J.Mol.Biol.* **319**, (2002).
43. M. M. Meyer *et al.*, *BMC Genomics* **10**, (2009).
44. S. Washietl, *Methods Mol Biol.* **395**, (2007).
45. I. L. Hofacker *et al.*, *Bioinformatics* **20**, (2004).
46. S. Will *et al.*, *PLoS Comput. Biol.*, **3**, (2007).
47. A. F. Bompfünewerer Consortium, *J Exp Zoolog B Mol Dev Evol.* **308**, (2007).

IDENTIFICATION AND CLASSIFICATION OF SMALL RNAS IN TRANSCRIPTOME SEQUENCE DATA

D. LANGENBERGER[1], C.I. BERMUDEZ-SANTANA[1,2], P.F. STADLER[1,3*], S. HOFFMANN[1]

[1] *University Leipzig*
Chair of Bioinformatics & Interdisciplinary Center for Bioinformatics,
Haertelstrasse 16-18,
D-04107 Leipzig, Germany
E-mail: steve@bioinf.uni-leipzig.de

[2] *Department of Biology*
Universidad Nacional de Colombia
Carrera 45, No. 26-85, Edificio Uriel Gutierrez
D.C., Colombia

[3] *Max-Planck-Insitute for Mathematics in Sciences (MPI-MIS)*
Inselstrasse 22,
D-04103 Leipzig, Germany

Current methods for high throughput sequencing (HTS) for the first time offer the opportunity to investigate the entire transcriptome in an essentially unbiased way. In many species, small non-coding RNAs with specific secondary structures constitute a significant part of the transcriptome. Some of these RNA classes, in particular microRNAs and snoRNAs, undergo maturation processes that lead to the production of shorter RNAs. After mapping the sequences to the reference genome specific patterns of short reads can be observed. These read patterns seem to reflect the processing and thus are specific for the RNA transcripts of which they are derived from. We explore here the potential of short read sequence data in the classification and identification of non-coding RNAs.

Keywords: High throughput sequencing; read patterns; small RNA processing; small RNA classification; machine learning

1. Introduction

Whole-Transcriptome analysis of many species and cell types reveals massive expression of non-coding RNA. It is widely believed that non-coding RNAs act as regulators upon transcription and translation. Recent investigations of whole RNA cDNA-Libraries based on high throughput sequencing (HTS) have shown that these libraries contain both primary and processed transcripts. Over the last years, several classes of small RNAs with a length of about 20nt have been discovered. The most prominent classes are miRNAs, piRNAs, and various variants of endogenous siRNAs.[1,2] In addition, small RNAs have been found to be associated with transcription start and stop sites of mRNAs.[3–5] Several studies reported that well-known ncRNA loci are also processed to give rise to small RNAs. MicroRNA precursor hairpins, for instance, are frequently processed to produce additional "off-set RNAs" (moRNAs) that appear to function like mature miRs. These moRNAs were discovered in *Ciona intestinalis*,[6] where they form an abundant class of processing products. At much lower expression levels they can also be found in the human transcriptome.[7] Specific cleavage and processing of tRNAs was observed in the fungus *Aspergillus fumigatus*[8] and later also found in human short read sequencing data.[9] Small nucleolar RNAs (snoRNAs) are also widely used as a source for specific miRNA-like short RNAs.[10–12] The same holds true for vault RNAs.[13,14] Little is known, however, about the mechanisms of these processing steps and their regulation. Here, we show that the production of short RNAs is correlated with RNA secondary structure and therefore exhibits features that are characteristic for individual ncRNA classes. The specific patterns of mapped HTS reads thus may be suitable to identify and

*P.F.S has external affiliations with the Fraunhofer Institute IZI, the Institute for Chemistry of the University of Vienna, and the Santa Fe Institute.

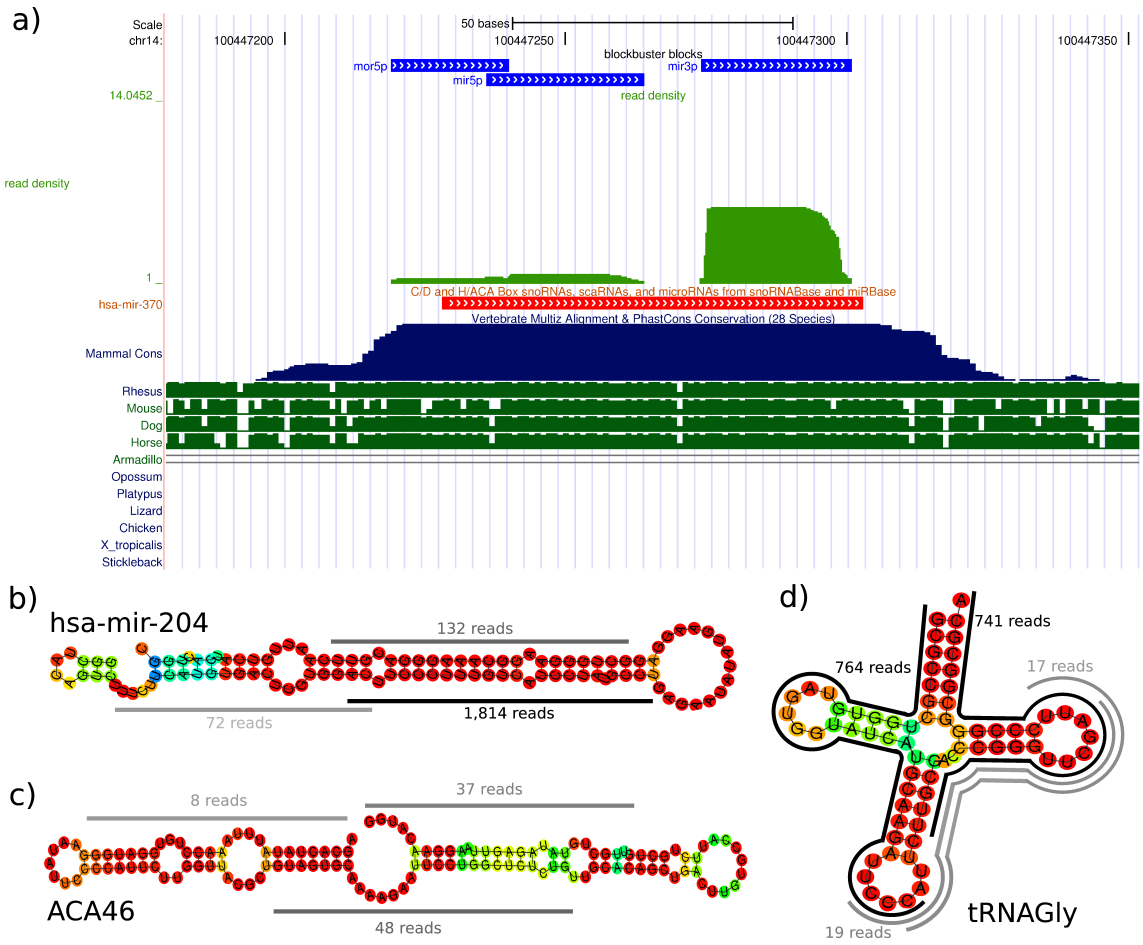


Fig. 1. Non-coding RNAs exhibit specific block patterns. (a) Distribution of short reads at the hsa-mir-370 locus. There are three clearly distinct blocks of reads: they correspond to moR (5'-end), miR* (center) and miR (3'-end) transcripts. The conservation pattern is shown below. (b) The class of miRNAs often shows a block pattern of two or three separated blocks. (c) snoRNAs tend to have miRNA-like mature and star blocks at their 5' and 3' hairpins with minor overlaps, while a series of overlapping blocks is striking for the tRNA class (d).

classify the ncRNAs from which they are processed. We explore here to what extent such an approach is feasible in practise.

The first step towards this goal is the identification of ncRNA loci from a collection of mapped HTS reads. We have recently developed the tool `blockbuster`[7] to simplify this task in genome-wide analyses. The program merges mapped HTS reads into *blocks* based on their location in the reference genome (Fig. 1a). After the assembly of blocks, specific block patterns for several ncRNA classes can be observed. For example, miRNAs typically show 2 blocks corresponding to the miR and miR* positions (Fig. 1b). A similar processing can be observed for snoRNAs (Fig. 1c). On the other hand, tRNAs show more complex block patterns with several overlapping blocks (Fig. 1d).

2. Methods

The dataset analyzed here was produced according to standard small RNA transcriptome sequencing protocols in the context of other projects and will be published in that context. In brief, total RNA was isolated from the frozen prefrontal cortex tissue using the TRIzol (Invitrogen, USA) protocol with no modifications. Low molecular weight RNA was isolated, ligated to the adapters, amplified, and sequenced following the

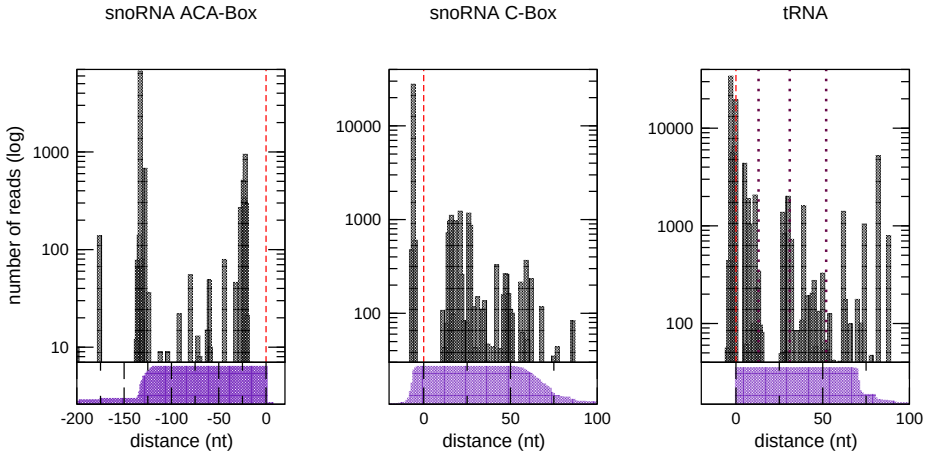


Fig. 2. HTS data reflects structural properties of ncRNAs. Upper panels show the number of 5'-ends of mapped HTS reads (bars) relative to aligned the 5'-ends (dashed vertical lines) of 27 ACA boxes (left), 81 CD boxes (middle) and 87 tRNAs (right). The area in the lower panel represents the number of boxes and tRNAs present at the distance relative to their aligned start sites. In accordance with Taft et al.[12] a sudden and sharp increase of 5'-ends is seen just upstream of the snoRNAs' ACA and C boxes, resp., indicating that read blocks reflect structural properties of snoRNAs. Similarily, the number of 5'-ends increases just upstream of the tRNA and the relative start sites of its three loop regions (dotted lines). Downstream the start sites there is a sudden drop in the number of reads.

Small RNA Preparation Protocol (Illumina, USA) with no modifications. All small RNAs, 17-28nt long, were mapped to the human genome (NCBI36.50 Release of July 2008) using `segemehl`,[15] a method based on a variant of enhanced suffix arrays that efficiently deals with both mismatches as well as insertions and deletions. We required small RNAs to map with an accuracy of at least 80% and only the best hit was selected. Reads mapping multiple times to the genome with an equivalent accuracy were discarded. After filtering the effective accuracy was $> 97\%$. Subsequently, all hits were sorted by their genomic position. Two reads were assigned to the same putative ncRNA locus, i.e. cluster, if separated by less than 100nt. Clusters consisting of less than 10 reads were discarded because of their low information content.

To detect specific expression patterns, we divided consecutive reads into blocks using `blockbuster`.[7] Here, we used a width parameter of $s = 0.5$, a value that requires blocks to be well separated to be recognizedas distinct. We required a cluster to have at least 2 blocks. In the following we refer to the number of reads comprised in a block as the *block height*. Using `blockbuster`, we identified 852 clusters across the whole human genome. This set comprises 2,538 individual blocks and 85,459 unique reads. 434 clusters were found within annotated ncRNA loci [`miRBase` v12 (727 entries), `tRNAscan-SE` (588 entries) and `snoRNAbase` v3 (451 entries)], see Tab. 1.

We then computed secondary structures (using `RNAfold`[16]) to assess the relationship of reads and structure. For each read, the base pairing probabilities were calculated for the sequences composed of the read itself and 50nt of flanking region both up- and downstream. These data were also collected separately for reads found within annotated miRNA, tRNA, and snoRNA loci, respectively.

In order to investigate whether the short reads patterns carry information on the particular ncRNA class from which they orginate, we selected three distinct ncRNA classes and performed a random forest

Table 1. In total 434 of 852 clusters were found within regions of annotated miRNA, tRNA and snoRNA loci. While the average number of blocks is similar for all three ncRNA classes, the number of reads differs significantly among the classes.

RNA class	source	loci found	blocks/cluster (mean)	reads/cluster (median)
microRNAs	miRBase v12	218	2.42 ± 1.04	4535.33
tRNAs	tRNAscan SE	87	3.22 ± 1.92	183.95
snoRNAs	snoRNAbase v3	129	2.60 ± 1.66	127.5

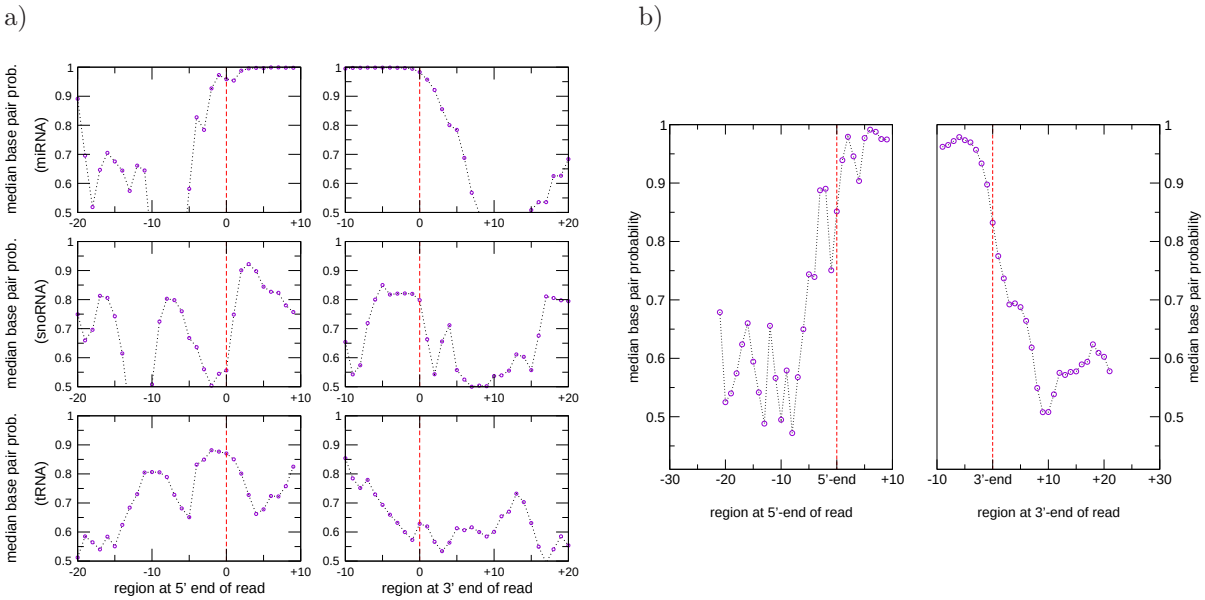


Fig. 3. Base pairing probabilities increase at the 5'-end and decrease at the 3'-end of reads mapped to ncRNA loci. (a) The 3'- and 5'-ends are indicated by dashed lines. The median base pairing probability increases sharply at the 5'-ends (upper left) and drops again at the 3'-ends of reads mapped to miRNA loci (upper right). A similar – but attentuated – effect is observed for snoRNAs (middle panel) and tRNAs (lower panel). (b) The median base pairing probabilities at 5'- (left panel) and 3'- ends (right panel) for all reads within the 852 clusters. The 5'- and 3'-ends are indicated by dashed vertical lines.

classification:[17,18] tRNAs ($n = 87$), miRNAs ($n = 218$) and snoRNAs ($n = 129$). Based on a visual inspection of the mapped reads, ten features were selected to train the random forest model: the number of blocks within a cluster (blocks), the length of a cluster (length), the number of nucleotides covered by at least two blocks (nt overlap), the number of overlapping blocks (block overlap), the maximum, minimum and the mean block height (max, min and mean height) in a cluster as well as the maximum, minimum and the mean distance between consecutive blocks (max, min and mean distance).

3. Results

The 5'-ends of reads arising from known snoRNAs preferentially map just upstream of the C- and ACA-boxes. This indicates the correlation of mapping patterns with processing steps and thus with structural properties of snoRNAs (Fig. 2). Based on earlier findings that miRNA-like products are derived from snoRNAs[12] and the observation that miRNA transcripts tend to have higher blocks (Tab. 1), the two peaks shown in the Figure 2 (left) probably represent small RNAs produced from the 5'- and 3'-hairpins of the HACA (see also Fig. 1c). CD-snoRNAs show, in contrast to the HACA-snoRNAs, only a single prominent peak at the 5'-end (Fig. 2, middle). An increased number of 5'-ends of HTS reads is also observed just upstream of loops of tRNAs (Fig. 2 (right)).

The pairing probabilities of bases covered by HTS reads are significantly increased (Fig. 3b). Just upstream the 5'-end of these reads, the median base pairing probability increases sharply and reaches a level of > 0.9. At the 3'-end the base pairing probability drops again. However, median base pairing probabilities of bases covered by the center of reads drop down to 70%. Although this effect is boosted by reads found within miRNA loci, it can also be observed unambiguously for reads within snoRNA and tRNA loci (Fig. 3a).

The observation that blocks reflect structural properties of ncRNAs was exploited to train a random forest classifier to automatically detect miRNAs, tRNAs and snoRNAs. After visual inspection of block patterns for some representatives of these classes, ten features were selected. Their evaluation reveals significant statistical differences among the chosen ncRNA classes (Fig. 4). As expected, the number of reads mapped to miRNA

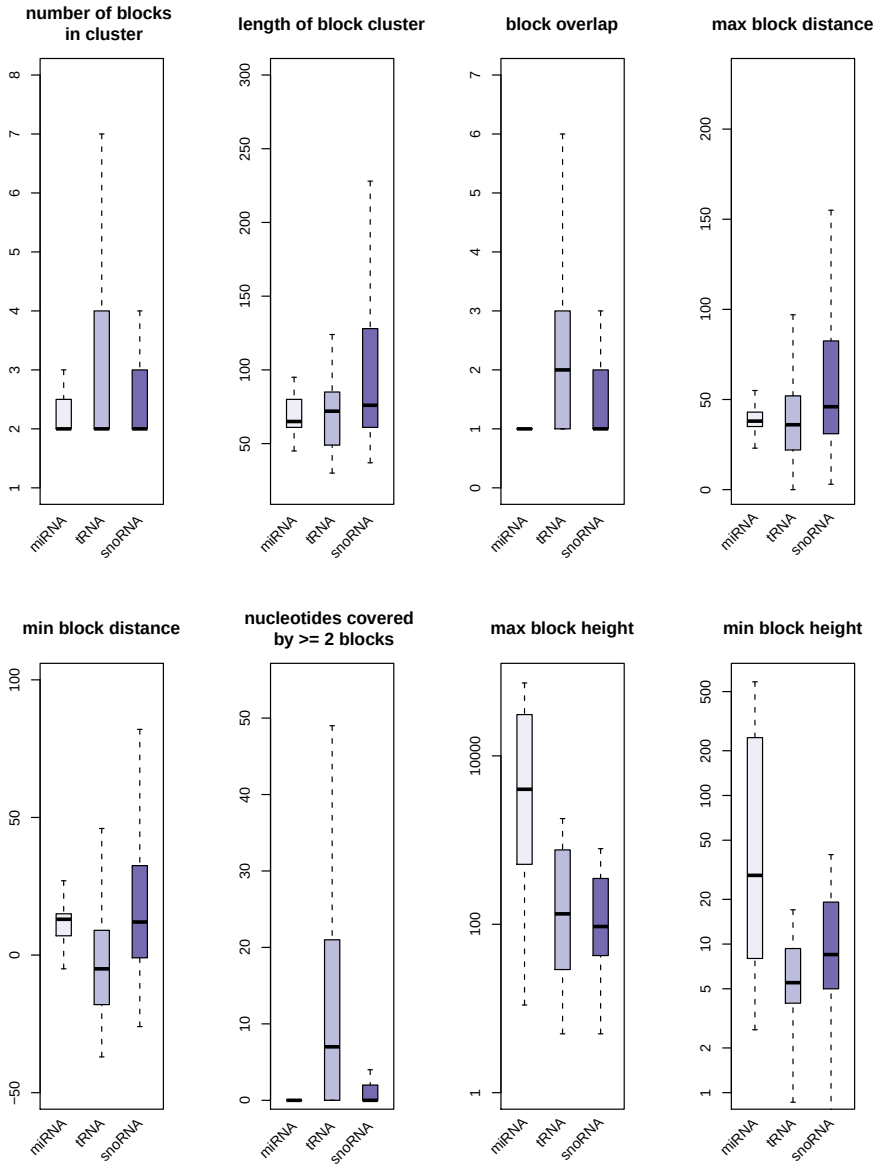


Fig. 4. Box plots for 8 different features selected to train the random forest classifier. The number of reads mapped to miRNA loci alone (max block height and min block height) effectively distinguish miRNAs from other ncRNAs. Likewise, the distribution of block distances seems to be a specific feature for miRNAs. Compared to other regions, tRNA loci frequently show block overlaps of two or more blocks. The minimum block distance shows a median overlap of ≈5nt for blocks in within tRNA loci. SnoRNAs typically have longer block distances than the other classes.

loci (minimum and maximum block height) clearly distinguishes miRNAs from other ncRNA classes. In contrast to tRNAs and snoRNAs the maximum block distance of miRNAs shows a very narrow distribution around 40nt, reflecting the distance between miR and miR* transcripts. Furthermore, the class of tRNAs frequently shows more block overlaps than snoRNAs and miRNAs. The distance of blocks is an important feature for snoRNAs: the maximum block as well as the minimum block distance is higher compared to both tRNAs and miRNAs.

The random forest model was repeatedly trained with randomly chosen annotated loci and different training set sizes in order to determine predictive values (PPV) and recall rates. For the training sets

Table 2. Positive predictive values (PPV) and recall rates for training sets of size 150 and 250. For each set size means, medians and standard deviations are calculated from 20 randomly sampled training sets.

		PPV		recall	
	#loci	mean	sdev	mean	sdev
Training size 250					
all	852	0.889	0.015	0.799	0.015
miRNA	227	0.932	0.020	0.918	0.023
tRNA	287	0.860	0.040	0.683	0.046
snoRNA	143	0.819	0.032	0.694	0.060
other	195				
Training size 150					
all	852	0.827	0.020	0.698	0.027
miRNA	236	0.900	0.027	0.847	0.041
tRNA	348	0.755	0.044	0.580	0.062
snoRNA	115	0.733	0.057	0.525	0.071
other	153				

comprising 150 clusters the random forest model shows a positive predictive value > 0.7 for all three ncRNA classes. The recall rate for miRNAs is well above 80%. However, with a rate of ≈ 0.55 the recall of snoRNAs and tRNAs is relatively poor (Tab. 2). For larger training sets containing 250 clusters, the positive predictive value (PPV) is > 0.8 for all classes. For miRNAs the classification achieves recall rates and PPVs of > 0.9. Likewise, the recall rates for snoRNAs and tRNAs rise to 0.7-level. In summary, for both training set sizes and all classes the random forest model achieves PPVs and recall rates of ≈ 0.8.

We applied the classifier to unannotated ncRNA loci. A list of miRNA, snoRNA, and tRNA candidates predicted is available from the supplementary page (`http://www.bioinf.uni-leipzig.de/~david/PSB/`). This resource includes the original reads, their mapping accuracy and their mapping location in machine-readable formats. Furthermore, the page provides links to the UCSC genome browser to visualize the block patterns. For microRNAs and snoRNAs, we also indicate whether the candidates are supported by independent ncRNA prediction tools.

The 29 miRNA predictions contained 3 miRNAs (hsa-mir1978, hsa-mir-2110, hsa-mir-1974) which have already been annotated in the most recent miRBase release (v.14), as well as a novel member of the mir-548 family, and another locus is the human ortholog of the bovine mir-2355. In addition, we found two clusters antisense to annotated miRNA loci (hsa-mir-219-2 and hsa-mir-625). Such antisense transcripts at known miRNA loci have been reported also in several previous publications,[20–22,24] lending further credibility to these predictions.

For the tRNAs and snoRNAs we expect a rather large false positive rate. The 78 tRNA predictions are indeed contaminated by rRNA fragments, but also contain interesting loci, such as sequence on Chr.10 that is identical with the mitochondrial tRNA-Ser. `SnoReport`,[23] a specific predictor for HACA snoRNAs based on sequence and secondary features, recognizes 44 (20%) of our 223 snoRNAs predictions.

Short RNAs are processed from virtually all structured ncRNAs. Complex read patterns are observed, for instance, for the 7SL (SRP) RNA and the U2 snRNA. Y RNAs, which have a panhandle-like secondary structure produce short reads mostly from their 5' and 3' ends, see Fig. 5.

4. Discussion

In extension of previous work establishing that various ncRNA families produce short processing products of defined length,[6,9,12] we show here that these short RNAs are generated from highly specific loci. The dominating majority of reads from short RNAs originates from base paired regions, suggesting that these RNAs are, like miRNAs, produced by Dicer or other specific RNAases. For example, specific cleavage products have recently been reported for tRNAs.[19] In this work we show that the block patterns are characteristic for three different ncRNA classes and thus suitable to recognize additional members of these classes. For

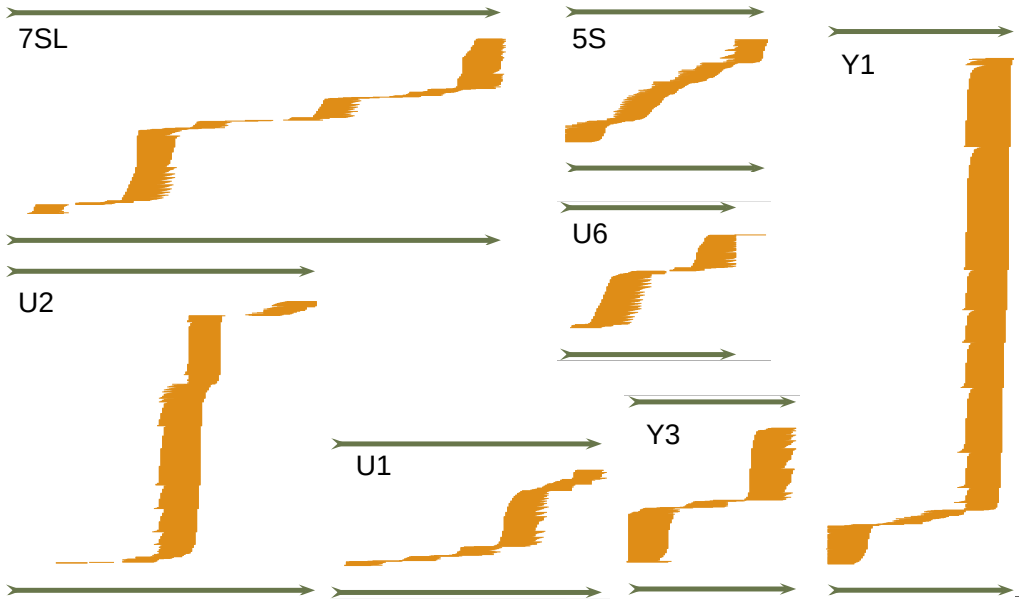


Fig. 5. Short reads are produced from a wide variety of structured ncRNAs. Green arrows indicate the ncRNA gene and its reading direction, individual short reads are shown as orange lines. The same scale is used for all examples.

instance, the random forest trained with loci annotated in the mirBase v12 predicted five additional miRNAs reported in the mirBase release 14 as well as two "antisense microRNA".

The block patterns for the evaluated ncRNAs show some interesting characteristics. Although miRNA loci accumulate far more reads than tRNAs and snoRNA loci, the reads are extremely unevenly distributed across the blocks. For tRNAs we observe series of overlapping blocks that are specific enough to separate this class from other classes with high positive predictive values.

However, the successful prediction of miRNAs heavily depends on the height of the blocks, i.e. the number of reads that map to a potential locus. In comparision tRNAs and snoRNAs show significantly lower positive predictive values and recall rates. A relatively large training set is required to achieve PPV's $> 80\%$. Obviously, the selection of appropriate features is crucial for the success of the presented approach. Hence, the random forest classifier is not sufficient as it stands and the identification of other characteristic features is subject to further research. The integration of secondary structure information of cluster regions is likely to enhance the prediction quality.

Beyond the classification by means of soft computing methods, this survey shows that HTS block patterns bear the potential to greatly improve and simplify ncRNA annotation. Given the striking relationship of HTS reads and secondary structure for some ncRNA classes, block patterns may also be used in the future to directly infer secondary structure properties of non-coding RNAs from transcriptome sequencing data. In this context, although not shown here, block patterns may also help to identify new classes of RNAs directly from transcriptome sequencing data.

5. Acknowledgements

This work was supported by the European Union (EDEN, contract 043251), the Deutsche Forschungsgemeinschaft (SPP-1174) (P.F.S), a PhD scolarship of the DAAD-AleCol program (C.B-S.), a formel.1 grant of the Medical Faculty of the University of Leipzig and the Freistaat Sachsen under the auspices of the LIFE project.

References

1. D. Moazed. *Nature* **457**, 413-420 (2009).
2. A. Tanzer *et al.* in *Evolutionary Genomics*, G. Caetano-Anolles (ed.), **in press** (2009).
3. P. Kapranov *et al.*, *Science* **316**, 1484-1488 (2007).
4. R.J. Taft *et al.*, *Nat. Genetics* **41**, 572-578 (2009).
5. R.J. Taft *et al.*, *Cell Cycle* **8**, 2332-2338 (2009).
6. W. Shi *et al.*, *Nat. Struct. Mol. Bio.* **16**, 183-189 (2009).
7. D. Langenberger *et al.*, *Bioinformatics* 10.1093/bioinformatics/btp-419, (2009).
8. C. Jöchl *et al. Nucleic Acids Res.* **36**, 2677-2689 (2008).
9. H. Kawaji *et al.*, *BMC Genomics* **9**, 157 (2008).
10. C. Ender *et al.*, *Mol. Cell* **32**, 519-528 (2008).
11. A. Saraiya *et al.*, *PLoS Pathog.* **4**, e1000224 (2009).
12. R.J. Taft *et al.*, *RNA* **15**, 1233-1240 (2009).
13. P.F. Stadler *et al.*, *Mol. Biol. Evol.* **26**, 1975-1991 (2009).
14. H. Persson *et al.*, *Nat. Cell. Biol.* doi:10.1038/ncb1972 (2009).
15. S. Hoffmann *et al.*, *PLoS Comp. Biol.* **5**, e1000502 (2009).
16. I.L. Hofacker *et al.*, *Bioinformatics* **22**, 1172-1176 (2006).
17. I. Witten *Data Mining: practical machine learning tools and techniques*, 2nd edn., (2005).
18. L. Breiman, *Machine Learning* **45**, 5-32 (2001).
19. D.M. Thompson *et al.*, *Cell* **138**, 215-219 (2009).
20. E.A. Glazov *et al.*, *PLoS One* **4**, e6349 (2009).
21. A. Stark *et al.*, *Genes & Development* **22**, 8-13 (2008).
22. W. Bender *Genes & Development* **22**, 14-19 (2008).
23. J. Hertel *et al.*, *Bioinformatics* **24**, 158-164 (2008).
24. D.M. Tyler *Genes & Development* **22**, 26-36 (2008).

IMPROVEMENT OF STRUCTURE CONSERVATION INDEX WITH CENTROID ESTIMATORS

YOHEI OKADA

Department of Biosciences and Informatics, Keio University,
3–14–1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa 223–8522, Japan
E-mail: okada@dna.bio.keio.ac.jp

KENGO SATO

Japan Biological Informatics Consortium (JBIC),
2–45 Aomi, Koto-ku, Tokyo 135–8073, Japan
E-mail: sato-kengo@aist.go.jp

YASUBUMI SAKAKIBARA

Department of Biosciences and Informatics, Keio University,
3–14–1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa 223–8522, Japan
E-mail: yasu@bio.keio.ac.jp

RNAz, a support vector machine (SVM) approach for identifying functional non-coding RNAs (ncRNAs), has been proven to be one of the most accurate tools for this goal. Among the measurements used in RNAz, the Structure Conservation Index (SCI) which evaluates the evolutionary conservation of RNA secondary structures in terms of folding energies, has been reported to have an extremely high discrimination capability. However, for practical use of RNAz on the genome-wide search, a relatively high false discovery rate has unfortunately been estimated. It is conceivable that multiple alignments produced by a standard aligner that does not consider any secondary structures are not suitable for identifying ncRNAs in some cases and incur high false discovery rate. In this study, we propose C-SCI, an improved measurement based on the SCI applying γ-centroid estimators to incorporate the robustness against low quality multiple alignments. Our experiments show that the C-SCI achieves higher accuracy than the original SCI for not only human-curated structural alignments but also low quality alignments produced by CLUSTAL W. Furthermore, the accuracy of the C-SCI on CLUSTAL W alignments is comparable with that of the original SCI on structural alignments generated with RAF for which 4.7-fold expensive computational time is required on average.

Keywords: structure conservation index; centroid estimators; non-coding RNAs

1. Introduction

Many studies have recently discovered essential roles of non-protein-coding functional RNAs (ncRNAs) in cells such as translation, post-transcriptional gene regulation and maturation of rRNAs, tRNAs and mRNAs.[1,2] Therefore, to identify ncRNAs in genomes and analyze their functions is a crucial task for not only molecular cell biology but also bioinformatics.

It is well-known that such biological functions of ncRNAs are deeply related to their secondary structures which consist of hydrogen-bonded base-pairs including the Watson-Crick base-pairs (`A-U` and `G-C`), the wobble base-pairs (`G-U`) and other non-canonical base-pairs. These base-pairs stabilize the structure of RNAs in terms of the free energy. Thus, the secondary structure with the minimum free energy (MFE) has been regarded as the most reliable prediction of RNA secondary structures.

However, MFE alone could not be an appropriate measure for identifying ncRNAs since the free energy is heavily biased by the nucleotide composition.[3] Therefore, several comparative approaches for identifying ncRNAs have been proposed.[4–10] For this purpose, Washietl *et al.* have developed RNAz which uses a support vector machine (SVM) approach, and have proposed the Structure Conservation Index (SCI) as a feature to measure the evolutionary conservation in terms of secondary structures.[7] Assuming that MFE for the consensus secondary structure is close to that for each sequence if a given multiple alignment is structurally conserved, the SCI is defined as the rate of MFE for the common secondary structure to averaged MFE for

each sequence. MFEs for each sequence and the common secondary structure are calculated by RNAfold and RNAalifold both part of the Vienna RNA packages,[11,12] respectively.

RNAz with the SCI has been proven to be the one of the most accurate tools for identifying ncRNAs.[7,13] However, for practical use of RNAz on the genome-wide search, a relatively high false discovery rate has unfortunately been estimated.[14] It is conceivable that multiple alignments produced by a standard aligner that does not consider any secondary structures are not suitable for identifying ncRNAs in some cases and incur high false discovery rate. Wang *et al.* have also suggested that the genome-wide alignments in the UCSC Genome Browser[15] produced by MULTIZ[16] should be improved in some regions for identifying ncRNAs.[17] To improve the accuracy, two strategies can be considered: the one is to employ a structural aligner such as RAF[18] to produce high quality alignments, and the other is to develop a more robust method against low quality alignments. Since the former strategy will consume impractical execution time for structural alignments, this study takes the latter strategy.

Recently, CentroidFold which employs γ-centroid estimators for predicting RNA secondary structures has been developed, and has been shown to be more accurate than other existing tools.[9] Especially, CentroidFold can predict much more accurate common secondary structures for low quality multiple alignments produced by CLUSTAL W[19] than RNAalifold.

In this study, we propose C-SCI, an improved measurement based on the SCI applying γ-centroid estimators for (common) secondary structure prediction, instead of RNAfold and RNAalifold, to incorporate the robustness against low quality multiple alignments. Our experiments show that the C-SCI achieves higher accuracy than the original SCI for not only human-curated structural alignments but also low quality alignments produced by CLUSTAL W. Furthermore, the accuracy of the C-SCI on CLUSTAL W alignments is comparable with that of the original SCI on RAF alignments for which 4.7-fold expensive computational time is required on average.

2. Method

2.1. *Structure Conservation Index*

The Structure Conservation Index (SCI) evaluates secondary structure conservation of a given multiple alignment of RNAs in terms of the minimum free energy (MFE). We denote with $\mathcal{S}(x)$ the entire folding space of a single sequence x and denote with $\mathcal{S}(A)$ the entire consensus folding space of an alignment A. The SCI is defined as

$$SCI(A) = \frac{E_{Align}(y_A^{MFE})}{\frac{1}{\#A}\sum_{x\in A} E(y_x^{MFE})}, \tag{1}$$

where $\#A$ is the number of sequences in the alignment A. For a single sequence x, $E(y)$ denotes the free energy of a secondary structure $y \in \mathcal{S}(x)$, and $y_x^{MFE} = \arg\min_{y\in\mathcal{S}(x)} E(y)$ is defined to be the MFE structure of x calculated by RNAfold.[11] Similarly, for an alignment A, $E_{Align}(y)$ is the free energy of a consensus structure $y \in \mathcal{S}(A)$, and $y_A^{MFE} = \arg\min_{y\in\mathcal{S}(A)} E_{Align}(A)$ is the consensus MFE structure of A calculated by RNAalifold.[12] The free energy of a consensus structure is defined as the average of the energy contributions of the single sequences plus covariance scores for bonuses of compensatory and consistent co-mutation in the alignment.

The consensus MFE alone could be used to identify functional RNAs likelihood of functional RNAs in terms of thermodynamic stability of consensus folded structures. However, it is difficult to make straightforward use of it, since the folding energy is heavily biased by the nucleotide composition and the length of the alignment. The SCI solved this problem by normalizing $E_{Align}(y_A^{MFE})$ with the average of $E(y_x^{MFE})$ for all $x \in A$. From a different view, the SCI reflects the idea that for a well-conserved alignment the structure of each sequence resembles each other and the consensus structure resembles all of them, so $E_{Align}(y_A^{MFE})$ would have as low value as $E(y_x^{MFE})$, otherwise $E_{Align}(y_A^{MFE})$ would not. The SCI is near 0 for an alignment that is not structurally conserved, whereas the SCI is near 1 or above for an alignment that is structurally

conserved. Especially, if the alignment is structurally well-conserved and compensatory and consistent mutation often occurs, the SCI may be above 1.

As shown in the definition (1) of the SCI, the SCI obviously depends on the accuracy of common secondary structure prediction, which is also deeply influenced by the quality of multiple alignments of RNAs. This fact is supported by a previous study[20] and our results shown in Sec. 3. For the genome-wide search, high quality alignments that consider RNA secondary structures cannot be obtained easily due to the computational cost for calculating structural alignments. Therefore, a robust method that does not require high quality alignments is required.

2.2. *γ-Centroid Estimator*

CentroidFold implements a γ-centroid estimator which predicts secondary structures with the maximum expected accuracy by a kind of posterior decoding methods on the base-pairing probability matrix. CentroidFold employs a gain function between a true secondary structure θ and a predicted secondary structure y on x defined as

$$G_\gamma(\theta, y) = \sum_{1 \le i < j \le |x|} \{\gamma I(y_{ij} = 1) I(\theta_{ij} = 1) + I(y_{ij} = 0) I(\theta_{ij} = 0)\}, \tag{2}$$

where γ is a weight for base-pairs, y_{ij} is 1 if the i-th and j-th nucleotides form a base-pair in y, and $I(\mathit{condition})$ is the indicator function, which takes 1 or 0 relying on whether *condition* is true or false. The gain function (2) is equal to the weighted sum of the number of true positives and the number of true negatives of base-pairs. CentroidFold predicts a secondary structure $y \in \mathcal{S}(x)$ which maximizes the expectation of $G_\gamma(\theta, y)$ with respect to an ensemble of all possible secondary structure $\mathcal{S}(x)$ which is distributed under a posterior distribution $p(\theta|x)$,

$$\begin{aligned} \mathbb{E}_{p(\theta|x)}[G_\gamma(\theta, y)] &= \sum_{\theta \in \mathcal{S}(x)} G_\gamma(\theta, y) p(\theta|x) \\ &= \sum_{1 \le i < j \le |x|} ((\gamma + 1) p_{ij} - 1) I(y_{ij} = 1) + C, \end{aligned} \tag{3}$$

where C is a constant independent of y, and $p_{ij} = \mathbb{E}_{p(\theta|x)}[\theta_{ij}]$ is the base-pairing probability that the i-th and j-th bases form a base-pair. The optimal secondary structure $\hat{y} = \arg\max_{y \in \mathcal{S}(x)} \mathbb{E}_{p(\theta|x)}[G_\gamma(\theta, y)]$ can be calculated efficiently by using the following DP algorithm:

$$M_{i,j} = \max \begin{cases} M_{i+1,j} \\ M_{i,j-1} \\ M_{i+1,j-1} + (\gamma + 1) p_{ij} - 1 \\ \max_k [M_{i,k-1} + M_{k,j}] \end{cases}, \tag{4}$$

and tracing back from $M_{1,|x|}$ to calculate $\hat{y}$. The model of the posterior distribution $p(\theta|x)$ can be chosen from various implementations including the McCaskill model[21] based on the Boltzmann free energy and the CONTRAfold model[22] based on a machine learning technique.

CentroidFold can also predict a common secondary structure of a multiple alignment of RNAs by using averaged γ-centroid estimators. The optimal common secondary structure which maximizes the sum of the expected gain (2) for all $x \in A$, that is,

$$\hat{y} = \arg \max_{y \in \mathcal{S}(A)} \sum_{x \in A} \mathbb{E}_{p(\theta|x)}[G_\gamma(\theta, y)]$$

can similarly be calculated by using (4) with the averaged base-pairing probability[23] defined as

$$\bar{p}_{ij} = \frac{1}{\#A} \sum_{x \in A} \mathbb{E}_{p(\theta|x)}[\theta_{ij}],$$

instead of p_{ij}.

The weight γ in the definition (2) controls the number of predicted base-pairs, that is, the trade-off between specificity and sensitivity of predicted base-pairs. If $\gamma = 1$, this estimator is equivalent to the centroid estimator.[24,25]

CentroidFold has been shown to be more accurate than other existing tools.[9] Especially, CentroidFold can predict much more accurate common secondary structures than RNAalifold for low quality multiple alignments produced by CLUSTAL W.

2.3. *The C-SCI*

Now, we propose an improved measurement of secondary structure conservation based on the SCI by employing CentroidFold for (common) secondary structure prediction, instead of RNAfold and RNAalifold, to incorporate the robustness against low quality multiple alignments.

At first, we predict the consensus centroid structure for an alignment A, denoted by y_A^C, and centroid structures for each sequence $x \in A$, denoted by y_x^C, by using CentroidFold. For a single sequence, we map a predicted structure onto each sequence x and calculate its free energy $E(y_x^C)$ for all of the sequences. For an alignment, we map a predicted consensus structure onto each sequence x and get rid of gaps and corresponding parts of the structure. In removing a gap, if the part of structure corresponding to the gap is represented as unpaired, the compartment is removed, whereas if the corresponding part is represented as paired, the compartment is removed and its pair is removed or converted to unpaired depending on whether the pair corresponds gap or not. To calculate the energy, we use RNAeval[11] with the predicted structure on the sequence. The free energy of a consensus secondary structure is calculated from the averaged free energy for all sequences and the covariance score which is implemented according to RNAalifold.[12]

Then, the C-SCI is calculated as follows:

$$C\text{-}SCI(A) = \frac{E_{Align}(y_A^C)}{\frac{1}{\#A}\sum_{x \in A} E(y_x^C)}. \tag{5}$$

The C-SCI has two parameters which affect the discrimination capability of the C-SCI. We denote γ_A as the parameter γ for predicting consensus secondary structures on multiple alignments, and γ_S as γ for predicting secondary structures on single sequences. These parameters were determined by 10-fold cross-validation with the grid search on $\gamma \in \{2^k : -10 \leq k \leq 10, k \in \mathbb{Z}\}$ for γ_A and γ_S. The detail of how the 10-fold cross-validation was performed is written in the section 3.1. Furthermore, the C-SCI has a modification that if the predicted structure is unstable and the energy has positive value, the energy is treated as 0. This is because C-SCI may get a high value regardless of secondary structure conservation, if the numerator and the denominator of C-SCI are positive.

3. Result

3.1. *Evaluation*

To confirm the discrimination capability of the C-SCI, we performed the experiments along with the previous study[26] on BRAliBase 2.1 data set,[27] which is constituted with 18,990 reference alignments of 36 RNA families and the same number of the corresponding sets of sequences which are not aligned. Reference alignments included in BRAliBase 2.1 are human-curated alignments which are made from Rfam database[28] aiming for evaluating structural alignments. We also produced multiple alignments using CLUSTAL W[19] version 1.83 with standard settings to investigate the discrimination capability on low quality alignments. For each alignment, we generated negative controls by utilizing `shuffle-aln.pl`.[29] This program shuffles columns of a given alignment to destroy its secondary structure, while maintaining gap patterns, nucleotide compositions and sequence length. We generated five negative controls for each alignment. These alignments were binned according to their normalized Shannon entropy by the size of 0.05. The normalized Shannon entropy is defined as the average of the Shannon entropy for the individual column over all columns in the

Table 1. Detail information about reference alignments.

entropy	number of alignments						average pairwise sequence identity					
	2	3	5	7	10	15	2	3	5	7	10	15
0.10	827	111	11	2	0	0	92.3	93.8	94.7	94.9	—	—
0.15	922	329	48	27	16	6	87.5	90.9	93.1	93.6	93.9	94.0
0.20	974	502	148	50	16	8	83.1	87.4	89.9	91.0	91.7	92.8
0.25	432	479	253	158	58	18	77.6	84.1	87.2	88.3	89.2	89.4
0.30	391	178	262	138	108	65	72.6	80.6	84.6	86.0	87.0	87.7
0.35	456	108	71	95	71	47	67.4	76.6	81.8	83.7	84.8	85.5
0.40	554	134	32	23	16	14	62.6	72.7	78.3	80.6	82.4	83.3
0.45	588	194	48	10	8	3	57.4	68.5	74.2	75.8	79.5	77.8
0.50	559	195	68	38	13	5	52.6	64.5	69.9	71.0	72.1	75.0
0.55	739	194	83	53	27	16	47.5	61.1	65.8	67.0	69.3	69.5
0.60	797	196	82	44	34	20	42.5	58.0	64.1	65.0	66.6	67.2
0.65	589	234	61	21	10	3	37.7	55.1	63.9	62.7	64.2	63.4
0.70	478	320	43	10	5	2	32.5	51.7	61.6	63.1	64.6	63.8
0.75	244	274	39	18	2	1	27.8	48.2	57.9	62.9	59.6	58.9
0.80	126	313	71	17	8	6	22.6	44.3	54.6	56.4	61.6	59.5
0.85	37	326	117	22	11	2	18.2	40.9	53.2	56.1	59.4	60.5
0.90	2	227	139	39	12	2	14.1	37.1	49.9	55.1	56.6	56.2
0.95	0	130	125	68	24	2	—	33.8	46.4	51.2	54.2	58.9
1.00	0	131	168	62	25	13	—	31.0	43.6	49.2	51.8	53.8
1.05	0	41	141	79	46	18	—	29.2	41.6	45.3	49.4	52.2
1.10	0	4	100	99	34	18	—	26.2	39.1	42.7	45.9	51.1
1.15	0	2	61	92	48	25	—	24.0	37.5	40.5	42.7	44.2

In the content of "number of alignments", each column corresponds to the number of alignments constituted with the designated number of sequences (2, 3, 5, 7, 10 or 15 sequences). Similarly in the content of "average pairwise sequence identity", each column means the average of average pairwise sequence identity in the alignments with the designated number of sequences.

alignment whose length is $|A|$,

$$H = -\frac{1}{|A|}\sum_{i=1}^{|A|}\sum_{j\in\Sigma} p_j^i \log_2 p_j^i, \tag{6}$$

where j is in the alphabet $\Sigma = \{\mathtt{A}, \mathtt{U}, \mathtt{G}, \mathtt{C}, \mathtt{-}\}$ constituted with the four nucleotides and the gap character "-", and p_j^i is the probability observing the character j in column i. We used the alignments in the bins from 0.1 to 1.15 of normalized Shannon entropy according to Ref. 26. The number of alignments and the average of averaged pairwise sequence identity (APSI) on reference alignments for each normalized Shannon entropy bin are summarized in Tab. 1. This shows that higher entropy regions tend to include the alignments with lower APSI or with larger number of sequences. Therefore, most of alignments with small number of sequences appear in low entropy region.

To evaluate the performance of the various strategies, we performed the receiver operating characteristic (ROC) curve analysis. An ROC curve is a plot of true positive rate versus false positive rate in varying the discrimination threshold of a classifier. The area under the ROC curve (AUC) is used for evaluation of the discrimination: the shift of AUC to 1 means better discrimination capability. Calculation of AUC for each entropy subset was done by using ROCR package.[30]

In our study, we compared the C-SCI with the original SCI and the measurement "base-pairing distance" (pairwise, consensus), which have been reported to achieve as high AUC as the SCI.[26] Base-pairing distance is a measurement to compare two single structures by using the Hamming distance. Here "pairwise" means the comparison of each structure of a single sequence with each other, and "consensus" means the comparison of each structure of a single sequence with the consensus structure. For the structures compared in base-pairing distance, we adopted MFE structures. The SCI and base-pairing distance were implemented by using RNAfold with options "`-d2`" and RNAalifold with options "`-d2`". RNAalifold has recently been updated by replacing the simple covariance scores with a more sophisticated RIBOSUM[31]-like scoring matrices.[32] However, it has been reported that the new covariance scoring matrices failed to improve the accuracy of

Table 2. Averaged AUC of each measurement.

Method	Reference	CLUSTAL W
C-SCI (McCaskill model)	0.950	0.899
C-SCI (CONTRAfold model)	0.955	0.912
SCI	0.927	0.853
Base-pair distance (consensus)	0.905	0.849
Base-pair distance (pairwise)	0.900	0.854

the SCI although this update improved the accuracy of common secondary structure predictions. Therefore, we employed the previous covariance scores described in Ref. 12. For implementing the C-SCI, we used CentroidFold version 0.0.4 for predicting secondary structures, and RNAeval[11] from Vienna RNA Package version 1.7.2 for calculating the free energy of predicted structures. The C-SCI has two parameters γ_A and γ_S, and we performed 10-fold cross-validation for each bin of normalized Shannon entropy. We determined γ_A and γ_S which maximize AUC on the 90% of the dataset in each bin, and calculate AUC on the rest 10% of the dataset. As for evaluation we adopted the average of 10 AUCs.

3.2. *Discrimination capability*

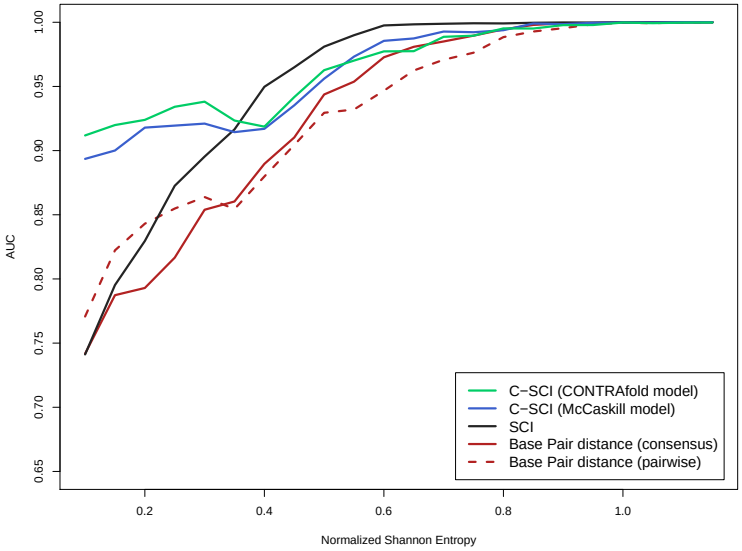
Figure 1 shows the results of AUC analysis of the C-SCI (McCaskill model, CONTRAfold model), the SCI and base-pair distance (consensus, pairwise) on reference alignments and CLUSTAL W alignments for each bin of normalized Shannon entropy, indicating that the C-SCI achieved the highest AUC, especially on low entropy region. Table 2 shows the summarized result by averaging AUC values in all bins. This indicates that the C-SCI achieves higher AUC on both alignments than the other measurements. Especially with CONTRAfold model, the C-SCI achieved the highest AUC in the C-SCI variants. The parameters γ_A and γ_S used in 10-fold cross-validation are written in Table S1 in the Supplemental material.

Furthermore, to investigate the reason why on the low entropy region the C-SCI could achieve extremely higher AUC than the other measurements on both alignments, we plotted the behavior of the median value of the score on reference alignments for each bin of normalized Shannon entropy. To clarify the difference between the SCI and the C-SCI, we show the result of two measurements: the SCI and the C-SCI with CONTRAfold model which has the highest averaged AUC as shown in Fig. 2. For the SCI, the score distribution of the positive data and that of the negative data is so close that even 25%-quantile of the positives and 75%-quantile of the negatives overlap on low entropy region. On the other hand, the C-SCI could clearly separate these score distributions well with γ_A and γ_S optimized by 10-fold cross-validation for each bin of normalized Shannon entropy. This suggests that the C-SCI has higher discriminant power than the SCI, especially, on low entropy region.

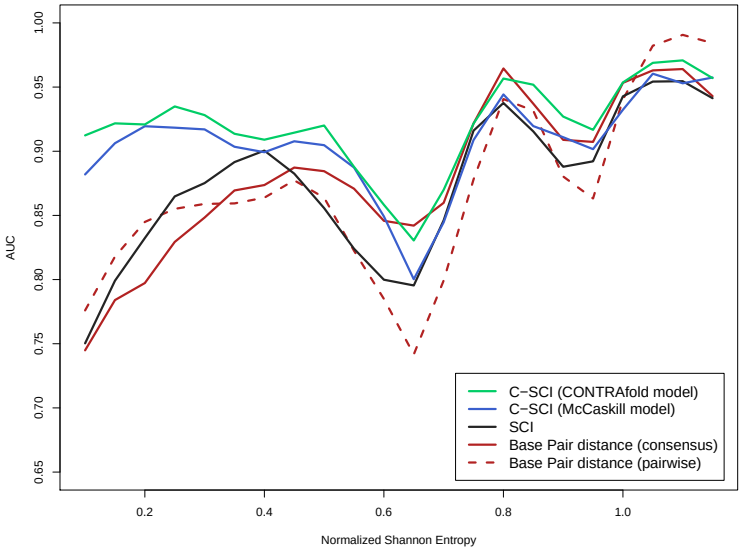
3.3. *Computational complexity*

To address the genome-wide search, the computational cost is a serious problem. It is obvious that the use of reference alignments which are structurally corrected by human curation is impractical although all the measurements achieve high accuracy on reference alignments. Two alternative approaches are to use structural aligners which can align RNA sequences with conserving their secondary structures, and to use the standard aligners like CLUSTAL W. We produced the structural alignments using RAF[18] version 1.00 with default settings. RAF is one of the most efficient structural aligners based on the Sankoff algorithm[33] which simultaneously aligns and folds given RNA sequences. All the experiments was executed on a Linux machine with AMD Opteron 2200SE (2.8GHz).

As shown in Tab. 3, all the measurements on RAF alignments achieve as high accuracy as those on reference alignments, and much higher than those on CLUSTAL W alignments. However, huge computational time is required for producing structural alignments even by RAF (the elapsed time: 1.86 seconds on average), which is known as the most efficient structural aligner, comparing with CLUSTAL W (0.0147 seconds on average). On the other hand, our approach the C-SCI, has an advanced property of robustness against low



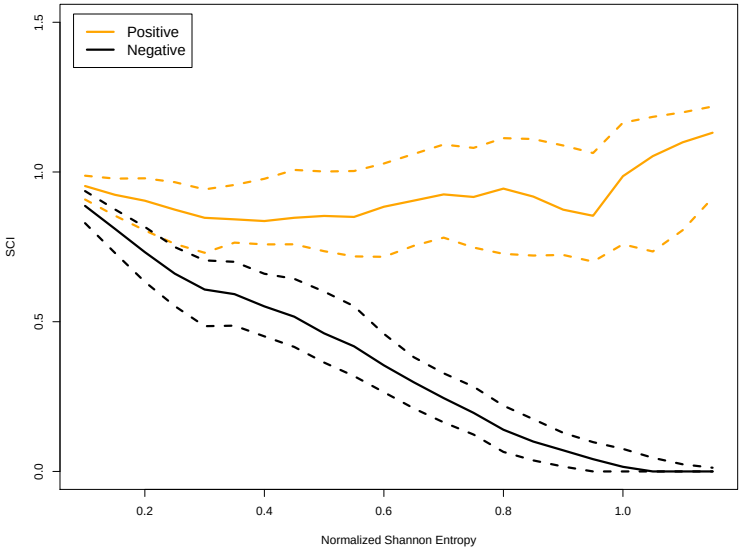
(a) reference alignments



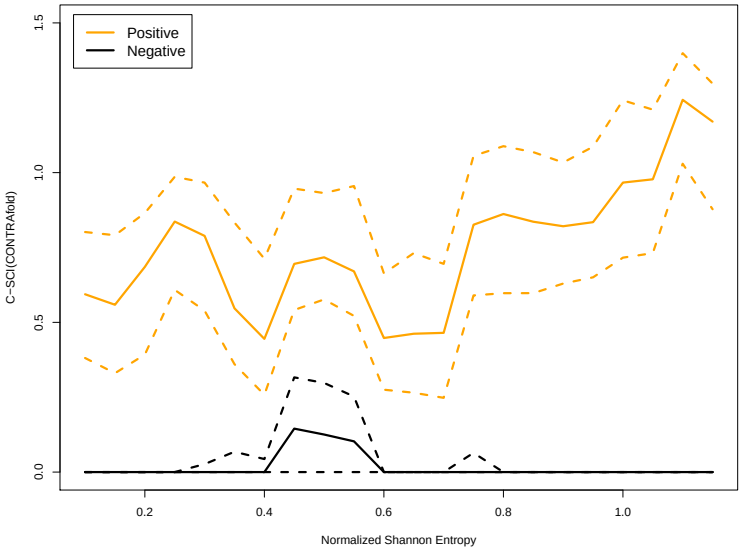
(b) CLUSTAL W alignments

Fig. 1. The discrimination capacity of the C-SCI, the SCI and base-pair distance in AUC on reference alignments and CLUSTAL W alignments for each bin of normalized Shannon entropy.

quality alignments. In fact, Tab. 3 indicates that averaged AUC of the C-SCI with CONTRAfold model on CLUSTAL W alignments is comparable with that of the SCI on RAF alignments. Furthermore, the elapsed time for calculating the SCI through RAF alignments was 1.99 seconds for each alignment on average, whereas that of the C-SCI with CONTRAfold model through CLUSTAL W alignments was only 0.426 seconds for each alignment on average. In case that structural alignments might be unavailable such as the genome-wide search, the C-SCI is practical to use and is expected to have as high discriminant power as the SCI on structural alignments.



(a) SCI



(b) C-SCI (CONTRAfold model)

Fig. 2. The behavior of median of the SCI and the C-SCI distribution along with the entropy on reference alignments. In each of positive and negative line sets, the solid line means the median of the distribution and the lower and upper dashed lines mean 25%-quantile and 75%-quantile of the distribution.

4. Discussion

We proposed the C-SCI, an improved measurement of secondary structure conservation, and examined its performance. The result was summarized in Tab. 2, which shows that the C-SCI is much more discriminative than the SCI and other measurements. This is because the C-SCI outperforms others in low entropy area as shown in Fig. 1. By the observation that important genes have high sequence identities between related species on the sequence alignment, the alignments with high sequence identity can be in the major part of data on which calculation of the measurement are performed. Therefore, the improvement of the accuracy on low entropy will be of great benefit. Table 2 also shows that the C-SCI with CONTRAfold model exceeds the C-SCI with McCaskill model. This is because CONTRAfold model has more appropriate parameters to

Table 3. Calculation time and AUC of each measurement. The result of calculation time is shown on the second scale.

	RAF			CLUSTAL W		
Method	AUC	Time[a]	Total time[b]	AUC	Time[a]	Total time[b]
C-SCI (McCaskill model)	0.953	0.241	2.10	0.899	0.222	0.237
C-SCI (CONTRAfold model)	0.957	0.445	2.30	0.912	0.411	0.426
SCI	0.923	0.130	1.99	0.853	0.135	0.150
Base-pair distance (consensus)	0.908	0.195	2.06	0.849	0.188	0.203
Base-pair distance (pairwise)	0.901	0.179	2.04	0.854	0.166	0.181

Time[a]: elapsed time for calculating the SCI or the C-SCI only. Total time[b]: total elapsed time for aligning sequences and calculating the SCI or the C-SCI.

estimate a secondary structure.

To examine the reason why the improvement on low entropy region occurs, we further calculated the score distribution of positive data and negative data of each measurement and showed that the C-SCI could separate these data more clearly in Fig. 2. This also shows that the median of the C-SCI on negative controls gets close to 0, whereas that of the SCI does not in low entropy region. We can discuss two things: why the C-SCI on negative controls tends to get close to 0 and why the C-SCI exhibits higher discrimination capability than the SCI. For the former question, we suppose that this is because the tendency that the consensus secondary structure is the open chain or an unstable structure is strong on the negative controls whereas not on the positive data with the proper γ_A and γ_S. For the latter one, we suppose that this is because MFE of consensus structure does not increase so much by shuffling columns, whereas the energy of the consensus structure calculated by a γ-centroid estimator increase significantly. Note that we cannot exclude the possibility that the shuffling algorithm used in our experiments does not work uniformly for all the bin of the entropy to preserve gap patterns and conservation patterns of columns. The number of possible pairs of columns to be shuffled depends on the gap patterns and conservation patterns which are reflected in the entropy. Further investigation should be done by using more sophisticated negative controls generating algorithms such as `SISSIz`[10] which can preserve dinucleotide composition in alignments in expectation.

Moreover, we investigated the computational time for calculating measurement and alignment shown in Tab. 3. This shows that the C-SCI on CLUSTAL W alignment is expected to be as discriminative as the SCI on structural alignment although the C-SCI through CLUSTAL W alignment is 4.7 times faster. Hence the C-SCI is practical, also considering that calculating structural alignments is not reasonable for genome-wide search. We can conclude that the C-SCI is computationally practical to use as well as much more discriminative than the SCI.

Supplemental materials

The parameters γ_A and γ_S optimized by 10-fold cross-validation for each bin of normalized Shannon entropy in CLUSTAL W alignments are written in the supplemental material. See the following file.
http://www.dna.bio.keio.ac.jp/~okada/psb2010/supplemental1.pdf

Acknowledgements

This work was supported in part by a grant from "Functional RNA Project" funded by the New Energy and Industrial Technology Development Organization (NEDO) of Japan, and was also supported in part by Grant-in-Aid for Scientific Research on Priority Area "Comparative Genomics" from the Ministry of Education, Culture, Sports, Science and Technology of Japan. We thank Michiaki Hamada and Yutaka Saito for fruitful discussions.

References

1. S. R. Eddy, *Nat Rev Genet* **2**, 919(Dec 2001).
2. J. S. Mattick and I. V. Makunin, *Hum Mol Genet* **15 Spec No 1**, R17(Apr 2006).

3. E. Rivas and S. R. Eddy, *Bioinformatics* **16**, 583(Jul 2000).
4. E. Rivas and S. R. Eddy, *BMC Bioinformatics* **2**, p. 8 (2001).
5. D. di Bernardo, T. Down and T. Hubbard, *Bioinformatics* **19**, 1606(Sep 2003).
6. A. Coventry, D. J. Kleitman and B. Berger, *Proc Natl Acad Sci U S A* **101**, 12102(Aug 2004).
7. S. Washietl, I. L. Hofacker and P. F. Stadler, *Proc Natl Acad Sci U S A* **102**, 2454(Feb 2005).
8. J. S. Pedersen, G. Bejerano, A. Siepel, K. Rosenbloom, K. Lindblad-Toh, E. S. Lander, J. Kent, W. Miller and D. Haussler, *PLoS Comput Biol* **2**, p. e33(Apr 2006).
9. M. Hamada, H. Kiryu, K. Sato, T. Mituyama and K. Asai, *Bioinformatics* **25**, 465(Feb 2009).
10. T. Gesell and S. Washietl, *BMC Bioinformatics* **9**, p. 248 (2008).
11. I. L. Hofacker, *Nucleic Acids Res* **31**, 3429(Jul 2003).
12. I. L. Hofacker, M. Fekete and P. F. Stadler, *J Mol Biol* **319**, 1059(Jun 2002).
13. S. Washietl, I. L. Hofacker, M. Lukasser, A. Hüttenhofer and P. F. Stadler, *Nat Biotechnol* **23**, 1383(Nov 2005).
14. S. Washietl, J. S. Pedersen, J. O. Korbel, C. Stocsits, A. R. Gruber, J. Hackermüller, J. Hertel, M. Lindemeyer, K. Reiche, A. Tanzer, C. Ucla, C. Wyss, S. E. Antonarakis, F. Denoeud, J. Lagarde, J. Drenkow, P. Kapranov, T. R. Gingeras, R. Guigò, M. Snyder, M. B. Gerstein, A. Reymond, I. L. Hofacker and P. F. Stadler, *Genome Res* **17**, 852(Jun 2007).
15. W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler and D. Haussler, *Genome Res* **12**, 996(Jun 2002).
16. M. Blanchette, W. J. Kent, C. Riemer, L. Elnitski, A. F. A. Smit, K. M. Roskin, R. Baertsch, K. Rosenbloom, H. Clawson, E. D. Green, D. Haussler and W. Miller, *Genome Res* **14**, 708(Apr 2004).
17. A. X. Wang, W. L. Ruzzo and M. Tompa, *BMC Bioinformatics* **8**, p. 417 (2007).
18. C. B. Do, C.-S. Foo and S. Batzoglou, *Bioinformatics* **24**, i68(Jul 2008).
19. J. D. Thompson, D. G. Higgins and T. J. Gibson, *Nucleic Acids Res* **22**, 4673(Nov 1994).
20. I. H. A.R. Gruber, Stephan H Bernhart and S. Washietl, *BMC Bioinformatics* **9**, p. 122 (2008).
21. J. S. McCaskill, *Biopolymers* **29**, 1105 (1990).
22. C. B. Do, D. A. Woods and S. Batzoglou, *Bioinformatics* **22**, e90(Jul 2006).
23. H. Kiryu, T. Kin and K. Asai, *Bioinformatics* **23**, 434(Feb 2007).
24. Y. Ding, C. Y. Chan and C. E. Lawrence, *RNA* **11**, 1157(Aug 2005).
25. L. E. Carvalho and C. E. Lawrence, *Proc Natl Acad Sci U S A* **105**, 3209(Mar 2008).
26. A. R. Gruber, S. H. Bernhart, I. L. Hofacker and S. Washietl, *BMC Bioinformatics* **9**, p. 122 (2008).
27. A. Wilm, I. Mainz and G. Steger, *Algorithms Mol Biol* **1**, p. 19 (2006).
28. S. Griffiths-Jones, S. Moxon, M. Marshall, A. Khanna, S. R. Eddy and A. Bateman, *Nucleic Acids Res* **33**, D121(Jan 2005).
29. S. Washietl and I. L. Hofacker, *J Mol Biol* **342**, 19(Sep 2004).
30. T. Sing, O. Sander, N. Beerenwinkel and T. Lengauer, *Bioinformatics* **21**, 3940(Oct 2005).
31. R. J. Klein and S. R. Eddy, *BMC Bioinformatics* **4**, p. 44(Sep 2003).
32. S. Bernhart, I. Hofacker, S. Will, A. Gruber and P. Stadler, *BMC Bioinformatics* **9**, p. 474(Nov 2008).
33. D. Sankoff, *SIAM Journal on Applied Mathematics* **45**, 810 (1985).

DYNAMIC PROGRAMMING ALGORITHMS FOR RNA STRUCTURE PREDICTION WITH BINDING SITES

UNYANEE POOLSAP*, YUKI KATO*†, TATSUYA AKUTSU

Bioinformatics Center, Institute for Chemical Research, Kyoto University, Gokasho, Uji, Kyoto 611-0011, Japan
E-mail: {unyanee,ykato,takutsu}@kuicr.kyoto-u.ac.jp

Noncoding antisense RNAs have recently occupied considerable attention and several computational studies have been made on RNA-RNA interaction prediction. In this paper, we present novel dynamic programming algorithms for predicting the minimum energy secondary structure when binding sites of one of the two interacting RNAs are known. Experimental results on several known RNA-RNA interaction data show that our proposed method achieves good performance in accuracy and time.

Keywords: RNA secondary structure; RNA-RNA interaction; dynamic programming

1. Introduction

In recent years, analysis of noncoding RNAs has attained great importance. They play a crucial role in some biological processes including post-transcriptional regulation of gene expression. Some noncoding RNAs, called *antisense RNAs*, aim at inhibiting their target RNA function through base complementary binding. Some antisense RNAs use full complementarity to their target for binding, whereas a number of antisense RNAs use partial complementarity,[1] and several *kissing hairpin* structures (Fig. 1) caused by loop-loop interaction have been reported.[2]

To predict joint secondary structures of interacting RNAs, several dynamic programming (DP) algorithms have been proposed so far. Andronescu *et al.*[3] developed the PairFold algorithm for secondary structure prediction of two interacting RNAs of minimum free energy. Since this algorithm is based on the Zuker's algorithm[4] for predicting pseudoknot-free structure of a single RNA, its time complexity is $O((n+m)^3)$ where n and m are respective lengths of two input sequences. The PairFold algorithm, however, cannot deal with any kissing hairpins, which are essentially equivalent to pseudoknotted structures when concatenating two interacting sequences. On the other hand, DP algorithms presented by Pervouchine,[5] Alkan *et al.*[6] and Kato *et al.*[7] can predict joint secondary structures including kissing hairpins in $O(n^3m^3)$ time. However, the time complexity of these algorithms is prohibitive in case $n \simeq m$ (i.e., $O(n^6)$ time), which is the same complexity of a prediction algorithm for pseudoknots.[8]

Viewing RNA-RNA interaction prediction from a different angle inspires us to consider the situation where we aim at predicting the secondary structure with binding sites of one of the two interacting RNAs (e.g., target RNA) on condition that interacting sites of the other RNA (e.g., antisense RNA) are known. In fact, we assume that a "profile" of intermolecular binding is given in advance, which can be obtained from the known secondary structure of the antisense RNA. This assumption could be reasonable since we can reduce computational complexity of a kind of interaction prediction and discover new target RNAs for antisense RNAs with known profiles. In this paper, we propose novel DP algorithms for predicting RNA secondary structures with binding site locations. Note that our formulation of the prediction problem requires that the order in which binding sites appear in an antisense RNA should be the same as the order in its target RNA (see Fig. 1). To deal with binding sites as well as base-paired structures, we design an extension of the classical Nussinov's algorithm,[9] which essentially minimizes the sum of base pair energies. In addition, we develop another DP algorithm that can incorporate stacking energy, which is based on the Zuker's algorithm.[4] Both of our proposed algorithms can run in $O(N^3n^3)$ time where N is the number of binding sites and n is a

*These authors contributed equally to this work.
†To whom correspondence should be addressed.

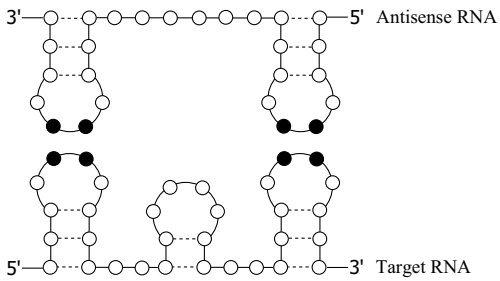


Fig. 1. An example of RNA-RNA interaction containing kissing hairpins. A black circle indicates one base of a binding site.

Table 1. An energy function e cited from Ref. 10.

Base pair	Energy value
{G, C}	-5
{A, U}	-4
{G, U}	-1

sequence length. Since N can be regarded as a constant in most cases, the time complexity of our algorithms can be evaluated as $O(n^3)$. We demonstrate the performance of our approach using the proposed algorithms on some data sets.

2. Methods

In this section, we will present dynamic programming (DP) algorithms for predicting RNA secondary structures with binding sites. Before going through the details of the algorithms, let us begin with definitions of RNA secondary structure and the prediction problem considering binding sites.

2.1. *Preliminaries*

Definition 2.1 (RNA secondary structure). *For an RNA sequence $s = s_1 s_2 \cdots s_n$ where $s_i \in \Sigma = \{A, C, G, U\}$ $(1 \leq i \leq n)$, a secondary structure of s is defined as a set R of position pairs (i, j) that satisfies the following conditions:*

- $1 \leq i < i + 1 < j \leq n$;
- $\forall (i, j), (i', j') \in R$; $i = i' \iff j = j'$.

Next, let us formally define the binding site profile.

Definition 2.2 (Binding site profile). *Let N be the number of binding sites and $\bar{b}_p = \bar{s}_{p,1}\bar{s}_{p,2}\cdots\bar{s}_{p,\ell_p} \in \Sigma^*$ $(1 \leq p \leq N)$ denote a binding site (subsequence) of an antisense RNA sequence. Let $s_i s_{i+1} \cdots s_j \in \Sigma^*$ be a subsequence of a target RNA sequence. Then, for each p $(1 \leq p \leq N)$, a binding site profile $I_p(i, j)$ of $s_i s_{i+1} \cdots s_j$ is defined as follows:*

$$I_p(i,j) = \begin{cases} \gamma \sum_{k=1}^{\ell_p} e(s_{i+k-1}, \bar{s}_{p,k}) & (j = i + \ell_p - 1, \text{ and } \forall k;\ s_{i+k-1} \text{ is complementary to } \bar{s}_{p,k}), \\ \infty & (\text{otherwise}), \end{cases} \tag{1}$$

where γ is a positive weight parameter, and e is an energy function that maps from a valid base pair to the corresponding energy value (see Table 1).

It should be noted that we do not know the actual binding sites of the target RNA in advance even though the actual binding sites of the antisense RNA are given. Instead of using the binding site profile, estimates

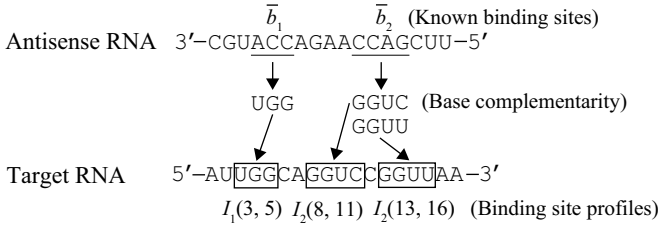


Fig. 2. An example of binding site profile.

of binding energies could be used in our algorithms if the binding sites of the antisense RNA are known and the binding energies depend on the stacking energies of the base pairs involved in the binding sites.

Example 2.1. Consider an antisense RNA sequence shown in Fig. 2 with known binding sites $\bar{b}_1 = \text{ACC}$ and $\bar{b}_2 = \text{CCAG}$. We first compute their complementary subsequences UGG for $\bar{b}_1$ and GGUC, GGUU for $\bar{b}_2$. Notice that we also take the wobble pair {G, U} into account. Then, we search for those complementary subsequences through the target sequence. Since UGG matches at the location from 3 to 5 in the target, we have $I_1(3,5) = -4-5-5 = -14$ where $\gamma = 1$ in Eq. (1) and the energy function shown in Table 1 are used. The rest of the elements of I_1 are equal to ∞ because no other sites of length three in the target match UGG. In a similar way, we obtain $I_2(8,11) = -19$ and $I_2(13,16) = -15$, and the rest of the elements of I_2 are set at ∞.

With these definitions, we define the prediction problem of RNA secondary structure with binding sites.

Definition 2.3 (RNA secondary structure prediction with binding sites).

Input: *A target RNA sequence $s = s_1 s_2 \cdots s_n \in \Sigma^*$ and N binding site profiles $I_1, I_2, \ldots, I_N$ of s.*

Output: *The optimum secondary structure of s whose subsequences match the binding sites in the order from I_1 to I_N.*

2.2. *DP Algorithms*

We develop two prediction models based on DP. The first DP-based model is an extension of the Nussinov's algorithm[9] using a simple base pair energy function. For the second model, we extend the first model to utilize the stacking energy and loop energy functions, which is based on the Zuker's algorithm.[4]

2.2.1. *Base pair energy model*

In the beginning, we define DP tables to design the algorithm. Let $s = s_1 s_2 \cdots s_n$ be an RNA sequence. As in the conventional case, we let $W(i,j)$ denote the minimum free energy of secondary structure formed from a subsequence $s_i s_{i+1} \cdots s_j$ of s. In addition, let $W_{pq}(i,j)$ be the minimum free energy of secondary structure for $s_i s_{i+1} \cdots s_j$ that contains binding sites corresponding to $I_p, I_{p+1}, \ldots, I_q$ $(1 \le p \le q \le N)$. Note that $W_{pq}(i,j)$ is a four dimensional table. We use this notation to facilitate comparison with the other DP table $W(i,j)$.

These DP tables are initialized as follows:

$$W(i,i) = 0,\ W_{pq}(i,i) = \infty \quad (1 \le \forall i \le n;\ 1 \le \forall p \le \forall q \le N).$$

The recursions are classified into three cases as shown below. In the first case, we use the simple Nussinov's algorithm for predicting secondary structure without binding sites. The second case is used for dealing with the structure with just one binding site. The third case is used for predicting the structure with two or more binding sites.

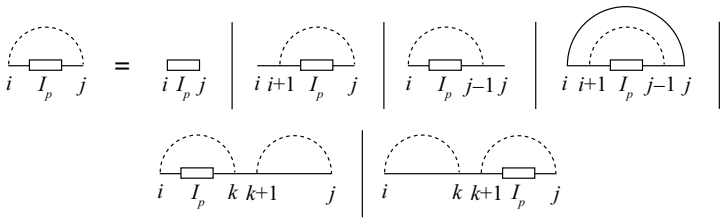


Fig. 3. Recursion for $W_{pp}(i,j)$. A dashed curve indicates that we do not know whether or not two bases connected by the curve form a base pair, and a solid curve shows that two bases connected by it definitely form a base pair.

Case 1 (the Nussinov's algorithm):

$$W(i,j) = \min \begin{cases} W(i+1,j), \\ W(i,j-1), \\ W(i+1,j-1) + e(i,j), \\ \min\limits_{i \le k < j} \{W(i,k) + W(k+1,j)\}, \end{cases} \tag{2}$$

where $e(i,j)$ is the simple energy function for a base pair (s_i, s_j). In the above DP recursion, the first and the second cases of minimization represent the cases where s_i and s_j do not form a base pair. The third case says that s_i and s_j form a base pair, and the resulting energy $e(i,j)$ is added to the present value of W. The forth formula represents the bifurcation structure. Note that k is the position at which the structure bifurcates in such a way that the sum of energies of two substructures is minimized.

Case 2 ($p = q$):

$$W_{pp}(i,j) = \min \begin{cases} I_p(i,j), \\ W_{pp}(i+1,j), \\ W_{pp}(i,j-1), \\ W_{pp}(i+1,j-1) + e(i,j), \\ \min\limits_{i \le k < j} \{W_{pp}(i,k) + W(k+1,j)\}, \\ \min\limits_{i \le k < j} \{W(i,k) + W_{pp}(k+1,j)\}. \end{cases} \tag{3}$$

The first case means that $s_i s_{i+1} \cdots s_j$ is a possible binding site and we adopt the corresponding score $I_p(i,j)$ computed in Eq. (1). The formulas from the second through the fourth are similar to the ones from the first through the third in Eq. (2). The fifth case represents the bifurcation structure where the binding site is contained in the former part of the bifurcation. Since the latter part of the bifurcation does not contain any binding sites, we use W computed in Eq. (2). The last case is a counterpart of the fifth case. Following a diagrammatic representation in Ref. 8, we provide a schematic representation of the recursion for $W_{pp}(i,j)$ in Fig. 3.

Case 3 ($q \ge p+1$):

$$W_{pq}(i,j) = \min \begin{cases} \min\limits_{i \le k < j} \min\limits_{p \le r < q} \{W_{pr}(i,k) + W_{r+1,q}(k+1,j)\}, \\ W_{pq}(i+1,j), \\ W_{pq}(i,j-1), \\ W_{pq}(i+1,j-1) + e(i,j), \\ \min\limits_{i \le k < j} \{W_{pq}(i,k) + W(k+1,j)\}, \\ \min\limits_{i \le k < j} \{W(i,k) + W_{pq}(k+1,j)\}. \end{cases} \tag{4}$$

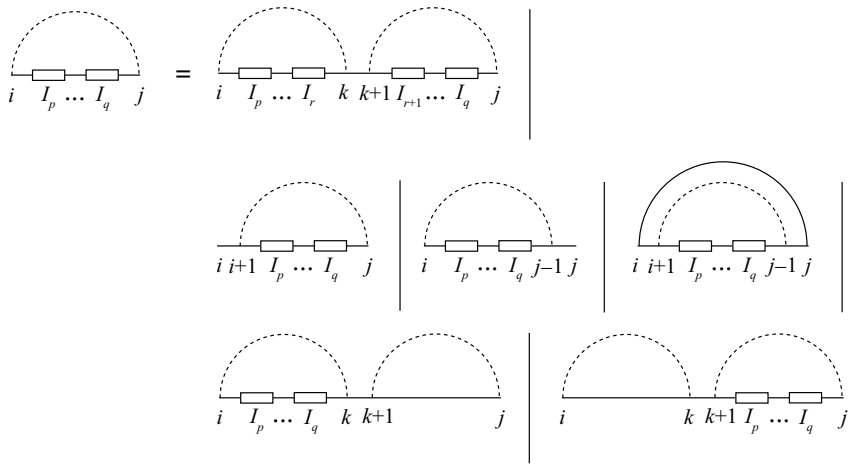


Fig. 4. Recursion for $W_{pq}(i, j)$.

The first case is designed for computing the bifurcation of secondary substructures, each of which contains the binding sites. It should be noted that the former part of the bifurcation contains the binding sites corresponding to $I_p, \ldots, I_r$, whereas the latter part corresponds to the substructure with binding sites for $I_{r+1}, \ldots, I_q$. The other cases can be interpreted as in Case 2. Figure 4 illustrates the above DP recursion.

We now evaluate the complexity of the above algorithm. Computing Eq. (2) takes $O(n^3)$ time. Equations (3) and (4) can be computed in $O(Nn^3)$ and $O(N^3n^3)$ time, respectively. Therefore, the overall time complexity is evaluated as $O(N^3n^3)$. By similar evaluation, we can see that the space complexity is $O(N^2n^2)$.

The minimum energy of the secondary structure of the input sequence is equivalent to $W_{1,N}(1, n)$, and the optimum secondary structure can be retrieved by tracing back the DP tables from $W_{1,N}(1, n)$.

2.2.2. *Stacking energy model*

Since the energy function used in the above DP algorithm is very simple, there is room for further improvement of our DP model. It is widely accepted that calculating contributions for stacking energy rather than individual contributions for each base pair yields better prediction. Hence, we extend the above DP algorithm based on this idea. In order to incorporate stacking energy into our previous DP model, we introduce additional DP tables. Let $V(i, j)$ be the minimum free energy of secondary structure formed from a subsequence $s_i s_{i+1} \cdots s_j$ such that s_i and s_j form a base pair. Let $V_{pq}(i, j)$ be the minimum free energy of secondary structure for $s_i s_{i+1} \cdots s_j$ that contains binding sites corresponding to $I_p, I_{p+1}, \ldots, I_q$ such that s_i and s_j form a base pair. Note that $W(i, j)$ and $W_{pq}(i, j)$ are defined in the same way as in the base pair energy model. Although energies of multi-branched and exterior loops could be incorporated into the recursions of W and W_{pq}, we exclude such energy rules for simplicity.

Initialization conditions for W and V are as follows:

$$W(i, i) = \infty,\ V(i, i) = \infty,\ W_{pq}(i, i) = \infty,\ V_{pq}(i, i) = \infty \quad (1 \le \forall i \le n;\ 1 \le \forall p \le \forall q \le N).$$

The revised version of the DP recursions is as follows:

Case 1 (the Zuker's algorithm):

$$W(i, j) = \min \begin{cases} W(i + 1, j), \\ W(i, j - 1), \\ V(i, j), \\ \min\limits_{i \le k < j} \{W(i, k) + W(k + 1, j)\}, \end{cases} \tag{5}$$

$$V(i,j) = \min \begin{cases} eh(i,j), \\ V(i+1,j-1) + es(i,i+1,j-1,j), \\ \min\limits_{i<i'<j'<j} \{V(i',j') + ebi(i,i',j',j)\}, \\ \min\limits_{i<k<j-1} \{W(i+1,k) + W(k+1,j-1)\} + b, \end{cases} \tag{6}$$

where $eh(i,j)$ is the destabilizing energy of a hairpin loop closed by a pair of (s_i, s_j), $es(i,i+1,j-1,j)$ is the stacking energy of two pairs (s_i, s_j) and (s_{i+1}, s_{j-1}), $ebi(i,i',j',j)$ is the destabilizing energy of a bulge or an interior loop closed by pairs (s_i, s_j) and $(s_{i'}, s_{j'})$, and b is a penalty for a bifurcation structure. Notice that in Eq. (5), the case where s_i and s_j form a base pair is represented by $V(i,j)$. As can be seen in Eq. (6), $V(i,j)$ is computed by minimizing among the four cases. The first case represents the energy of a hairpin loop closed by (s_i, s_j). The second formula adds the stacking energy of (s_i, s_j) and (s_{i+1}, s_{j-1}) to the present value of V. The third case represents a substructure where a bulge or an interior loop occurs in $s_i \cdots s_{i'}$ and $s_{j'} \cdots s_j$. The fourth formula is used for computing bifurcation.

Case 2 $(p = q)$:

$$W_{pp}(i,j) = \min \begin{cases} I_p(i,j), \\ W_{pp}(i+1,j), \\ W_{pp}(i,j-1), \\ V_{pp}(i,j), \\ \min\limits_{i \le k < j} \{W_{pp}(i,k) + W(k+1,j)\}, \\ \min\limits_{i \le k < j} \{W(i,k) + W_{pp}(k+1,j)\}, \end{cases} \tag{7}$$

$$V_{pp}(i,j) = \min \begin{cases} \min\limits_{i<i'<j'<j} \{W_{pp}(i',j') + er(i,i',j',j)\}, \\ V_{pp}(i+1,j-1) + es(i,i+1,j-1,j), \\ \min\limits_{i<i'<j'<j} \{V_{pp}(i',j') + ebi(i,i',j',j)\}, \\ \min\limits_{i<k<j-1} \{W_{pp}(i+1,k) + W(k+1,j-1)\} + b, \\ \min\limits_{i<k<j-1} \{W(i+1,k) + W_{pp}(k+1,j-1)\} + b, \end{cases} \tag{8}$$

where $er(i,i',j',j)$ is the approximate destabilizing energy of a pair of subsequences $(s_{i+1} \cdots s_{i'-1}, s_{j'+1} \cdots s_{j-1})$, which is obtained by removing $s_{i'} \cdots s_{j'}$ from $s_{i+1} \cdots s_{j-1}$. $V_{pp}(i,j)$ is computed by minimizing among the five choices. The first formula represents the case where the binding site corresponding to I_p is contained in the sequence closed by a base pair (s_i, s_j). The other cases are similar to those of the $V(i,j)$ recursion.

Case 3 $(q \ge p+1)$:

$$W_{pq}(i,j) = \min \begin{cases} \min\limits_{i \le k < j} \min\limits_{p \le r < q} \{W_{pr}(i,k) + W_{r+1,q}(k+1,j)\}, \\ W_{pq}(i+1,j), \\ W_{pq}(i,j-1), \\ V_{pq}(i,j), \\ \min\limits_{i \le k < j} \{W_{pq}(i,k) + W(k+1,j)\}, \\ \min\limits_{i \le k < j} \{W(i,k) + W_{pq}(k+1,j)\}, \end{cases} \tag{9}$$

Table 2. Results of the base pair energy model (BPEM), where n is the length of a target sequence and N is the number of binding sites. Note that for the ATP sensitive ribozyme-Substrate, n indicates the length of the antisense sequence. Since the substrate does not fold into secondary structure, only the binding sites can be detected by the algorithms, which is too simple. Therefore, we used the substrate to compute the binding site profile and predicted secondary structure and binding sites of the ATP sensitive ribozyme.

Antisense-Target	n	N	SEN (%)	PPV (%)	Time (s)
Tar-Tar*[11]	16	1	100.00	90.00	0.20
R1inv-R2inv[12]	19	1	100.00	100.00	0.23
DIS-DIS[13]	35	1	82.35	73.68	0.85
CopA-CopT[14]	57	3	100.00	93.94	17.76
ATP sensitive ribozyme-Substrate[15]	59	2	52.17	36.36	9.01
IncRNA$_{54}$-RepZ[16]	61	2	100.00	94.87	9.72
RyhB-SodB[17]	87	1	37.50	25.00	10.27
OxyS-fhlA[14]	100	2	59.09	52.00	43.25
Average			78.89	70.73	11.41

$$V_{pq}(i,j) = \min \begin{cases} \min\limits_{i<i'<j'<j} \{W_{pq}(i',j') + er(i,i',j',j)\}, \\ V_{pq}(i+1,j-1) + es(i,i+1,j-1,j), \\ \min\limits_{i<i'<j'<j} \{V_{pq}(i',j') + ebi(i,i',j',j)\}, \\ \min\limits_{i<k<j-1} \{W_{pq}(i+1,k) + W(k+1,j-1)\} + b, \\ \min\limits_{i<k<j-1} \{W(i+1,k) + W_{pq}(k+1,j-1)\} + b. \end{cases} \tag{10}$$

$V_{pq}(i,j)$ in Case 3 differs from $V_{pp}(i,j)$ in Case 2 in that the present subsequence $s_i s_{i+1} \cdots s_j$ contains at least two binding sites.

Finally, we evaluate the complexity of this algorithm. Obviously, complexity for computing Eqs. (9) and (10) dominates the overall complexity of the algorithm. Computing the first formula of Eq. (9) takes $O(N^3 n^3)$ time. Exact analysis of the first and third formulas of Eq. (10) reveals time complexity of $O(N^2 n^4)$. In actual case, however, the loop size is bounded by a constant, and thus the complexity can be reduced to $O(N^2 n^2)$. Therefore, the overall time complexity is evaluated as $O(N^3 n^3)$. The space complexity is $O(N^2 n^2)$.

3. Results

Our two DP models were tested on the data set comprising eight antisense-target RNA complexes with known structures, taken from several literatures (see Tables 2–4). In fact, an antisense sequence was used for constructing a binding profile, whereas the corresponding target sequence was used for predicting its structure with binding sites. For the binding site profile computation, we used $\gamma = 2$ in Eq. (1). We employed Table 1 for the simple energy parameter e, and adopted sophisticated energy parameters for folding at 37°C provided by the Turner Group[18] (the recent version is available online at http://www.bioinfo.rpi.edu/zukerm/rna/energy/) for other parameters including eh, es, etc. We limited the size of interior and bulge loops to at most four nucleotides. The penalty for a bifurcation structure b was set at 1. We implemented the algorithms in Java on a machine with Intel Core 2 Duo CPU 1.20GHz and 2.00GB RAM. Prediction accuracy was measured using sensitivity (SEN) and positive predictive value (PPV) defined below:

$$\text{SEN} = \frac{\sharp \text{ of correctly predicted base pairs} + \sharp \text{ of correctly predicted bases of binding sites}}{\sharp \text{ of observed base pairs} + \sharp \text{ of observed bases of binding sites}},$$

$$\text{PPV} = \frac{\sharp \text{ of correctly predicted base pairs} + \sharp \text{ of correctly predicted bases of binding sites}}{\sharp \text{ of predicted base pairs} + \sharp \text{ of predicted bases of binding sites}}.$$

Note that $\sharp$ represents the number.

Tables 2 and 3 show the prediction accuracy of the base pair energy model (BPEM) and that of the stacking energy model (SEM), respectively. Figure 5 depicts predicted structures of the fhlA RNA of the longest sequence in the data set. We can see that SEM outperforms BPEM in terms of accuracy.

Table 3. Results of the stacking energy model (SEM).

Antisense-Target	SEN (%)	PPV (%)	Time (s)
Tar-Tar*	100.00	100.00	0.39
R1inv-R2inv	92.31	100.00	0.53
DIS-DIS	100.00	100.00	2.92
CopA-CopT	96.77	100.00	50.09
ATP sensitive ribozyme-Substrate	100.00	92.00	29.11
IncRNA$_{54}$-RepZ	100.00	97.37	30.23
RyhB-SodB	83.33	64.52	31.77
OxyS-fhlA	90.91	90.91	115.40
Average	95.42	93.10	32.56

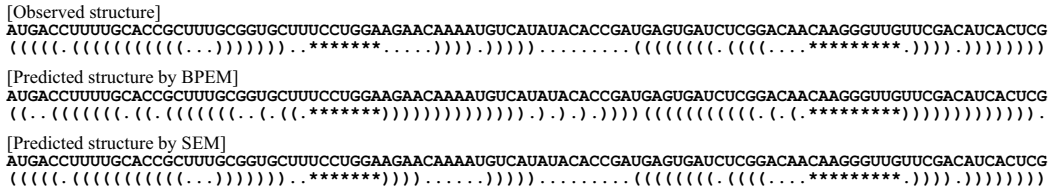


Fig. 5. Prediction results for the fhlA RNA. A pair of parentheses denotes a base pair and a series of asterisks represents a binding site.

Table 4. Comparison of F-measure (%) between our models, the stacked pair model (SPM) and the loop model (LM) presented in Ref. 6. Note that LM returned no base pair when ATP sensitive ribozyme-Substrate was used as an input.

Antisense-Target	**BPEM**	**SEM**	SPM	LM
Tar-Tar*	94.74	**100.00**	90.00	90.00
R1inv-R2inv	**100.00**	96.00	**100.00**	**100.00**
DIS-DIS	77.78	**100.00**	82.35	82.35
CopA-CopT	96.88	**98.36**	83.33	78.79
ATP sensitive ribozyme-Substrate	42.86	**95.83**	55.32	0.00
IncRNA$_{54}$-RepZ	97.37	**98.67**	81.58	81.58
RyhB-SodB	30.00	**72.73**	53.97	52.78
OxyS-fhlA	55.32	**90.91**	80.00	78.72
Average	74.37	**94.06**	78.17	70.68

We then compared the performance of our proposed methods with that of existing DP-based models, called the stacked pair model (SPM) and the loop model (LM) presented in Ref. 6, using the inteRNA web server[19] (see Table 4). In the web interface, we set gap penalty and maximum substructure length at 0 and 61, respectively, for all eight RNA pairs. SPM and LM took much time for prediction due to their high time complexity as stated in Sec. 1. We calculated F-measure F, which is the harmonic mean of SEN and PPV defined by $F = 2 \cdot \mathrm{SEN} \cdot \mathrm{PPV}/(\mathrm{SEN} + \mathrm{PPV})$. As Table 4 shows, the prediction performance of SEM is the best of all four models on average.

To demonstrate the applicability of our profile-based approach to target discovery, we further tried analyzing some RNAs with unknown structures and binding sites, given binding profiles of specific antisense RNAs. Test sequences were selected to be homologous to the known target sequence (query) for the antisense RNA using BLASTN. Figures 6 and 7 illustrate two candidate prediction results using SEM. We must note that in the figures it is not clear whether or not the predicted binding sites actually interact with the antisense RNAs since the number of detected binding sites was fewer than that of the known homologous target RNAs. The prediction result of CP001122 (see Fig. 6), which has 90% sequence identity with the known target CopT, shows that SEM detected two binding sites on CP001122. Note that the second binding site inside the hairpin loop (CUGC in Fig. 6) was ascribed to the profile computed from the third binding site located in the exterior loop of CopT, which leads to more uncertainty of predicted target sites. As for another example, SEM recognized only one binding sites on CU928158 (see Fig. 7), which has 89% sequence

```
AAACCCCGAUAAUCUUCGUCAAGUUUGGCGACUGCGAAGAAGAUUACCGGGGCCC
...(((((.(((((((((......******..****...)))))))))))))...
```

Fig. 6. A prediction result of a sequence CP001122 Salmonella enterica subsp. enterica serovar Kentucky str. CVM29188 plasmid pCVM29188_146, which has 90% sequence identity with the CopT RNA to which the antisense CopA binds. Notice that only two binding sites were predicted as compared with the known target CopT with three binding sites.

```
UUGCGGUGCAUUCCUGGAUAGCAAAAUGUCGUAUACACCAAUGAGCGAUCUCGGACAACAAGGGCUGUUCGAUAUCACUCG
.((((.(((..*******..)))......))))........((((.(((.(((((((........)))))))).))).))))
```

Fig. 7. A prediction result of a sequence CU928158 of the Escherichia fergusonii ATCC 35469 chromosome, which has 89% sequence identity with the fhlA RNA to which the antisense OxyS binds. Notice that only one binding site was predicted as compared with the known target fhlA with two binding sites.

identity with the known target fhlA. On the other hand, exactly two binding sites were found for another test sequence of 99% sequence identity with fhlA (the result is not shown here).

4. Conclusion

We proposed new dynamic programming algorithms for predicting RNA secondary structures with binding sites. The performance of the algorithms for the base pair energy model (BPEM) and the stacking energy model (SEM) was demonstrated for several known RNA-RNA interaction data. Judging from the results, it can be said that the advantages of BPEM are faster running time than that of SEM and the simplicity of DP, which is a natural extension of the classical prediction algorithm for single RNA folding. However, prediction accuracy of BPEM plummets for some inputs. On the other hand, SEM is a more complex model than BPEM but a robust one in a sense that prediction accuracy hardly fluctuates according to the length of an input sequence. We further compared the prediction results with those of other prediction methods, which shows that SEM outperforms those earlier models.

Our approach is a novel method of RNA-RNA interaction prediction from a different point of view (i.e., use of profile of intermolecular interaction), and achieves lower time complexity compared with earlier methods. Moreover, use of our profile-based method can improve prediction performance. Our method will also be useful in discovering new target sites for an antisense RNA with a known binding profile. To ensure this advantage, exhaustive search for candidate targets has to be performed as our future work. If the profile of an interacting protein is available, our method could be applied to RNA-protein interaction prediction, which is also left as a challenging task.

Acknowledgments

The first author thanks the Japanese Government (Monbukagakusho:MEXT) Scholarship from the Ministry of Education, Culture, Sports, Science and Technology. This work was supported in part by Grant-in-Aid for Young Scientists (Start-up) (KAKENHI) #20800023 from Japan Society for the Promotion of Science (JSPS), and by Grant-in-Aid for Scientific Research (KAKENHI) #19200022 from JSPS.

References

1. S. Brantl, *Biochim. Biophys. Acta* **1575**, 1–3 (2002).
2. C. Brunel, R. Marquet, P. Romby and C. Ehresmann, *Biochimie* **84**, 9 (2002).
3. M. Andronescu, Z.C. Zhang and A. Condon, *J. Mol. Biol.* **345**, 5 (2005).
4. M. Zuker and P. Stiegler, *Nucl. Acids Res.* **9**, 1 (1981).
5. D.D. Pervouchine, *Genome Inform.* **15** (2004).
6. C. Alkan, E. Karakoç, J.H. Nadeau, S.C. Şahinalp and K. Zhang, *J. Comput. Biol.* **13**, 2 (2006).
7. Y. Kato, T. Akutsu and H. Seki, *Pattern Recogn.* **42**, 4 (2009).
8. E. Rivas and S.R. Eddy, *J. Mol. Biol.* **285**, 5 (1999).
9. R. Nussinov, G. Pieczenik, J.R. Griggs and D.J. Kleitman, *SIAM J. Appl. Math.* **35**, 1 (1978).

10. P. Clote and R. Backofen, *Computational Molecular Biology*, John Wiley & Sons, Ltd (2000).
11. K.-Y. Chang and I. Tinoco Jr, *J. Mol. Biol.* **269**, 1 (1997).
12. M.J. Rist and J.P. Marino, *Nucl. Acids Res.* **29**, 11 (2001).
13. J.-C. Paillart, E. Skripkin, B. Ehresmann, C. Ehresmann and R. Marquet, *Proc. Natl. Acad. Sci. USA* **93**, 11 (1996).
14. E.G.H. Wagner and K. Flardh, *Trends Genet.* **18**, 5 (2002).
15. J. Tang and R.R. Breaker, *Nucl. Acids Res.* **26**, 18 (1998).
16. K. Asano and K. Mizobuchi, *J. Biol. Chem.* **275**, 2 (2000).
17. T.A. Geissmann and D. Touati, *The EMBO J.* **23**, 2 (2004).
18. D.H. Turner, N. Sugimoto and S.M. Freier, *Ann. Rev. Biophys. Biophys. Chem.* **17** (1988).
19. C. Aksay, R. Salari, E. Karakoç, C. Alkan and S.C. Şahinalp, *Nucl. Acids Res.* **35** (2007).

AN ALGORITHM FOR THE ENERGY BARRIER PROBLEM WITHOUT PSEUDOKNOTS AND TEMPORARY ARCS

CHRIS THACHUK[1], JÁN MAŇUCH[2], ARASH RAFIEY[2],
LEIGH-ANNE MATHIESON[1], LADISLAV STACHO[3] and ANNE CONDON[1]

(1) Department of Computer Science, University of British Columbia, Canada
(2) School of Computing Science, Simon Fraser University, Canada
(3) Department of Mathematics, Simon Fraser University, Canada

We make two new contributions to the problem of calculating pseudoknot-free folding pathways with minimum energy barrier between pairs $(\mathcal{A}, \mathcal{B})$ of RNA secondary structures. Our first contribution pertains to a problem posed by Morgan and Higgs: find a min-barrier *direct* folding pathway for a simple energy model in which each base pair contributes -1. In a direct folding pathway, intermediate structures contain only base pairs in $\mathcal{A}$ and $\mathcal{B}$ and are of length $|\mathcal{A} \triangle \mathcal{B}|$ (the size of the symmetric difference of the two structures). We show how to solve this problem exactly, using techniques for deconstructing bipartite graphs. The problem is NP-hard and so our algorithm requires exponential time in the worst case but performs quite well empirically on pairs of structures that are hundreds of nucleotides long. Our second contribution shows that for the simple energy model, repeatedly adding or removing a base pair from $\mathcal{A} \cup \mathcal{B}$ along a pathway is not useful in minimizing the energy barrier. Two consequences of this result are that (i) the problem of determining the min-barrier pseudoknot-free folding pathway from the space of direct pathways with repeats is NP-hard and (ii) our new algorithm finds the min-barrier pathway not only from the space of direct folding pathways but in fact from the space of direct pathways with repeats.

Keywords: RNA secondary structure; RNA folding pathways; energy barrier problem

1. Introduction

We present new algorithms for *exactly* computing direct folding pathways between two RNA structures that have minimum energy barrier, for a simple energy model. We first briefly motivate the energy barrier computation problem, describe previous work on energy barrier calculation and summarize our results.

Motivation. RNA molecules play vital roles in the cell, not only because of the diverse structures they can form but also because of their ability to fold into alternative structures under changing environmental conditions.[1–6] Thus knowledge of folding pathways between pairs of alternative RNA structures is very valuable for inferring RNA function in such environments, and is valuable also for predicting RNA structure, e.g., in light of co-transcriptional folding.[2,7–10]

Computational approaches for predicting folding pathways focus on secondary structure— the set of base pairs that form when an RNA molecule folds. As illustrated in Fig. 1, the folding pathway from an initial to a final structure is a sequence of intermediate structures, each differing from the previous one by a single base pair (or equivalently by a single arc in the arc diagram representation). Much focus to date has been on pathways of pseudoknot-free secondary structures— structures in which no base pairs cross. Since folding is a thermodynamically-driven probabilistic process, folding pathways tend to avoid high-energy structures. As a result, many methods for predicting folding pathways or energy landscapes— particularly course-grained methods designed to work for long structures which do not attempt to model the complete energy landscape— are guided by calculations of the *energy barrier*.[2,11] Energy barrier calculations have also been useful in constructing barrier trees and to study properties of disordered systems in statistical physics.[12–14]

If $\pi = \mathcal{P}_0, \mathcal{P}_1, \ldots, \mathcal{P}_m$ is a folding pathway from structure $\mathcal{A}$ to structure $\mathcal{B}$ (i.e., $\mathcal{A} = \mathcal{P}_0$ and $\mathcal{B} = \mathcal{P}_m$), the *energy barrier* of π is $\max_{1 \leq i \leq m} E(\mathcal{P}_i) - E(\mathcal{P}_0)$, where $E(\mathcal{S})$ denotes the free energy change of secondary structure S. In this paper we focus on a simple energy model in which each base pair contributes -1 to the total energy. The energy barrier of pair $(\mathcal{A}, \mathcal{B})$ is the lowest energy barrier, taken over all pathways from $\mathcal{A}$ to $\mathcal{B}$. Note that there is always a folding pathway from $\mathcal{A}$ to $\mathcal{B}$ in which first all arcs (base pairs) of $\mathcal{A}$ are removed and then all arcs of $\mathcal{B}$ are added. The question is whether there is another pathway that avoids

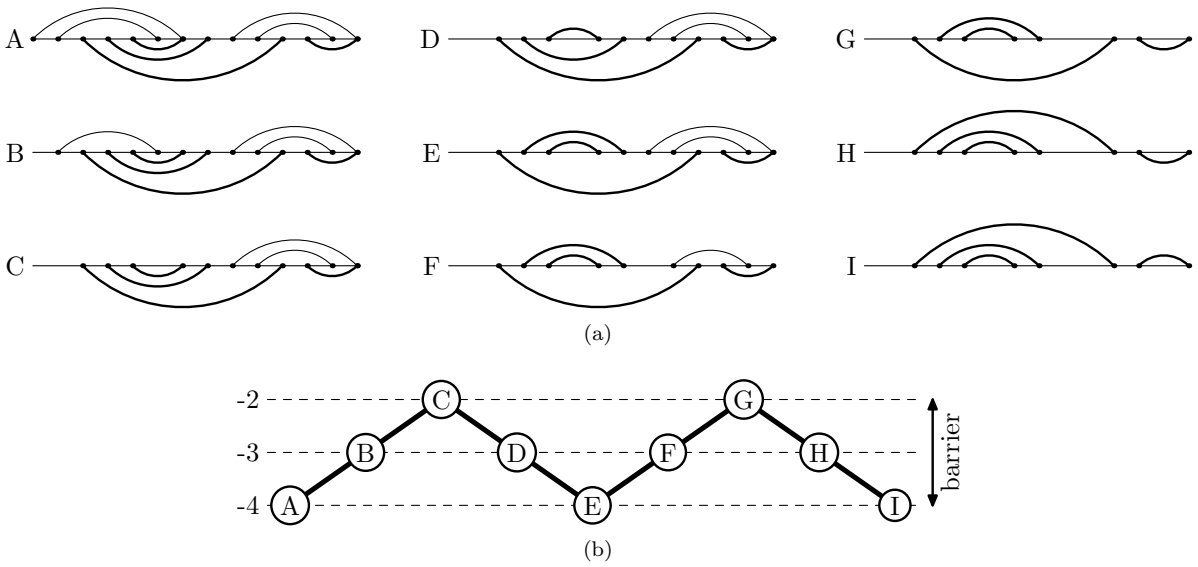


Fig. 1: (a) A possible pseudoknot-free folding pathway is shown for an initial structure A transitioning through intermediate structures $(B, C, \ldots)$ until the final structure I is reached. Structures are represented as arc diagrams. For a particular position in the pathway, the top of the arc diagram denotes the base pairs of the structure and the bottom of the diagram denoes the base pairs that are currently not in the structure but need to be in the final structure. Each structure along the pathway differs from its neighbours by at most one arc. Furthermore, each arc is either removed (if in A) or added (if in I) at most once along the pathway and thus the pathway is *direct*. (b) The corresponding energy plot. The barrier in this example is two.

such a high energy barrier, by adding arcs of $\mathcal{B}$ before all arcs of $\mathcal{A}$ are removed. Throughout, we consider only pseudoknot-free pathways, in which all intermediate structures are pseudoknot-free.

Background and previous work. There is a rich literature on the problem of predicting folding pathways and energy landscapes; see the recent work of Geis at al.,[2] Tang et al.[11] and the references therein. We focus here on algorithms for energy barrier calculation which are an important component of many approaches to estimation of folding pathways and energy landscapes. Such methods have been proposed, for example, by Morgan and Higgs,[13] Wolfinger et al.,[12,14] Flamm et al.[15] and Geis et al.[2]

Several versions of the energy barrier problem have been studied, which are distinguished by properties of the intermediate structures. Morgan and Higgs focus on *direct* folding pathways from structure $\mathcal{A}$ to structure $\mathcal{B}$ in which intermediate structures contain only arcs in $\mathcal{A} \cup \mathcal{B}$ and such that the total pathway length is $|\mathcal{A} \triangle \mathcal{B}|$. In such pathways, each arc from the initial structure not also in the final structure is removed exactly once and each arc from the final structure not also in the initial structure is added exactly once along the pathway. A larger class of pathways is obtained by allowing the the length of the pathway to exceed $|\mathcal{A} \triangle \mathcal{B}|$. We call such pathways *direct-with-repeats* pathways since an arc from $\mathcal{A}$ or $\mathcal{B}$ may be added or removed multiple times along the pathway. An even more general class of pathways allow intermediate structures to contain "temporary" base pairs which are neither in $\mathcal{A}$ nor in $\mathcal{B}$. Morgan and Higgs call such pathways *indirect*. Thus, direct pathways are a subclass of direct-with-repeats pathways, which in turn are a subclass of indirect pathways.

Morgan and Higgs assume the simple energy model in which each base pair contributes -1 to the total free energy. Using a randomized greedy approach, they construct several low-barrier direct pathways and take the minimum energy barrier of these as their estimate. (They also construct indirect pathways using a "single link clustering" method.) Wolfinger et al. use a barrier tree to represent the folding landscape; identifying nodes in the tree (which are called saddle points) is analogous to calculating energy barriers. Flamm et al.'s

method[15] for approximating energy barriers explores direct pathways by performing a breadth-first search, maintaining the best m candidate solutions at each step. As m becomes large, the search does become exhaustive, yielding an exact solution, however exponential runtime and memory are required. The program `barriers`[12] is capable of computing exact direct and indirect pathways, provided a complete sample of low energy states separating the two structures is provided. However, this approach is also exponential in runtime and space and thus unsuitable for medium or large problem instances.

For their Kinwalker folding pathway predictor, Geis et al.[2] describe a heuristic which explores the space of possible direct pathways in a more sophisticated manner than does the Morgan-Higgs heuristic, incorporating a parameter lookahead technique to avoid excessive runtimes. While their method uses the Turner energy model to evaluate energy barriers, it relies on simple addition and removal of base pairs (and thus the simple energy model) while generating putative low-barrier pathways.

In summary, all current methods are either heuristic in nature and thus are not guaranteed to find the exact energy barrier between two structures, or are exponential in both runtime and space, precluding their use on even medium sized problem instances. Thus there is strong motivation for finding a *fast* method which can *exactly* compute the energy barrier between two structures. Indeed, the Geis et al. method can estimate energy barriers for structures of long sequences (1,500nt or more) but the authors note that "as the performance of Kinwalker crucially depends on approximating saddle heights, further improvements to the Morgan-Higgs heuristic as well as alternative approaches will be investigated".

Finally, in earlier work, we showed that the problem of finding the energy barrier of direct pathways between two structures is NP-hard.[16] Thus we do not expect to find an algorithm for energy barrier calculation whose running time is bounded by polynomial in its input, in the worst case.

Our results. Our main contributions are new approaches for exactly calculating the energy barrier between two RNA secondary structures along direct, pseudoknot-free pathways. To develop a sound theoretical basis for energy barrier calculation we follow the approach of Morgan and Higgs: we assume a simple energy model in which each base pair contributes -1 to the total energy and we focus on the problem of finding min-barrier pathways between structures with minimum free energy. Structure $\mathcal{S}$ is a *minimum free energy* (MFE) structure for a given sequence if no other structure for the sequence has lower energy than $\mathcal{S}$. A sequence may have more than one MFE structure.

Our methods exploit elegant algorithms for bipartite graphs to split a problem into independent subproblems where possible. The theoretical run-time complexity of our algorithms is exponential in the worst case; this is not surprising since the problem is NP-hard.[16] However, our empirical analysis shows that implementations of our algorithms can often find the energy barrier on problem instances with hundreds of nucleotides in seconds. Our algorithms are highly amenable to parallelization and have potential to work with more sophisticated energy models that include Turner parameters for base stacking, for example. Furthermore, our methods could be integrated with current heuristics in order to improve energy barrier estimation on structures with thousands of nucleotides.

Our second contribution addresses the following question: for the simple Morgan-Higgs energy model could direct-with-repeats pathways have lower energy barriers than direct pathways? We show that the answer to this question is no: repeated arcs cannot help to lower the energy barrier of a pathway. Two consequences of this contribution are that (i) the problem of determining the min-barrier pseudoknot-free folding pathway from the space of direct-with-repeats pathways is NP-hard and (ii) our algorithm finds the min-barrier pathway not only from the space of direct folding pathways but in fact from the space of direct-with-repeats folding pathways.

In Section 2 we present our result that repeated arcs do not help to decrease the energy barrier in direct pathways. In Section 3 we present our new algorithmic techniques for exactly computing min-barrier pseudoknot-free folding pathways. We present our empirical analysis in Section 4. We provide a brief discussion and suggest directions for future work in Section 5.

2. Repeated base pairs do not lower the energy barrier

In this section we show that on a folding pathway from structure $\mathcal{A}$ to structure $\mathcal{B}$, repeatedly adding or removing arcs from $\mathcal{A}$ or $\mathcal{B}$ along the pathway cannot help to reduce the energy barrier. This enables us to consider only direct (i.e., repeat-free) pathways in our later algorithms without loss of generality.

It is convenient in this section to slightly generalize the definition of a folding pathway, so that a structure on the pathway differs from its predecessor by at most one (rather than exactly one) arc. A sequence $T = T[1], T[2], \ldots, T[m]$ is a *transformation sequence* for a direct-with-repeats pathway $\pi = \mathcal{P}_0, \ldots, \mathcal{P}_m$ if for all $i \in [1, m]$, one of the following holds:

- $T[i] = +a$, $\mathcal{P}_i = \mathcal{P}_{i-1} \cup \{a\}$, and $a \notin \mathcal{P}_{i-1}$;
- $T[i] = -a$, $\mathcal{P}_i = \mathcal{P}_{i-1} \setminus \{a\}$, and $a \in \mathcal{P}_{i-1}$;
- $T[i] =$ no-op and $\mathcal{P}_i = \mathcal{P}_{i-1}$;

where $a \in \mathcal{P}_0 \cup \mathcal{P}_m$. We call each $T[i]$ an *operation*. Let $\hat{T}[i] = a$ if $T[i] \in \{+a, -a\}$.

Theorem 2.1. *For any direct-with-repeats pseudoknot-free pathway π from structure $\mathcal{A}$ to structure $\mathcal{B}$ there is a direct pseudoknot-free pathway from $\mathcal{A}$ to $\mathcal{B}$ with energy barrier at most that of π.*

Proof. Consider a direct-with-repeats pathway $\pi = \mathcal{P}_0, \ldots, \mathcal{P}_m$ from $\mathcal{A} = \mathcal{P}_0$ to $\mathcal{B} = \mathcal{P}_m$ and let $T[1], \ldots, T[m]$ be a transformation sequence for π. For each $a \in \mathcal{A}$, we call each occurrence of $+a$ or $-a$ in T, except for the *last* $-a$, a *repeat operation*. Similarly, for each $b \in \mathcal{B}$, each occurrence of $+b$ or $-b$ in T except for the last $+b$ is a repeat operation.

If π is direct we are done, so suppose that π is not direct. We will show how to construct a pathway π' and corresponding transformation sequence which has fewer repeats than does T, with the barrier of π' being at most that of π. The proof follows since we can iterate this construction until we obtain a direct transformation sequence from $\mathcal{A}$ to $\mathcal{B}$.

Let $I^+_{\mathcal{A}}$ and $I^-_{\mathcal{A}}$ be the subset of indices $i \in [1, m]$ for which $T[i]$ is a repeat operation of the form $+a$ and $-a$, respectively, where $a \in \mathcal{A}$. Similarly, let $I^+_{\mathcal{B}}$ and $I^-_{\mathcal{B}}$ be the subset of indices $i \in [1, m]$ for which $T[i]$ is a repeat operation of the form $+b$ and $-b$, respectively, where $b \in \mathcal{B}$. To construct a new pathway π' with a smaller number of repeats we consider two cases:
Case 1: $|I^-_{\mathcal{A}}| \geq |I^+_{\mathcal{B}}|$.
Case 2: $|I^-_{\mathcal{A}}| < |I^+_{\mathcal{B}}|$.

We will construct π' only in Case 1. The construction in Case 2 follows by symmetry. Indeed, by reversing the pathway π and its transformation sequence T and replacing each $+$ by a $-$ and vice versa in T, we obtain a pathway $\bar{\pi}$ from $\bar{\mathcal{A}} = \mathcal{B}$ to $\bar{\mathcal{B}} = \mathcal{A}$ and a corresponding transformation sequence $\bar{T}$. Now we have that $|I^-_{\bar{\mathcal{A}}}| = |I^+_{\mathcal{B}}| > |I^-_{\mathcal{A}}| = |I^+_{\bar{\mathcal{B}}}|$. Therefore, the construction in Case 1 produces a transformation sequence from $\mathcal{B}$ to $\mathcal{A}$ with a smaller number of repeat operations. Its reversal, with $+$'s changed to $-$'s and vice versa, is a transformation sequence from $\mathcal{A}$ to $\mathcal{B}$ which has a smaller number of repeat operations than does T.

We use the following notation. For each $i \in I^-_{\mathcal{A}}$ (i.e., $T[i] = -a$ for some $a \in \mathcal{A}$ and is not the last such $-a$), find the smallest index $i' > i$ for which $T[i'] = +a$. We say that i is a partner of i' and vice versa, and write $\text{partner}(i) = i'$ and $\text{partner}(i') = i$. Similarly, every $j \in I^+_{\mathcal{B}}$ is partnered with the smallest $j' > j$ for which $T[j'] = -b$. For any $J \subseteq [1, m]$, let $\text{partner}(J) = \{\text{partner}(j); j \in J\}$.

Consider Case 1. We will identify subsets $I \subseteq I^-_{\mathcal{A}}$ and $J \subseteq I^+_{\mathcal{B}}$ such that the transformation sequence $T(I, J)$ obtained by replacing all operations in T at positions $I \cup \text{partner}(I) \cup J \cup \text{partner}(J)$ with no-op's has a smaller number of repeat operations and corresponds to a valid pathway π' from $\mathcal{A}$ to $\mathcal{B}$. By valid, we mean that π' is pseudoknot-free and has barrier no greater than that of π. The sets $I^-_{\mathcal{A}}$ and $I^+_{\mathcal{B}}$ satisfy a useful property. To describe this, we say that i *conflicts with* j if $T[i] = -a$, $T[j] = +b$, arcs a and b cross and $j \in [i, i']$ where $i' = \text{partner}(i)$. For any $I \subseteq I^-_{\mathcal{A}}$, let $\text{conflict}(I)$ be the subset of $[1, m]$ that conflicts with indices in I. Note that if i conflicts with j, it must be that $[j, \text{partner}(j)] \subset [i, \text{partner}(i)]$. The useful property we have is that $\text{conflict}(I^-_{\mathcal{A}}) \subseteq I^+_{\mathcal{B}}$.

We initially set $I = I^-_{\mathcal{A}}$ and $J = I^+_{\mathcal{B}}$. We will cull sets I and J until $T(I, J)$ is a valid transformation sequence. After each culling step sets I and J will satisfy the following two properties: (a) $|I| > |J|$ and (b) conflict$(I) \subseteq J$. Before each culling step (including the first one), we have $|I| \geq |J|$ and conflict$(I) \subseteq J$.

A culling step is as follows. For any $r \in [1, m]$ let I_r be the subset of $I \cap [1, r]$ whose partners are after position r and let J_r be the subset of $J \cap [1, r]$ whose partners are after position r. Suppose that for some r,

$$|I_r| < |J_r|. \tag{1}$$

Let $I' = I - I_r$ and let $J' = J - J_r$. The sets I' and J' are results of the culling step. If there is no r such that $|I_r| < |J_r|$, no culling step is performed.

We will show that after the culling step, I' and J' satisfy properties (a) and (b). We immediately have that $|I'| > |J'|$ because of the fact that the removed subsets satisfy inequality (1) and that $|I| \geq |J|$. For property (b), we have that conflict$(I') \subseteq$ conflict$(I) \subseteq J$. Hence, to prove that conflict$(I') \subseteq J'$ it is enough to show that conflict$(I') \cap J_r$ is empty. Assume to the contrary that some $i \in I'$ conflicts with $j \in J_r$. Since $j \in J_r$, $r \in [j, j']$, where $j' =$ partner(j). Since i and j conflict, $[j, j'] \subset [i, i']$, where $i' =$ partner(i). Hence, $r \in [i, i']$, a contradiction with the assumption that $i \in I' = I - I_r$.

Now set $I = I'$ and $J = J'$. Repeat the culling step for the resulting (new) sets I and J, until there no longer is any r with $|I_r| < |J_r|$. Note that there can only be a finite number of culling steps, since each removes a non-empty subset from J (since, $|J_r| > |I_r|$, J_r, the set which is removed from J, is non-empty). Thus, at the end of the culling process, we have sets I and J with the following properties:

(1) $I \cup J$ is non-empty,
(2) conflict$(I) \subseteq J$ and
(3) for all $i \in [1, m]$, $|I_i| \geq |J_i|$.

Since $I \cup J$ is non-empty, $T(I, J)$ has fewer repeats than does T. Using property (2) we can show that the pathway π' corresponding $T(I, J)$ is pseudoknot free. Finally, property (3) implies that the energy barrier of π' is at most that of π (details will appear in the full paper). □

We have shown that the direct pseudoknot-free energy barrier problem is NP-complete.[16] The above result implies the following corollary.

Corollary 2.1. *The direct-with-repeats pseudoknot-free energy barrier problem is NP-complete.*

Example.

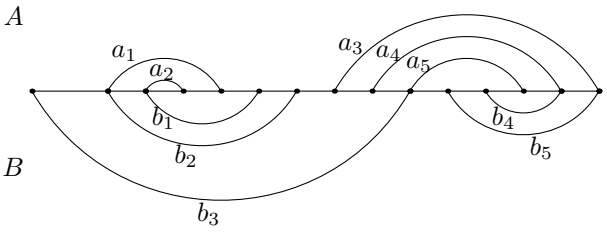


Fig. 2: Initial structure $\mathcal{A}$ and final structure $\mathcal{B}$

Consider the pair of structures $(\mathcal{A}, \mathcal{B})$ in Fig. 2 and the following transformation sequence for a pathway from $\mathcal{A}$ to $\mathcal{B}$:

$-a_1$	$-a_2$	$+b_1$	$+b_2$	$-b_1$	$+a_2$	$-b_2$	$-a_3$	$-a_4$	$+b_2$	$-a_2$	$+b_1$
$T[1]$	$T[2]$	$T[3]$	$T[4]$	$T[5]$	$T[6]$	$T[7]$	$T[8]$	$T[9]$	$T[10]$	$T[11]$	$T[12]$
$-a_5$	$+b_3$	$-b_3$	$+a_3$	$+a_4$	$+a_5$	$-a_3$	$-a_4$	$-a_5$	$+b_3$	$+b_4$	$+b_5$
$T[13]$	$T[14]$	$T[15]$	$T[16]$	$T[17]$	$T[18]$	$T[19]$	$T[20]$	$T[21]$	$T[22]$	$T[23]$	$T[24]$

We begin by identifying the sets $I^-_{\mathcal{A}} = \{2, 8, 9, 13\}$ and $I^+_{\mathcal{B}} = \{3, 4, 14\}$. Now, we find partners for the indexes in these sets. First, $I^-_{\mathcal{A}}$: partner$(2) = 6$, partner$(8) = 16$, partner$(9) = 17$, and partner$(13) = 18$. Now, $I^+_{\mathcal{B}}$: partner$(3) = 5$, partner$(4) = 7$ and partner$(14) = 15$.

Initially, we set $I = I^-_{\mathcal{A}} = \{2, 8, 9, 13\}$ and $J = J^+_{\mathcal{B}} = \{3, 4, 14\}$. Note that the conditions $|I| = 4 > 3 = |J|$ and conflict$(I) = \{14\} \subseteq J$ are true before the first culling step. Now, there is an r such that $|I_r| < |J_r|$: in particular, if $r = 7$, $I_r = \{2\}$ and $J_r = \{3, 4\}$. Hence, the new I is set to $I - I_r = \{8, 9, 13\}$ and the new J to $J - J_r = \{14\}$. We can now set $I = I'$ and $J = J'$. After this step we can see that I is non-empty, conflict$(I) = \{14\} \subseteq J$ and for all $i \in [1, m], |I_i| \geq |J_i|$ since the only element in J occurs after all the elements in I. We now replace operations at positions $I \cup \text{partner}(I) \cup J \cup \text{partner}(J)$ with no-ops. We will denote no-ops by ϵ. The transformation sequence is now:

$-a_1$	$-a_2$	$+b_1$	$+b_2$	$-b_1$	$+a_2$	$-b_2$	ϵ	ϵ	$+b_2$	$-a_2$	$+b_1$
$T[1]$	$T[2]$	$T[3]$	$T[4]$	$T[5]$	$T[6]$	$T[7]$	$T[8]$	$T[9]$	$T[10]$	$T[11]$	$T[12]$
ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	$-a_3$	$-a_4$	$-a_5$	$+b_3$	$+b_4$	$+b_5$
$T[13]$	$T[14]$	$T[15]$	$T[16]$	$T[17]$	$T[18]$	$T[19]$	$T[20]$	$T[21]$	$T[22]$	$T[23]$	$T[24]$

Once again, we identify the sets $I^-_{\mathcal{A}} = \{2\}$ and $I^+_{\mathcal{B}} = \{3, 4\}$. Since $|I^-_{\mathcal{A}}| < |I^+_{\mathcal{B}}|$ we are now in Case 2. We proceed by reversing the transformation sequence and replacing each + with a −, and vice versa to obtain a transformation sequence $\bar{T}$ from $\bar{\mathcal{A}} = \mathcal{B}$ to $\bar{\mathcal{B}} = \mathcal{A}$. This results in the following transformation sequence:

$-b_5$	$-b_4$	$-b_3$	$+a_5$	$+a_4$	$+a_3$	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
$T[1]$	$T[2]$	$T[3]$	$T[4]$	$T[5]$	$T[6]$	$T[7]$	$T[8]$	$T[9]$	$T[10]$	$T[11]$	$T[12]$
$-b_1$	$+a_2$	$-b_2$	ϵ	ϵ	$+b_2$	$-a_2$	$+b_1$	$-b_2$	$-b_1$	$+a_2$	$+a_1$
$T[13]$	$T[14]$	$T[15]$	$T[16]$	$T[17]$	$T[18]$	$T[19]$	$T[20]$	$T[21]$	$T[22]$	$T[23]$	$T[24]$

Now our new $I^-_{\bar{\mathcal{A}}} = \{13, 15\}$ and $I^+_{\bar{\mathcal{B}}} = \{14\}$, so we are back in case 1. Next, we identify the partners of these repeats: partner$(13) = 20$, partner$(15) = 18$, and partner$(14) = 19$. As before, we initially set $I = I^-_{\bar{\mathcal{A}}} = \{13, 15\}$ and $J = I^+_{\bar{\mathcal{B}}} = \{14\}$. We know that there is no r such that $|I_r| < |J_r|$ because J's only element occurs after an element in I, so $|I_r|$ and $|J_r|$ need to be at least equal for any r. So, sets I and J do have all three expected properties: I is non-empty, conflict$(I) = \{14\} \subseteq J$ and for all $i \in [1, m], |I_i| \geq |J_i|$. We now replace operations at positions $I \cup \text{partner}(I) \cup J \cup \text{partner}(J)$ with no-ops, obtaining the new transformation sequence (here, for brevity, we collapse $T[7]$ through $T[20]$ since they are all no-ops:

$-b_5$	$-b_4$	$-b_3$	$+a_5$	$+a_4$	$+a_3$	ϵ	$-b_2$	$-b_1$	$+a_2$	$+a_1$
$T[1]$	$T[2]$	$T[3]$	$T[4]$	$T[5]$	$T[6]$	$T[7-20]$	$T[21]$	$T[22]$	$T[23]$	$T[24]$

Since we are interested in a pathway from $\mathcal{A}$ to $\mathcal{B}$, we need to reverse the transformation sequence and replace every + with a −.The transformation sequence is now repeat-free:

$-a_1$	$-a_2$	$+b_1$	$+b_2$	ϵ	$-a_3$	$-a_4$	$-a_5$	$+b_3$	$+b_4$	$+b_5$
$T[1]$	$T[2]$	$T[3]$	$T[4]$	$T[5-18]$	$T[19]$	$T[20]$	$T[21]$	$T[22]$	$T[23]$	$T[24]$

3. Algorithms for exactly computing energy barriers

In this section we describe our algorithms for finding the min-barrier pathway between two structures. We model the problem in terms of bipartite graphs.

For a pair of pseudoknot-free structures for the same RNA sequence, we define the *conflict graph* to be a bipartite graph $G = (A, B; E)$ where A is the set of arcs from the first structure, B is the set of arcs from the second structure and there is an edge in E between arc $a \in A$ and arc $b \in B$ if and only if a and b are crossing. Throughout, we denote the neighbours (in A) of a subset B' of B by $N(B')$. Also, we denote the subgraph of G induced by subsets $A' \subseteq A$ and $B' \subseteq B$ by $G/(A', B')$. We need a notion analogous to that of a pair of MFE structures in the context of bipartite graphs. We say that G is *pairwise-optimal* if the size

of the maximum independent set in the graph is $|A|(=|B|)$. If A and B are MFE structures then G must be pairwise-optimal; otherwise the largest independent in the conflict graph G would be a set of arcs with lower free energy than either A or B. We let $[i,j]$ represent the interval (set of integers) $\{i, i+1, \ldots, j\}$.

Let $G = (A, B; E)$ be a pairwise-optimal bipartite graph. A *set pathway* for G is a sequence of independent sets $S_0, \ldots, S_m$, each of which is a subset of $A \cup B$, such that (i) $S_0 = A$, (ii) $S_m = B$ and (iii) for every $i = 1, \ldots, m$, $|S_{i-1} \triangle S_i| = 1$ (the size of symmetric difference is one, i.e., at each step one arc is either added or removed). The *transformation sequence* corresponding to this set pathway is the sequence of singletons $S_0 \triangle S_1, \ldots, S_{m-1} \triangle S_m$. The set pathway is *direct* if its corresponding transformation sequence has no repeating elements. The *barrier* of the pathway (or its corresponding transformation sequence) is $k = \max_i |A| - |S_i|$. (Since A is a maximum independent set of $(A, B; E)$, it must be that $|A| - |S_i| \geq 0$ for all $i, 1 \leq i \leq m$.) We say that a set pathway is a $(\leq k)$-barrier set pathway or a k-barrier pathway if its barrier is $\leq k$ or $= k$, respectively. A *min-barrier* set pathway is a set pathway whose barrier is less than or equal to the barriers of any other set pathway for G. Consider the following problem:

Direct Set Barrier Problem. Given a pairwise-optimal bipartite graph $G = (A, B; E)$ and an integer k, find a direct set pathway with barrier at most k for G if one exists.

An instance of the Direct Energy Barrier Problem can be mapped to an instance of the Direct Set Barrier Problem by contructing its conflict graph. However, the Direct Set Barrier Problem is actually a more general problem since not every bipartite graph is realizable by a pair of pseudoknot-free structures.

Algorithm Overview. Our algorithm for the Direct Set Barrier Problem uses two key ideas. The first is a *splitting strategy*: if for some proper non-empty subset B_1 of B the induced subgraph $G/(A_1, B_1)$ is pairwise-optimal where $A_1 = N(B_1)$ then we can determine the solution for $(A, B; E)$ by recursively solving the problem on the induced subgraphs $G/(A_1, B_1)$ and $G/(A \setminus A_1, B \setminus B_1)$ and combining the solutions to the subproblems. Specifically, if $G/(A_1, B_1)$ is pairwise-optimal then we can show three properties described in the following lemma (due to space limitations we omit details of the proof of correctness of our algorithm here and elsewhere but these will be provided in the full paper):

Lemma 3.1. *Let $G = (A, B; E)$ be a pairwise-optimal bipartite graph and let $G_1 = G/(A_1, B_1)$ be pairwise-optimal where B_1 is a proper non-empty subset of B and $A_1 = N(B_1)$. Let $G' = G/(A \setminus A_1, B \setminus B_1)$. Then*

(1) G' is pairwise-optimal,
(2) if T_1 and T' are $(\leq k)$-barrier transformation sequences for G_1 and G' respectively then T_1, T' is a $(\leq k)$-barrier transformation sequence for G and
(3) G has a $(\leq k)$-barrier set pathway only if both G_1 and G' do.

For efficiency reasons, our implementation generalizes this splitting idea by splitting the problem into as many "minimal" pairwise-optimal subproblems as possible. We say that $(A, B; E)$ is *minimal pairwise-optimal* if $|A| = |B|$ and the only independent sets in $(A, B; E)$ of size $|A|$ are A and B.

The second key idea is a *cutting strategy* for reducing the size of minimal pairwise-optimal problem instances. We have developed two cutting strategies. The first, the *two-sided cutting strategy*, generates the subgraphs $G/(A \setminus \{a\}, B \setminus \{b\})$ for each choice of $a \in A$ and $b \in B$ and recursively solves each of the resulting subproblems with the barrier set to $k - 1$. The following lemma states that if we do this for all possible choices of a and b, we are guaranteed to find a $(\leq k)$-barrier set pathway for G if one exists.

Lemma 3.2. *Let $G = (A, B; E)$ be minimal pairwise-optimal. Then*

(1) $G/(A \setminus \{a\}, B \setminus \{b\})$ is pairwise-optimal for all $a \in A$ and $b \in B$,
(2) if $G/(A \setminus \{a\}, B \setminus \{b\})$ has a transformation sequence T' with barrier at most $k-1$ then $T = \{a\}, T', \{b\}$ is a transformation sequence for $(A, B; E)$ with barrier at most k and
(3) G has a transformation sequence with barrier at most k only if $G/(A \setminus \{a\}, B \setminus \{b\})$ has a transformation sequence with barrier at most $k - 1$ for some $a \in A$ and $b \in B$.

The second cutting strategy is a *one-sided cutting strategy*. Suppose that in some ($\leq k$)-barrier pathway for G, b is the first node of B removed in the corresponding transformation sequence T. Let A' be the set of nodes of A which conflict with b (i.e., $N(b') = A'$) and let B' be the set of nodes of B such that $N(B') \subseteq A'$. Then we can assume without loss of generality that all nodes of A' are first removed in T and immediately thereafter all nodes of B' are added. Abusing notation slightly, we can write that $T = -A', +B', \ldots$ where we mean that the singleton subsets of A' are listed first in an unspecified order, followed by the singleton subsets of B' in an unspecified order. Given A' and B', we'd like to use the algorithm recursively on the remaining subgraph $G/(A - A', B - B')$. However this subgraph is not pairwise-optimal and in fact $|A - A'| < |B - B'|$ (otherwise we would be able to apply the splitting strategy). To circumvent this problem we create a new bipartite graph $G'(b)$ by adding $k' = |A'| - |B'|$ "artificial" nodes to G and connecting them to all nodes in $B - B'$. To obtain a ($\leq k$)-barrier pathway for G, we then recursively obtain the transformation sequence T' of a ($\leq k$)-barrier pathway for the new graph $G'(b)$, remove the singletons of T' that involve the k' extra nodes and finally append what's left of T' to A', B'. The following lemma states that if we do this for all possible choices for b, we are guaranteed to find a ($\leq k$)-barrier set pathway for G if one exists.

Lemma 3.3. *Let $G = (A, B; E)$ be minimal pairwise-optimal. Then*

(1) $G'(b)$ is pairwise-optimal for all $b \in B$,
(2) if T' is a ($\leq k$)-barrier transformation sequence for $G'(b)$ for some b then the transformation sequence T obtained from T' via the cutting strategy is a ($\leq k$)-barrier transformation sequence for G and
(3) G has ($\leq k$)-barrier transformation sequence only if for some $b \in B$, $G'(b)$ has a ($\leq k$)-barrier transformation sequence.

The **Direct-SetBarrier** algorithm, presented as Algorithm 3.1 below, incorporates the splitting and two-sided cutting strategy. First, the input graph $(A, B; E)$ is split to yield one or more subproblems (line 3), via a call to procedure Split$(A, B; E)$. We describe later how this procedure works. The subproblems produced by this split procedure cannot be split further, so our cutting strategy reduces these to smaller subproblems (lines 12-19), unless they are already trivial to solve (lines 9-10), and concatenates the solutions to the subproblems. By changing the inner loop, the one-sided cutting strategy can be implemented.

In the rest of this section we provide more details of the Split procedure. We conclude with a note on the theoretical run-time and space complexity of the algorithm. In Section 4 we provide an empirical analysis of the performance of our algorithm.

Details of the Split procedure. Given as input a pairwise-optimal bipartite graph $G = (A, B; E)$, our Split procedure produces a sequence of non-empty bipartite graphs $G_1, G_2, \ldots, G_p$ via the following steps.

First find a maximum matching $\mathcal{M}$ in $(A, B; E)$ using the Hopcroft-Karp algorithm.[17] (As we note in our correctness proof, well-known results on bipartite graphs show that such a matching exists.) Second, create the *precedence graph* for G and $\mathcal{M}$. By precedence graph, we mean the directed bipartite graph $D = (A, B; E')$ where $E' = \{(b, a) | b \in B \wedge (b, a) \in E\} \cup \{(a, b) | a \in A \wedge (a, b) \in \mathcal{M}\}$. Third, using Tarjan's algorithm,[18] find the strongly connected components in the precedence graph. Finally, create a total order of these components in a manner that is consistent with their topological ordering in the associated *condensation graph*, i.e. the directed acyclic graph in which each strongly connected component is condensed into a single node. Let (A_i, B_i) be the set of nodes in the ith component in this total ordering. Then the graph G_i is chosen to be the subgraph of $G/(A - \cup_{j=1}^{i-1} A_j, B - \cup_{j=1}^{i-1} B_j)$ induced by the nodes (A_i, B_i). The next lemma summarizes important properties of the Split procedure.

Lemma 3.4. *Given as input a pairwise-optimal bipartite graph $G = (A, B; E)$ the* Split *procedure produces a sequence $G_1 = (A_1, B_1; E_1), G_2 = (A_2, B_2; E_2), \ldots, G_p = (A_p, B_p; E_p)$ such that*

(1) G_i is minimal pairwise-optimal for each $i, 1 \leq i \leq p$ and
(2) $A_i = N(B_i)$ in the graph $G'_i = G/(A - \cup_{j=1}^{i-1} A_j, B - \cup_{j=1}^{i-1} B_j)$ for all i, $1 \leq i \leq p$.

Algorithm 3.1 Direct Set Barrier Algorithm **Direct-SetBarrier**$((A, B; E), k)$

```
// INPUT: a non-empty pairwise-optimal bipartite graph (A, B; E) and a barrier k ≥ 0
// OUTPUT: a direct transformation sequence from A to B with barrier at most k, or "no solution"
call procedure SPLIT(A, B; E) to obtain subproblems G_1 = (A_1, B_1; E_1), ..., G_p = (A_p, B_p; E_p)
T ← ∅ // empty sequence
if k ≤ 0 then
  return "no solution"
else
  for i = 1 to p do
    if |A_i| ≤ k then
      append A_i, B_i to T
    else
      // the inner loop:
      for all a ∈ A_i and b ∈ B_i do
        call DIRECT-SETBARRIER(G_i/(A_i \ {a}, B_i \ {b}), k − 1)
        if a sequence T' was returned then
          append {a}, T', {b} to T
          exit the inner loop
        end if
      end for
      if no transformation sequence was found in the inner loop then
        return "no solution"
      end if
    end if
  end for
  return T
end if
```

Run-time and space complexity. The run-time of the SPLIT$(A, B; E)$ procedure is dominated by the time needed to find a maximum matching in a bipartite graph. This time is $O(n^{2.5})$, or more precisely $O(n^{1/2}m)$ using the Hopcroft-Karp algorithm[17] where $n = |A| = |B|$ and $m = |E|$. The remaining steps of the SPLIT$(A, B; E)$ procedure take $O(n + m)$ time.

The DIRECT-SETBARRIER$((A, B; E), k)$ algorithm with the two-sided cutting strategy has theoretical run-time complexity $O(n^{2k+2.5})$. Recall that k is the input which specifies the allowable barrier, i.e. the algorithm determines whether $(A, B; E)$ has a set with barrier at most k. The worst case arises when every call to the SPLIT procedure produces just one subproblem, because in this case we are not able to split a problem (or its recursive subproblems) into smaller independent problems. In this case the inner loop is called $O(n^2)$ times on a subproblem with allowable barrier $k - 1$ and the recursion bottoms out when k reaches 0. We therefore have $O(n^{2k})$ calls to the DIRECT-SETBARRIER$((A, B; E), k)$ algorithm (Algorithm 3.1), each taking time $O(n^{2.5})$ for a total running time of $O(n^{2k+2.5})$. Note that the run-time is exponential in k. However, since the problem is NP-hard,[16] we cannot hope for an algorithm whose run-time is polynomial in n, m and k in the worst case.

The worst-case run-time complexity of the algorithm with the one-sided cutting strategy is $n^{O(n)}$ since in this case k is not reduced on recursive calls and so the depth of recursion depends on the degree to which the problem size is reduced at each level of recursion.

Thus which cutting strategy is best depends on the input. When k is small, the two-sided cutting strategy dominates but when the one-sided cutting strategy succeeds in reducing the problem size consistently, then the latter is more effective. Our empirical analysis sheds more insight on the trade-offs.

All of the techniques used by the algorithm use $O(n + m)$ space. Finally we note that Direct-SetBarrier takes as input a specific candidate k for the min-barrier. To find the barrier when k is unknown, we can use a divide and conquer approach. Initially, we know k is in the range $[1, |A|]$. We can try $k = |A|/2$; if we find a solution we then we know the range is $[1, |A|/2]$. If not, we know the range is $[|A|/2 + 1, k]$. We continue to reduce the range by a factor of 2 until the range is 1, at which point we know the min-barrier. This increases the run-time by a factor that is logarithmic in the input size.

4. Empirical Results

We implemented both of our algorithms for the Direct-SetBarrier problem, in order to study their efficiency in practice on biologically motivated data. Here we describe our experimental setup and protocol for generating problem instances and describe our results on the performance of our algorithm.

Implementation and experimental environment. Both algorithms were coded in C++ and compiled using g++ (GCC version 4.2.1). All experiments were run on our reference PCs with 2.4Ghz Intel Pentium IV processors with 256KB L2 cache and 1GB RAM, running SUSE Linux version 10.3.

Generation of problem instances. With the motivation of studying algorithm performance across a variety of problem instances, we randomly sampled five sequences for each of four different classes of non-coding RNA—*Transfer RNA, Transfer Messenger RNA, Ribonuclease P RNA,* and *5S Ribosomal RNA,*—found in the RNA STRAND database.[19] For each sequence, five MFE structures—with respect to number of base pairs—were determined using a modified version of the Nussinov-Jacobsen algorithm.[20] The modified algorithm stored all optimal paths within the traceback matrix. In this way, we were able to randomly sample five different MFE structures for the same sequence. Identical structures were discarded. Every possible pairing of structures for the same sequence formed a new problem instance. Thus, ten problem instances were created for each sequence, resulting in 200 problem instances overall. The distribution of sequence length and the resulting number of conflicting base pairs between paired structures can be seen in Fig. 3. In general, and as expected, the number of conflicting bases pairs increases with sequence length.

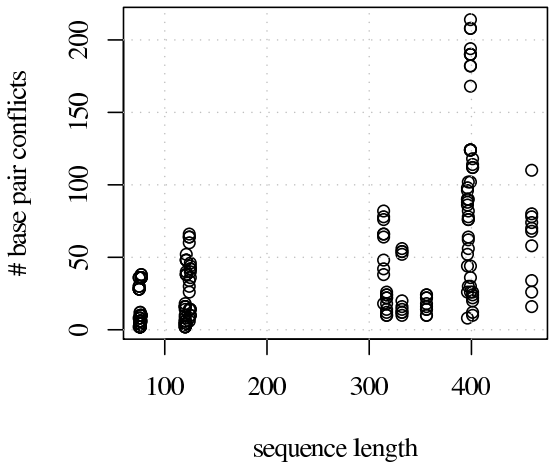


Fig. 3: Distribution of conflicting base pairs for generated problem instances.

Algorithm runtime performance. Both algorithms were run for a maximum of 1 CPU hour on each of the 200 hundred problem instances. The $n^{O(n)}$ algorithm found solutions to 183 instances, while the $O(n^{2k+2.5})$ algorithm found solutions to 184 instances. Interestingly, the $n^{O(n)}$ algorithm found solutions to three instances not found by the $O(n^{2k+2.5})$ algorithm; likewise, four instances were found by the $O(n^{2k+2.5})$

algorithm not found by the $n^{O(n)}$ algorithm. Of the instances that were solved, optimal barriers were found within 1 CPU second by both algorithms in 90% of the cases with barrier height ranging from 1 to 8. The barrier of harder instances ranged from 6 to 11, with a mean of 9. In general, the $O(n^{2k+2.5})$ algorithm was the best performing for harder instances. However, as can be seen in Fig. 4, both algorithms excelled for certain instances relative to one another.

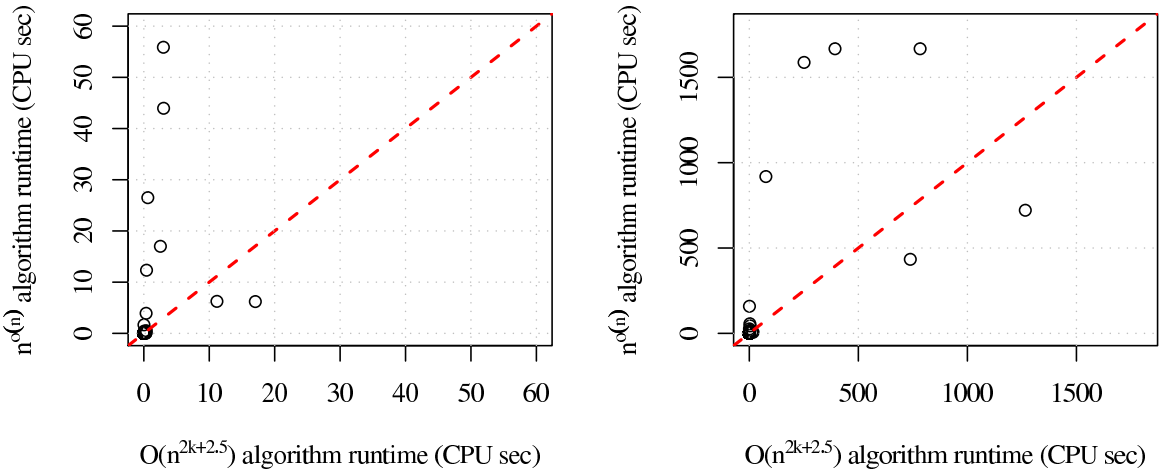


Fig. 4: The required time to find an optimal barrier pathway shown for two time scales.

The instances which failed to be solved within our cut-off time tended to have the highest number of conflicting base pairs. We also found that the instances which failed to be solved tended to have the largest minimally pairwise-optimal subproblems generated by the Split procedure. For each problem instance, we recorded the size of the maximum subproblem, as well as the average size of all subproblems, produced by the Split procedure at the top level of recursion, We measured size as number of base pairs. Fig. 5 shows the frequency of problem instances which have given maximum (left) and average (right) subproblem size. The problem instances which have maximum subproblems of size 200 or more were those that failed to be solved. Alternative methods for splitting or recursing on such subproblems would clearly be valuable.

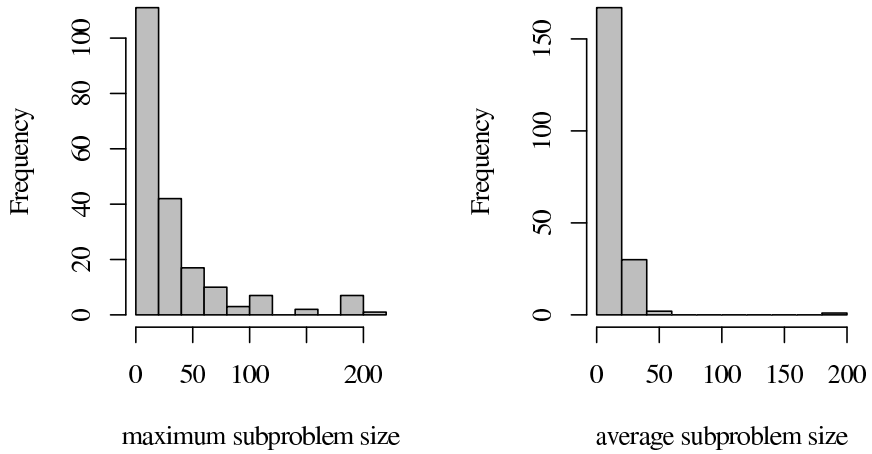


Fig. 5: Frequency of maximum (left) and average (right) subproblem sizes, measured as number of base pairs in the subproblem produced by the first call to the Split procedure for a given instance. The max and average are taken over all subproblems generated for a given instance.

5. Conclusion

We have presented new approaches for calculating energy barriers between RNA secondary structures using classical bipartite graph algorithms, and can prove that our methods find the min-barrier direct folding pathway between two MFE secondary structures. Our algorithms run very efficiently on structures of length up to 300nt and solved the vast majority of our generated instances of length up to 500nt.

Our methods show much promise for further development. They can be generalized to find energy barriers between non-MFE structures by exploiting the introduction of "artificial" nodes as in the one-sided cutting strategy. Our algorithms are highly amenable to parallelization: subproblems generated both by the SPLIT procedure and the cutting strategy can be solved independently. Furthermore, there is more potential for creative means to split or otherwise reduce the size of a problem.

Even on instances that are intractable for our algorithm, preliminary analysis suggests that our algorithm might yield tight approximate bounds. For example, while the algorithm may be slow on some long instances when given the "true" min-barrier k, it can be fast when given $k-1$ and $k+1$, thereby narrowing the range to 2. The ability to establish a narrow range could be sufficient for some methods that approximate RNA energy landscapes.[2,11] We plan to investigate this further.

Other important directions for future work are to determine whether low-barrier pathways produced by our methods for the simple energy model are competitive with pathways found by state-of-the-art heuristics using a thermodynamic energy model, or whether our techniques can be extended to obtain exact energy barriers with more realistic energy models. Also of interest would be a follow-up of the Morgan and Higgs study to determine how barrier heights scale with sequence length, particularly for long sequences. Further testing of our method on more biologically-relevant pairs of structures will be necessary to answer these questions. Finally, development of efficient exact methods for determining min-barrier *indirect* pathways, i.e. pathways that allow temporary edges, is an interesting unresolved challenge.

Acknowledgments

We thank the anonymous reviewers for their insightful comments and suggestions for future work.

References

1. T. Baumstark, A. R. Schroder and D. Riesner, *EMBO J.* **16**, 599 (1997).
2. M. Geis, C. Flamm, M. T. Wolfinger, A. Tanzer, I. L. Hofacker, M. Middendorf, C. Mandl, P. F. Stadler and C. Thurner, *J. Mol. Biol.* **379**, 160 (2008).
3. A. Roth and R. R. Breaker, *Annual Rev. Biochem.* **78**, 305 (2009).
4. E. A. Schultes and D. P. Bartel, *Science* **289**, 448(July 2000).
5. T. B. J and R. R. Breaker, *Curr. Opin. Struct. Biol.* **15**, p. 342 (2005).
6. C. Yanofsky, *RNA* **13**, 1141 (2007).
7. S.-J. Chen and K. A. Dill, *Proc. Nat. Acad. Sci.* **97**, 646(January 2000).
8. R. Russell, X. Zhuang, H. Babcock, I. Millett, S. Doniach, S. Chu and D. Herschlag, *Proc. Nat. Acad. Sci.* **99**, 155 (2002).
9. I. Shcherbakova, S. Mitra, A. Laederach and M. Brenowitz, *Curr. Opin. Chem. Biol.* **12**, 655 (2008).
10. D. K. Treiber and J. R. Williamson, *Curr. Opin. Struc. Biol.* **11**, 309 (2001).
11. X. Tang, S. Thomas, L. Tapia, D. P. Giedroc and N. M. Amato, *J. Mol. Biol.* **381**, 1055 (2008).
12. C. Flamm, I. L. Hofacker, P. F. Stadler and M. T. Wolfinger, *Zeitschrift für Physikalische Chemie* **216**, 155 (2002).
13. S. R. Morgan and P. G. Higgs, *J. Phys. A: Math. Gen.* **31**, 3153 (1998).
14. M. T. Wolfinger, The energy landscape of RNA folding, Master's thesis, University Vienna (2001).
15. C. Flamm, I. L. Hofacker, S. Maurer-Stroh, P. F. Stadler and M. Zehl, *RNA* **7**, 254 (2001).
16. J. Maňuch, C. Thachuk, L. Stacho and A. Condon, *Proc. of the 15th Intl. Meeting on DNA Computing and Molecular Programming (DNA15)* (2009).
17. J. E. Hopcroft and R. M. Karp, *SIAM J. Comput.* **2**, 225 (1973).
18. R. Tarjan, *SIAM J. Comput.* **1**, 146 (1972).
19. M. Andronescu, V. Bereg, H. H. Hoos and A. Condon, *BMC Bioinformatics* **9**, 340 (2008).
20. R. Nussinov and A. B. Jacobson, *Proceedings of the National Academy of Sciences of the United States of America* **77**, 6309 (1980).

DYNAMICS OF BIOLOGICAL NETWORKS: SESSION INTRODUCTION

TANYA Y. BERGER-WOLF[1]
Department of Computer Science, University of Illinois at Chicago
Chicago IL 60607, USA

TERESA M. PRZYTYCKA[2]
National Center of Biotechnology Information, NLM, NIH
Bethesda MD 20814, USA

MONA SINGH[3]
Department of Computer Science, Lewis Sigler Institute for Integrative Genomics
Princeton University, Princeton NJ 08544, USA

DONNA K. SLONIM[4]
Department of Computer Science, Tufts University
Department of Pathology, Tufts University School of Medicine
Medford, MA 02155

Biological network analysis has become a central component of computational and systems biology. Because such analysis provides a unifying language to describe relations within complex systems, it has played an increasingly important role in understanding physiological function. Significant efforts have focused on analyzing and inferring the topology and structure of cellular networks and on relating them to cellular function and organization. However, much of this work has taken a static view of cellular networks, despite the knowledge that biological networks can change with time, context, and conditions. We introduce this session on The Dynamics of Biological Networks to encourage and support the development of computational methods that elucidate the dynamic interactome.

Biological networks encompass many types of variation. Temporal variation can be inferred via experimentally determined large-scale (static) cellular networks, along with other high-throughput experimental data sets that provide snapshots of biological systems at different times and conditions. These data are often integrated with static interaction data (e.g. protein-protein, domain-domain, or regulatory interactions), time- or environment-dependent expression data, protein localization data, or other contextual information. Temporal variation also operates at the evolutionary scale and can be inferred via comparison of biological networks across species. An evolutionary perspective on network dynamics can help shed light on molecular function and

[1] Work partially supported by NSF CAREER grant IIS-0747369 and NSF grants IIS-0705822 and IIS-0612044
[2] Supported by the Intramural Research Program of the National Institutes of Health and National Library of Medicine
[3] Work partially supported by NSF grant CCF-0542187, NIH grant GM076275 and NIH Center of Excellence grant P50 GM071508
[4] Supported in part by NIH grants LM009411 and HD058880.

behavior as well as on the phylogenetic relationships between species. Contextual variation overlaps heavily with temporal variation, but focuses more specifically on characterizing *reactive* variation and the conditions that cause it. Studying context may also encompass examining sequence or genetic variation within a population of contemporaries and exploring how that variation affects network topology and function.

Uncovering the dynamic nature of cellular networks has clear relevance to human health, as defects in signaling and regulatory pathways are associated with many serious diseases, such as cancer. Correlating changes in genotype with changes in disease phenotype and studying these changes in the context of protein-protein interaction networks, signaling networks and transcriptional networks has begun to provide new understating of mechanism of complex diseases as well as help uncover markers for disease recognition and classification.

As in the previous year, this year's session on Dynamics of Biological Networks brings together scientists working on various aspects of the dynamic nature of biological networks. The session includes an invited talk, six contributed papers, and a panel discussion. The invited talk, by Edward Marcotte, relates gene networks to disease and evolutionary dynamics. Specifically, he will discuss new methods for linking genes to traits that reveal surprising disease models and that are intimately connected to evolutionarily conserved gene networks.

The papers selected for presentation address a broad spectrum of problems related to network dynamics. Several papers explore context specificity. In particular, Fang, *et al.,* directly link gene expression data with context or medical conditions. They present an algorithm that finds sets of genes that are highly co-expressed in many samples from a given context (such as cells grown in a nutrient-deficient condition), but that are *not* co-expressed in most samples in a different context (such as cells grown in normal media). Their approach defines gene sets likely to form context-specific pathways.

Many approaches to finding functional modules in biological networks rely on graph-partitioning algorithms that optimize a single objective function. In contrast, the paper by Navlakha, *et al.* looks at ensembles of optimal and near-optimal partitions to determine node centrality in modules and module variability. The variation observed between the different partitions may shed light on context-specific variations in the composition of functional modules.

Chowdhury and Koyutürk also focus on finding functional modules or subnetworks of protein-protein interaction networks that are coordinately dysregulated in disease. Their work builds on the idea that a group of genes that are individually only marginally dysregulated with respect to the phenotype might provide a clearer signal when considered together.

Building and evaluating dynamic network models is another theme of the session. Context specificity is also an underpinning of the work of Morcos *et al.,* who use a combination of methods to predict protein and domain interactions and apply belief propagation to provide support for a theoretical model of mobility for *M. xanthus*.

Venkatraman *et al.* introduce a mathematical model of activation of Plasmin and urokinase-type plasminogen proteases regulating the extracellular environment. The computational simulation of

the dynamics of their model predicts that the system exhibits bistable behavior, a prediction that is then validated experimentally.

A final theme of the session focuses on studying dynamics at an evolutionary time scale. Chindelevitch, *et al.,* present a new algorithm for global alignment of protein-protein interaction networks. The edge-swapping technique that they use to search the space of possible alignments is not only efficient in practice, but also helps to generate hypotheses about the evolutionary dynamics of the protein interaction networks between the described species.

In addition to the oral presentations, one paper has been selected for presentation in the proceedings. In this work, Przulj *et al.* describe a new model for the evolution of protein-protein interaction networks, and compare a number of different evolutionary dynamic models on a range of data sets.

Though the topics are varied, all of the work presented in this session shares the common goal of relating system dynamics to our understanding and modeling of biological networks. We hope that this session stimulates further research into characterizing and interpreting the dynamic interactome.

Acknowledgements
We are grateful to those who submitted manuscripts for consideration for inclusion in this session, and we thank the reviewers for their valuable expertise and time throughout the peer review process.

LOCAL OPTIMIZATION FOR GLOBAL ALIGNMENT OF PROTEIN INTERACTION NETWORKS

LEONID CHINDELEVITCH CHUNG-SHOU LIAO * BONNIE BERGER †

Department of Mathematics and Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139.
E-mail: {leonidus, shou, bab}@csail.mit.edu

We propose a novel algorithm, PISwap, for computing global pairwise alignments of protein interaction networks, based on a local optimization heuristic that has previously demonstrated its effectiveness for a variety of other NP-hard problems, such as the Traveling Salesman Problem. Our algorithm begins with a sequence-based network alignment and then iteratively adjusts the alignment by incorporating network structure information. It has a worst-case pseudo-polynomial running-time bound and is very efficient in practice. It is shown to produce improved alignments in several well-studied cases. In addition, the flexible nature of this algorithm makes it suitable for different applications of network alignments. Finally, this algorithm can yield interesting insights into the evolutionary history of the compared species.

Keywords: Network alignment; Local optimization; Functional orthology

1. Introduction

Ever since high-throughput experimental screening techniques such as yeast two-hybrid analysis,[1] mass spectrometry,[2] and TAP[3] made protein interaction networks available for several species, efforts have been made in the bioinformatics community to extract useful biological information from these networks. One important goal has been to produce accurate alignments of two or more of these networks, with the expectation that this would help in establishing the biological function of unknown proteins by exhibiting their correspondence with the proteins of another species with known biological function, and providing insight into evolutionary dynamics.

Algorithms for network alignment can be broadly separated into two distinct categories: *local* and *global* algorithms. The distinction is similar to the one made for sequence alignment algorithms. More specifically, local network alignment[4] is concerned with identifying a subnetwork of one species closely matching a subnetwork of another species (or having a given topology). Typically, multiple closely matching subnetworks are identified by such algorithms, which may be mutually inconsistent.[5] On the other hand, global network alignment algorithms[5] attempt to match two or more networks as a whole, and their output is a single mapping between the nodes of the networks. In the present paper, which deals with the global alignment problem, we view this mapping as a bipartite matching, where the nodes on one side of the matching are the proteins from one network, and the nodes on the other side, the ones from the other network.

1.1. *Contribution*

We propose a novel method, based on local optimization, for computing pairwise alignments of protein interaction networks. The method begins by identifying an optimal alignment based purely on sequence data, which correctly determines functionally orthologous proteins in many, but not all, cases. In order to adjust this initial alignment, the algorithm uses the intuition that conserved interactions can compensate for matching proteins whose sequences are not particularly similar to one another. In this way, the topology of the networks is taken into account, and information is propagated from each node to its neighbors.

Using the protein interaction networks available for three species: yeast, fly, and worm, we perform pairwise alignments on each pair. The results compare favorably to those produced by other methods. Furthermore, we suggest that the algorithm can be used to produce several alignments with almost no

*Also at the Institute of Information Science (IIS), Academia Sinica, Taipei 115, Taiwan.
†Corresponding Author

additional cost, or fine-tuned by combining it with other algorithms for partitioning or clustering protein interaction networks. Finally, we explain how specific information produced by this algorithm can produce insights into the evolutionary dynamics of protein interactions.

1.2. *Related Work*

A number of techniques for the problem of PPI network alignment exist. These include NetworkBLAST-M,[6] Graemlin 2.0,[7] IsoRank,[5] IsoRankN[8] and PATH,[9] though a number of other techniques exist as well.[4,10–15] Some of these methods can handle more than two networks. NetworkBLAST-M computes a local alignment by greedily finding regions of high local conservation based on inferred phylogeny. Graemlin 2.0, in contrast, computes a global alignment by training how to infer networks from phylogenetic relationships on a known set of alignments, then optimizing the learned objective function on the set of all networks. IsoRank uses spectral graph theory to first find pairwise alignment scores across all pairs of networks, obtained by a spectral method on a product graph; IsoRank then uses these scores in a greedy algorithm to produce the final alignment. The more recent IsoRankN uses a different method of spectral clustering on the induced graph of pairwise alignment scores. As pointed out in a recent paper,[9] one of the main difficulties faced by network alignment algorithms is the lack of an accurate and reliable gold standard for evaluation purposes.

2. Problem Formulation

We consider the global alignment of a pair of protein-protein-interaction (PPI) networks. Each network is represented by a graph whose vertices correspond to proteins, and there is an undirected edge between two vertices if the corresponding proteins are found to interact with each other.

Given a pair of PPI networks and a list of pairwise sequence similarities between proteins in the two networks computed according to some criterion, the specific aim of global alignment is to find a mapping between the proteins of the two networks that best represents conserved biological function. We formulate this network alignment problem as a graph-theoretical problem.

Let $G_X = (X, E_X)$ and $G_Y = (Y, E_Y)$ be two PPI networks in which $e_x = (x, x') \in E_X$ and $e_y = (y, y') \in E_Y$, with x, $x' \in X$ and y, $y' \in Y$, represent interaction between two proteins in G_X and G_Y respectively.

Suppose $G = (X \cup Y, E)$ is an edge-weighted bipartite graph in which each edge $e = (x, y) \in E$ is associated with a nonnegative edge weight $s(e)$, which represents the sequence similarity between $x \in X$ and $y \in Y$. That is, $s : E \to \mathrm{Z}^+$ denotes a sequence similarity function on the edges of G, where sequence similarity on an edge, a pair of proteins, could be, for instance, the BLAST Bit-value of the sequences as retrieved from Ensembl.[19]

A matching $M \subseteq E$ of G is defined to be a subset of edges such that no two edges in M share an endpoint. In addition, given a matching M, we let a function $t : E \to \mathrm{Z}^+$ be a topology similarity function with respect to M if for an edge $e = (x, y)$, $t(e)$ represents the topology similarities between the neighborhoods of $x \in X$ and $y \in Y$. The topology similarity function represents the number of edges in G_X and G_Y conserved by the matching M.

More precisely, for an edge $e = (x, y)$, $t(e)$ is the number of edges between the neighborhoods of x and y, $N(x)$ of G_X and $N(y)$ of G_Y respectively, which are also in the matching M, i.e. $t(e) = |\{(x', y') \in M | x' \in N(x) \text{ and } y' \in N(y)\}|$.

Our objective is to find a matching M that maximizes the following weight function w:

$$w(M) = \sum_{e \in M} \{\alpha t(e) + (1 - \alpha) s(e)\}, \tag{1}$$

where $\alpha \in [0, 1]$ is a parameter controlling the importance of the network topology similarities relative to sequence similarities. This is equivalent to the objective function introduced by Singh et al.[5]

The above weight function contains two terms: the topology similarity function t and the sequence similarity function s. Tuning the parameter α allows us to change the relative importance of PPI network data in finding the optimal global alignment. For instance, $\alpha = 0$ implies no network data will be used, while $\alpha = 1$ indicates only network data will be used.

We will use a local search approach to pairwise PPI network alignment in a manner similar to `2-Opt` as follows: when given a maximum weighted bipartite matching M^* in $G = (X \cup Y, E)$, we define a subset $\text{prefer}_Y(x)$ for each $x \in X$, which consists of the c highest-weighted neighbors of x in Y (where the weight of a neighbor of x is given by its sequence similarity to x). Similarly, for every $y \in Y$, a vertex subset $\text{prefer}_X(y) \subseteq X$ is similarly defined to consist of the c highest-weighted neighbors of y in X. Here, c is some relatively small integer chosen ahead of time. It can be shown[20] that $c = 20$ suffices for most practical applications.

Our aim is to repeatedly find a candidate $e' = (u, v)$, $v \in \text{prefer}_Y(x)$, $u \in \text{prefer}_X(y)$ to swap with $e = (x, y)$, where e, $e' \in M^*$, such that the weight of the new matching, $w((M^* \setminus \{e, e'\}) \cup \{e_1, e_2\})$, where $e_1 = (x, v)$ and $e_2 = (u, y)$ are the edges obtained by swapping e and e', is higher than $w(M^*)$.

Just as in IsoRank,[5] we seek to maximize an objective function consisting of two terms, one accounting for sequence similarity between the proteins that get matched to each other, and the other one, for the number of interactions preserved by the matching. Our search space is the set of all matchings. We use the parameter $\alpha \in [0, 1]$ to control the relative importance of the topology information with respect to the sequence information. This formulation of the problem is known to be NP-hard, and even APX-hard,[16] which means that it does not admit a polynomial-time approximation scheme unless P = NP. Since the size of the search space (i.e. the number of possible matchings) grows exponentially with the number of proteins in each network, we use a local search technique adapted from other NP-hard optimization problems, which is described in detail in the following section.

3. Algorithmic results

The main idea of our algorithm for the global alignment of pairwise PPI networks is a two-phase approach to searching for a matching that maximizes our objective function. The special case of our problem with $\alpha = 0$ (i.e. one where only sequence information is used) is a variant of the *linear assignment problem*, which is that of finding a maximum-weight matching in a bipartite graph.

Hence the first stage of our algorithm finds a maximum-weight matching in the bipartite graph obtained by joining pairs of proteins in the two networks, where the weight of an edge is given by the sequence similarity of the two proteins that it joins. This matching can be obtained by the well-known *Hungarian algorithm* in polynomial time[24] and sped up with extensive use of priority queues and decomposition techniques.[23]

In the second stage of our algorithm, we apply the local search method, which is widely used in the combinatorial optimization field, to iteratively improve the initial matching while taking into account both the sequence score and the topology score of the matching.

From a variety of local search methods, we make use of the `2-Opt` algorithm, which was first proposed by Croes.[18] The `2-Opt` algorithm is one of the most famous heuristics for the well-known *Traveling Salesman Problem*.[25] Given a set of cities, the Traveling Salesman Problem is to find an ordering of cities that minimizes the total length of the tour when visiting all the cities in some order and returning to the starting city. The basic concept of the `2-Opt` algorithm is simple. A move deletes two edges of the original tour, thus breaking the tour into two paths, and then reconnects those paths by swapping these edges.

Local search algorithms do not appear to perform well from a theoretical point of view. Papadimitriou and Steiglitz[27] have shown that no local search algorithm (like `2-Opt`) that takes polynomial time per move can guarantee a constant approximation ratio for TSP unless P = NP. In addition, it has been shown that a sequence of exponential moves might be required by `2-Opt` before halting[26] and an analogous result[17] has been extended to `3-Opt` and k-`Opt`.

Although the worst-case analysis of the `2-Opt` algorithm is pessimistic, the average-case analysis is considerably more optimistic. A significant discovery[17] has shown that the expected approximation ratio to

the optimum is bounded by a constant. A similar improvement with respect to running time has been obtained as well. That is, the expected number of moves is polynomially bounded. Furthermore, `2-Opt` outperformed almost all the local search and greedy algorithms in experimental results for TSP.[20] More precisely, `2-Opt` (or k-`Opt`) gave better final tours than other local search algorithms for TSPLIB instances[20] with respect to both approximation ratio and running time.

The technique of iterative edge swaps is, however, not limited to TSP. It is also the basis of an algorithm for graph randomization, which attempts to produce a random graph with a given degree distribution.[21] It can be shown[22] that the corresponding Markov chain converges to the uniform distribution on the set of all connected simple graphs with the given degree distribution, thus giving an exact algorithm.

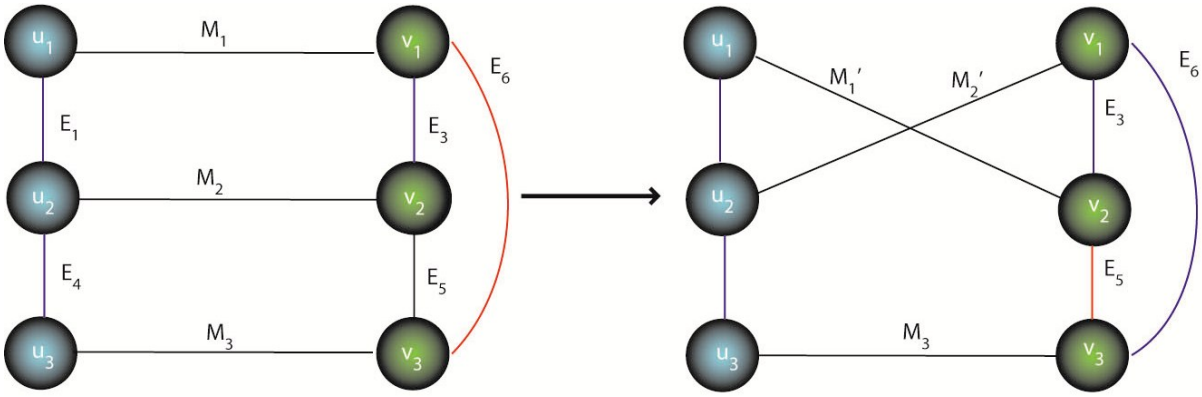


Fig. 1. The situation before and after a possible edge swap. The edges M_1 and M_2 of the matching are swapped out and replaced by $M_1^{'}$ and $M_2^{'}$. The matching is illustrated in black, conserved edges are in blue, and non-conserved edges are in red.

Figure 1 illustrates a matching on a pair of small networks transformed by an edge swap. It is important to note that an edge swap generally changes the conserved edges, provided these edges are incident to (i.e. share a vertex with) each of the two edges being swapped.

We now present the pseudocode of our algorithm. Its running-time is analyzed in the Appendix.

Algorithm 3.1. ***PISwap***

Given a weighted bipartite graph $G = (X \cup Y, E)$ with a maximum weighted matching M^* and parameters α and c,

(1) Compute topology similarities $t(e)$ and weight $w(e)$ for each edge $e \in M^*$, where the weight $w(e) = \alpha t(e) + (1-\alpha)s(e)$;

(2) Find a candidate set S consisting of every edge $e = (x, y) \in M^*$ which satisfies the following condition: There is $e' = (u, v) \in M^*$ with $v \in \text{prefer}_Y(x), u \in \text{prefer}_X(y)$ such that, for $e_1 = (x, v)$, $e_2 = (u, y)$,

$$swap(e, e') := \{w(e_1) + w(e_2) + \alpha(t(e_1) + t(e_2))\} - \{w(e) + w(e') + \alpha(t(e) + t(e'))\} > 0; \qquad (2)$$

(3) ***while*** *$S \neq \emptyset$*
do

3-1. Select the pair of edges $e = (x, y) \in S$ and $e' = (u, v)$ which achieve the maximum value of $swap(e, e')$;

3-2. Swap e, e' with $e_1 = (x, v)$, $e_2 = (u, y)$ to obtain a new matching $(M^* \setminus \{e, e'\}) \cup \{e_1, e_2\}$ and $S = S \setminus \{e, e'\}$;

3-3. Verify if the new inserted edges e_1, e_2 satisfy the condition (2) above and put them into S if necessary;

3-4. Update the topology similarities $t(e)$ for each edge $e \in M^*$ with one endpoint in $N(x) \cup N(u)$ and the other, in $N(y) \cup N(v)$ and modify $w(e)$ and $swap(e, e')$;

end while

(4) Output the final matching M^*.

4. Implementation details

The algorithm was implemented in Python 2.5.4[31] using the NetworkX[32] package, as well as Joris van Rantwijk's GPL implementation of the maximum-weight matching algorithm based on the blossom method for finding augmenting paths and the primal-dual method for finding a matching of maximum weight.[33] All experiments were performed on a Lenovo laptop with a 64-bit architecture running Windows Vista with an Intel Core 2 Duo T9600 CPU and 4 GB of RAM.

5. Experimental results

In spite of the analysis presented in the Appendix, the running-time of our algorithm is actually dominated by the preprocessing step, that of finding a maximum-weight bipartite matching. This is because the running time is cubic in the size of the input, whereas the number of iterations in the main loop of the algorithm was typically between 40 and 120. Thus, the upper bound presented in the analysis is overly pessimistic. We used the value $c = 200$ for our experiments, since this was close to the maximum degree Δ of the input networks (which was 190, 184 and 323 for C.elegans, D.melanogaster and S.cerevisiae, respectively).

In terms of actual time, the initial matching took between 5 and 10 minutes to compute on a single-processor laptop, but it was then saved and used with multiple values of α and c. Each of the runs of the second stage of the algorithm took no more than 15 to 20 seconds to produce the final mapping.

The ratio of α to $1 - \alpha$ can be thought of as giving the relative weight of sequence information to topology information. We found that choosing α to make this relative weight half of what it is for the initial mapping was a good rule of thumb. In other words, since the original mapping was based purely on sequence information, we chose α to favor the topology twice as much as the original mapping did, in all the reported experiments. We used both raw BLAST scores and normalized BLAST scores, retrieved from the IsoRank website.[34] The raw BLAST scores used were computed as $s(i, j) = B(i, j) + B(j, i)$, where $B(i, j)$ is the value given by BLAST on input i and j (this is because BLAST may occasionally produce "asymmetric" results). The normalized scores were computed as $\frac{s(i,j)}{\sqrt{s(i,i)s(j,j)}}$, and resulted in values between 0 and 1. Although our algorithm is described as dealing with integers, it is actually capable of handling fractional weights (although the running-time bound proved in the Appendix does not hold in that case). Also, it is important to note that since the initial matching was a maximum-weight matching, not all the proteins which had a non-zero similarity to some protein in the other species ended up in the matching — the size of the matching was roughly $0.9 \min(|V_1|, |V_2|)$, or 90% of the largest possible matching. We chose not to force matches between proteins with zero sequence similarity to all proteins in the other species, since topology alone does not suffice to create a reliable matching between them.[5]

Table 1 illustrates the results for the unnormalized input data (the results for the normalized data are qualitatively very similar). It clearly shows that PISwap compares favorably to other algorithms. The abbreviations denote the species used: CE = C.elegans, DM = D.melanogaster, SC = S.cerevisiae. The figures for the initial matching are provided for comparison purposes only, and it is the final alignments that are output by the algorithm.

The number of swaps required was 97 (102) for that between D.melanogaster and S.cerevisiae, 38 (54) for that between C.elegans and S.cerevisiae, and 71 (85) for the mapping between C.elegans and D.melanogaster. In each pair, the first number indicates the unnormalized case and the second, the normalized case. Note that these are significantly higher than the diameter (length of the path between the two most distant nodes) of the networks, which are 14, 11 and 9 for C.elegans, D.melanogaster and S.cerevisiae, respectively, meaning that information may have had a chance to propagate to all of the nodes.

To evaluate the output, we first used the HomoloGene database[35] which is thought to contain a reliable orthology mapping, though mainly based on sequence similarity between the proteins and the DNA regions that code for them.[35] We also looked at the number of conserved interactions. It is interesting to note that, compared to the initial matching, the final matching contains the same number of functional orthologs based on HomoloGene as the gold standard (except in the case of C.elegans and D.melanogaster, where the pair

Table 1. Evaluation of alignments based on unnnormalized sequence data

	DM-SC alignment		CE-SC alignment		CE-DM alignment	
	Initial	Final	Initial	Final	Initial	Final
Number of swaps	0	97	0	38	0	71
HomoloGene pairs	51	51	41	41	819	818
Conserved edges	272	398	106	150	80	154
Functional Coherence	N/A	0.510	N/A	0.172	N/A	0.355

(CO5C8.6, CG1826) becomes lost in the swapping process), while at the same time having significantly more conserved interactions (on average, 60% more). This shows that our algorithm is indeed performing its goal of achieving a high topology similarity while retaining an excellent sequence similarity. For comparison, let us mention that the D.melanogaster - S.cerevisiae alignment produced by the five algorithms (MP,[28] MRF,[29] IsoRank,[5] GA[9] and PATH[30]) studied by Zaslavskiy et al.[9] produce between 36 and 41 orthologous pairs from HomoloGene, compared to 51 for our algorithm (although the additional constraints imposed on the problem studied there make it difficult to perform a fair comparison).

In addition, we computed the Functional Coherence values, following the method outlined by Singh et al.[5] The method for computing Functional Coherence can be summarized as follows. First, the GO terms corresponding to each protein are collected. Then, each GO term is mapped to a subset of the so-called standardized GO terms, which in this case are its ancestors at a distance 5 from the root of the GO tree. Finally, the similarity between each pair of aligned proteins is computed as the median of the fractional overlaps of their corresponding sets of standardized GO terms. The Functional Coherence of an alignment is defined as the average pairwise Functional Coherence of the protein pairs that it matches. The values for the D.melanogaster-S.cerevisiae alignment produced by our algorithm was 0.510, comparable to the values of 0.519, 0.509 and 0.522 produced on the same input by IsoRank, GA and PATH, respectively.[9]

6. Evolutionary model

Although the edge-swapping technique was originally inspired by the field of combinatorial optimization, one can speculate that it can actually give us insights into the way two networks evolved from a common ancestor. If the networks of two closely related species are being analyzed, it is conceivable that at the outset, the proteins of the two networks are essentially identical in sequence, and hence, their correspondence can be determined exclusively on the basis of sequence information. Suppose, however, that as the two species evolve, a pair of proteins in one of the species have traded functions with one another. In that case, reconstructing the initial correspondence would require precisely an edge swap.

Comparing the network alignment problem to the (simpler) sequence alignment problem, one could say that edge swaps at the network level are the analog of compensatory mutations at the sequence level. One could then argue that, just as compensatory mutations can provide important clues for the evolutionary history of the sequences, function exchanges (represented by edge swaps) can provide important indications for the evolutionary history of the protein interaction networks. Unfortunately, function exchanges are much more difficult to detect than compensatory mutations, since network data is noisy, incomplete and unreliable.[5] Nevertheless, an algorithm such as PISwap could be adapted to estimating the number of function exchange events that have taken place during the evolutionary process.

While evolutionary events other than exchanges of function, such as duplications, insertions and deletions of proteins, certainly take place in biological networks,[14] this approach can still yield useful biological insights. In addition, the evolutionary distance between two species could in principle be computed from the number of evolutionary events (including function exchanges) that have taken place, and could perhaps provide a

more accurate estimate than the (appropriately defined and weighted) edit distance between two orthologous sequences present in those two species, since it would in some sense encompass all the protein sequences at once.

7. Discussion

We presented an algorithm for performing a global alignment of two protein interaction networks. In this algorithm, the parameter α plays an important role because it determines the relative importance of the topology data and the sequence data. Although our objective function is identical to that used in the IsoRank algorithm,[5] there are a number of important differences. IsoRank performs a random walk on the graph $G = G_X \otimes G_Y$, the tensor product of the two networks, where at each step, the walk is restarted with probability $1 - \alpha$ at a node $v = (x, y)$ in G chosen at random from the distribution proportional to the sequence similarity $s(x, y)$.[36] On the other hand, our algorithm can be thought of as performing a walk on the set of all matchings in the complete bipartite graph, and this walk is not random, but has the property that every step increases the value of the objective function. Another difference from IsoRank is that the output of IsoRank in terms of the pairwise alignment scores R_{ij} changes continuously with α, whereas in our algorithm, the set of possible matchings is discrete and it is clear that the interval $[0, 1]$ can be subdivided into non-overlapping subintervals such that on each one, the resulting matching is the same. Finally, our algorithm is not based on a spectral method, unlike both IsoRank and IsoRankN.

It is conceivable to obtain an "ensemble" or "fuzzy" mapping by using the mappings produced with several different values of α. The common part of these mappings would provide a more reliable set of functional orthologs, while the differences between the mappings could be used to evaluate the confidence of the assignments produced. In order to apply that strategy, it suffices to compute a single maximum-weight bipartite matching, which is the more time-intensive part of the algorithm, and save it in order to be able to reuse it with different values of the parameter α.

In order to speed up the algorithm without degrading its performance, it is possible to combine it with preprocessing by a graph-partitioning algorithm. For instance, if the two networks, G_X and G_Y, are each partitioned into two pieces (for instance, via a min-cut algorithm[37]), say, (G_X^1, G_X^2) and (G_Y^1, G_Y^2), one could apply the algorithm on each pair of pieces and then find the one that works best. Because there would be few edges between G_X^1 and G_X^2, as well as between G_Y^1 and G_Y^2, the difference in performance would not be significant. However, since the running time is $O(M^3)$, dominated in practice by the maximum-weight bipartite matching, if we assume that the number of vertices in both pieces is roughly equal, this would reduce the overall running time by a factor of 2. This strategy, of course, can also in principle be applied with other kinds of partitioning algorithms (such as community detection algorithms[38]).

Finally, as discussed in the previous section, this algorithm could yield new insights into the evolutionary history of the two species being analyzed.

Acknowledgments

The authors would like to thank Kelley Bailey, Mathieu Blanchette, Michael Schnall-Levin, Kanghao Lu, and Rohit Singh for assistance and useful discussion. A special thanks to Samujjal Purkayastha for creating Figure 1 and providing computational support. Leonid Chindelevitch was supported by the Postgraduate Award from the National Sciences and Engineering Research Council of Canada. Chung-Shou Liao was supported by the National Science Council under the Grants NSC95-2221-E-001-016-MY3.

References

1. T. Ito et al. A comprehensive two-hybrid analysis to explore the yeast protein interactome. In *Proceedings of the National Academy of Sciences USA*, **98(8)** (2001) pp. 4569-4574.
2. Y. Ho et al. Systematic identification of protein complexes in Saccharomyces cerevisae by mass spectrometry. *Nature* **415** (2002) pp. 180-183.

3. A.C. Gavin et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* **415 (6868)** (2002) pp. 141-147.
4. R. Sharan et al. Conserved patterns of protein interaction in multiple species. In *Proceedings of the National Academy of Sciences USA*, **102** (2005) pp. 1974-1979.
5. R. Singh et al. Global alignment of multiple protein interaction networks with application to functional orthology detection. In *Proceedings of the National Academy of Sciences USA*, **105** (2008) pp. 12763–12768.
6. M. Kalaev et al. Fast and accurate alignment of multiple protein networks. In Research in Computational Molecular Biology. Vol. 4955, Lecture Notes in Computer Science, Springer, Berlin/Heidelberg (2008) pp. 246-256.
7. J. Flannick. et al. Automatic parameter learning for multiple network alignment. In *Research in Computational Molecular Biology*. Vol. 4955, Lecture Notes in Computer Science, Springer, Berlin/Heidelberg (2008) pp. 214-231.
8. C.-S. Liao, K. Lu, M. Baym, R. Singh and B. Berger. IsoRankN: spectral methods for global alignment of multiple protein networks. *Proceedings of the International Conference on Intelligent Systems in Molecular Biology, ISMB '09* (2009), pp. 253-258. doi:10.1093/bioinformatics/btp203
9. M. Zaslavskiy, F. Bach, J.-P. Vert. Global alignment of protein-protein interaction networks by graph matching methods. *Proceedings of the International Conference on Intelligent Systems in Molecular Biology, ISMB '09* (2009), pp. 259-267. doi:10.1093/bioinformatics/btp196
10. J. Berg and M. Lässig. Cross-species analysis of biological networks by Bayesian alignment. In *Proceedings of the National Academy of Sciences USA*, **103** (2006) pp. 10967-10972.
11. J. Dutkowski and J. Tiuryn. Identification of functional modules from conserved ancestral protein-protein interactions. *Bioinformatics*, **23** (2007) pp. 149-158.
12. B.P. Kelley et al. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. In *Proceedings of the National Academy of Sciences USA*, **100** (2003) pp. 11394-11399.
13. B.P. Kelley et al. Pathblast: a tool for alignment of protein interaction networks. *Nucleic Acids Research*, **32** (2004) pp. 83-88.
14. M. Koyutürk et al. Pairwise Alignment of Protein Interaction Networks. *Journal of Computational Biology*, **13(2)** (2006) pp. 182–199.
15. B.S. Srinivasan et al. Integrated protein interaction networks for 11 microbes. In *Research in Computational Molecular Biology*. Vol. 3909, Lecture Notes in Computer Science, Springer, Berlin/Heidelberg (2006) pp. 1-14.
16. S. Sahni, T. Gonzales. P-complete approximation problems. *Journal of the Association for Computing Machinery*, **23** (1976) pp. 555-565.
17. B. Chandra, H. Karloff, and C. Tovey. New results on the old k-opt algorithm for the TSP. In *Proceedings of the 5th ACM-SIAM Symposium on Discrete Algorithm, SODA'94*, (1994) pp. 150–159.
18. G.A. Croes. A method for solving traveling salesman problems. *Operations Research*, **6** (1958) pp. 791–812.
19. T.J.P. Hubbard et al. Ensembl 2007. *Nucleoc Acids Research*, **35** (2007) pp. 610–617.
20. D.S. Johnson and L.A. McGeoch. The traveling salsman problem: a case study in local optimization. In *Local Search in Combinatorial Optimization*, John Wiley & Sons, London, (1997) pp. 215–310.
21. F. Viger, M. Latapy. Efficient and simple generation of random simple connected graphs with prescribed degree sequence. In *Proceedings of the International Computing and Combinatorics Conference, COCOON '05*, (2005) pp. 440–449.
22. R. Taylor. Constrained switchings in graphs. *Combinatorial Mathematics*, **8** (1980) pp. 314-336.
23. M.Y. Kao, T.W. Lam, W.K. Sung, and H.F. Ting. A decomposition theorem for maximum weight bipartite matchings. *SIAM Journal of Computing*, **31** (2001) pp. 18–26.
24. H.W. Kuhn. The Hungarian Method for the assignment problem. *Naval Research Logistics Quarterly*, **2** (1955) pp. 83–97.
25. E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys. The Traveling Salesman Problem, John Wiley & Sons, Chichester (1985).
26. G. Lueker, manuscript, Princeton University, (1976).
27. C.H. Papadimitriou and K. Steiglitz. On the complexity of local search for the traveling salesman problem. *SIAM Journal of Computing*, **6** (1977) pp. 76–83.
28. Jordan, M. (ed.) Learning in Graphical Models. The MIT Press, Cambridge, (2001).
29. Bandyopadhyay, S. et al. Systematic identification of functional orthologs based on protein network comparison. *Genome Research*, **16** (2006), pp. 428-435.
30. Zaslavskiy, M. et al. A path following algorithm for graph matching. In *Image and Signal Processing, Proceedings of the 3rd International Conference, ICISP*, (2008), pp. 329-337.
31. Python Software Foundation. *http://www.python.org/psf/*
32. A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference, SciPy '08*, (2008) pp. 11–15.

33. Z. Galil, Efficient Algorithms for Finding Maximum Matchings in Graphs. *ACM Computing Surveys*, **18:1** (1986), pp. 23–38.
34. IsoRank and IsoRankN. *http://isorank.csail.mit.edu* Last checked on September 21, 2009.
35. HomoloGene. *http://www.ncbi.nlm.nih.gov/homologene* Last checked on September 21 13, 2009.
36. L. Chindelevitch. Thoughts on the IsoRank algorithm. Unpublished manuscript.
37. A. A. Tsay, W. S. Lovejoy, D. R. Karger. Random Sampling in Cut, Flow, and Network Design Problems. *Mathematics of Operations Research*, **24:2** (1999) pp. 383-413.
38. M. E. Newman. Modularity and community structure in networks. In *Proceedings of the National Academy of Sciences USA*, **103:23** (2006) pp. 8577-8582.

8. Appendix

This appendix presents a running-time bound for our algorithm, with proof.

8.1. *Running-time analysis*

In what follows, M^* always denotes the mapping at the current iteration.

First, note that only topology similarities $t(e)$ for each edge $e \in M^*$ incident to a node in $N(x)$, $N(y)$, $N(u)$, and $N(v)$ are changed when we swap the edges (x, y) and (u, v) in M^*. In addition, as we consider the weight difference $w(M^* \setminus \{e, e'\} \cup \{e_1, e_2\}) - w(M^*)$, removing the edge $e = (x, y) \in M^*$ causes the weight loss $w(e)$, but it also causes neighbors in $N(x)$ and $N(y)$ to lose $\alpha t(e)$. Removing the edge $e' = (u, v)$ produces an analogous effect. On the other hand, there is a weight gain $w(e_1)$ as well as $\alpha t(e_1)$ from inserting the edge $e_1 = (x, v)$. Inserting the edge $e_1 = (x, v)$ produces an analogous effect.

Therefore, the weight of the matching M^* increases by the quantity $swap(e, e')$, defined in (2). The inequality in (2) thus ensures that the objective function increases after the swap.

Theorem 8.1.

Given a a weighted bipartite graph $G = (X \cup Y, E)$ with a maximum weighted matching M^* and parameters α and c, the running time of PISwap *is pseudo-polynomial time bounded in the worst case.*

Proof.

It is readily seen that the cardinality of a maximum-weight matching M^* is $|M^*| \leq \min\{|X|, |Y|\}$. Note that the preprocessing of PISwap to obtain a maximum weighted matching M^* by the *Hungarian algorithm* takes $O(|M^*|^3)$ time.

Let Δ denote the largest degree of a vertex in G_X and G_Y, i.e. the largest number of neighbors a vertex in $X \cup Y$ can have. Let B denote the largest similarity value for two sequences, i.e. $B = \max_{x \in X, y \in Y}\{s(x, y)\}$.

In Step 1 we compute the topology similarities $t(e)$ and the weights $w(e)$ for each edge $e = (x, y) \in M^*$. Since we consider all possible pairwise combinations between neighbors of x and neighbors of y, this requires $O(|M^*| \times \Delta^2)$ time.

In Step 2 we find the candidate set S. We first compute the subsets $\text{prefer}_Y(x)$ and $\text{prefer}_X(y)$ for each vertex in $X \cup Y$. The running time is bounded by $O(|X| \times |Y|)$ since it requires $O(|Y|)$ (respectively $O(|X|)$) time to find the c highest-weighted neighbors in Y (respectively X) for each vertex $x \in X$ (respectively $y \in Y$), for any constant c.

We then find all the edges $e' \in M^*$ satisfying the condition (2) for every edge $e \in M^*$. For every edge $e = (x, y) \in M^*$, there are at most c^2 edges in M^* with one endpoint in $\text{prefer}_X(y)$ and the other endpoint in $\text{prefer}_Y(x)$ that e can be swapped with. The weight difference $swap(e, e')$ can be computed in constant time from the topology similarities and sequence similarities for every edge. Hence Step 2 takes $O(c^2|M^*|)$ time. Without loss of generality, we assume $c \leq \sqrt{|M^*|}$; otherwise we can check all pairs of edges in M^* to see if they are able to be swapped, at a cost of $O(|M^*|^2)$.

Step 3 is an iteration, and we first consider the time complexity of one iteration. The maximum value of $swap(e, e')$ can be found in constant time by using a priority queue. The swap operation also takes constant time. For the two new inserted edges of M^*, we verify if they satisfy the property (2) in $O(c^2)$ time as above.

The last step to update the values of $t(e)$, $w(e)$, and $swap(e, e')$. This takes $O(\Delta^2)$ time as above, since only the edges $e \in M^*$ with one endpoint in $N(x) \cup N(u)$ and the other, in $N(y) \cup N(v)$, are affected. Therefore, each iteration requires $O(\max\{c^2, \Delta^2\})$ time.

Finally, consider the number of iterations of the while loop. The total sequence score is an integer and varies between 0 and $|M^*| \times B$, and similarly, the total topology score is an integer and varies between 0 and $|M^*|^2$; the consecutive values of $w(M^*)$ form a strictly increasing sequence whose length is bounded by $(|M^*| \times B + 1) \times (|M^*|^2 + 1) \in O(|M^*|^3 \times B)$. Since step 3 is the dominating one, PISwap runs in $O(\max\{c^2, \Delta^2\} \times B \times |M^*|^3)$ time. □

IDENTIFICATION OF COORDINATELY DYSREGULATED SUBNETWORKS IN COMPLEX PHENOTYPES

SALIM A. CHOWDHURY[1] AND MEHMET KOYUTÜRK[1,2]
[1] *Department of Electrical Engineering and Computer Science*
[2] *Center for Proteomics and Bioinformatics*
Case Western Reserve University, Cleveland, OH, USA
e-mail: {sxc426,koyuturk}@eecs.case.edu

In the study of complex phenotypes, single gene markers can only provide limited insights into the manifestation of phenotype. To this end, protein-protein interaction (PPI) networks prove useful in the identification of multiple interacting markers. Recent studies show that, when considered together, many proteins that are connected via physical and functional interactions exhibit significant differential expression with respect to various complex phenotypes, including cancers. As compared to single gene markers, these "coordinately dysregulated subnetworks" improve diagnosis and prognosis of cancer significantly and offer novel insights into the network dynamics of phenotype. However, the problem of identifying coordinately dysregulated subnetworks presents significant algorithmic challenges. Existing approaches utilize heuristics that aim to greedily maximize information-theoretic class separability measures, however, by definition of "coordinate" dysregulation, such greedy algorithms do not suit well to this problem. In this paper, we formulate coordinate dysregulation in the context of the well-known set-cover problem, with a view to capturing the coordination between multiple genes at a sample-specific resolution. Based on this formulation, we adapt state-of-the-art approximation algorithms for set-cover to the identification of coordinately dysregulated subnetworks. Comprehensive experimental results on human colorectal cancer (CRC) show that, when compared to existing algorithms, the proposed algorithm, NetCover, improves diagnosis of cancer and prediction of metastasis significantly. Our results also demonstrate that subnetworks in the neighborhood of known CRC driver genes exhibit significant coordinate dysregulation, indicating that the notion of coordinate dysregulation may indeed be useful in understanding the network dynamics of complex phenotypes.

1. Introduction

Variations among organisms occur in every aspect of biological systems, including their morphology, behavior, physiology, development and susceptibility to common diseases. Many of these phenotypes are controlled by multiple genetic and epigenetic factors and are therefore called complex phenotypes (multigenic traits), in contrast to phenotypes that are controlled by single genes (monogenic or Mendelian traits).[1]

In the past decade, genome-wide monitoring of gene expression, enabled by DNA microarray technology, has been commonly used as an important tool for the investigation of complex phenotypes, including human cancers. Differential analysis of gene expression facilitates identification of genes that are *dysregulated* with respect to the phenotype of interest; that is, genes that exhibit significant difference in the amount of mRNA transcripts present in a range of phenotype and control samples. To date, systematic analyses of differential gene expression has led to identification of genetic markers associated with many complex diseases, including leukemia,[2] breast cancer,[3] lung cancer[4] and prostate cancer,[5] as well as genes that are associated with tumor grade, metastasis, and disease recurrence .[6–9]

While investigation of differential gene expression for individual genes proves useful in identification of single-gene markers, it offers limited insights into the interplay between multiple interacting factors. Therefore, research on complex phenotypes rapidly shifts toward identification of multiple genes that are together differentially expressed. Such multiple markers are likely to shed light on the underlying molecular mechanisms of complex phenotypes. Knowledge of molecular interactions proves extremely useful in identification of multiple markers, in that it establishes the physical basis for understanding the dynamics of the interplay between multiple factors, through network models. Indeed, integration of genome-wide expression data with protein-protein interactions (PPIs) is shown to be useful in extracting subnetworks composed of genes with correlated expression profiles across diverse conditions.[10,11]

In the context of complex phenotypes, a range of algorithmic approaches are developed for the identification of phenotype-implicated subnetworks. Earlier studies quantify differential expression for each gene individually and search for subnetworks with significant aggregate differential expression.[12–15] While these

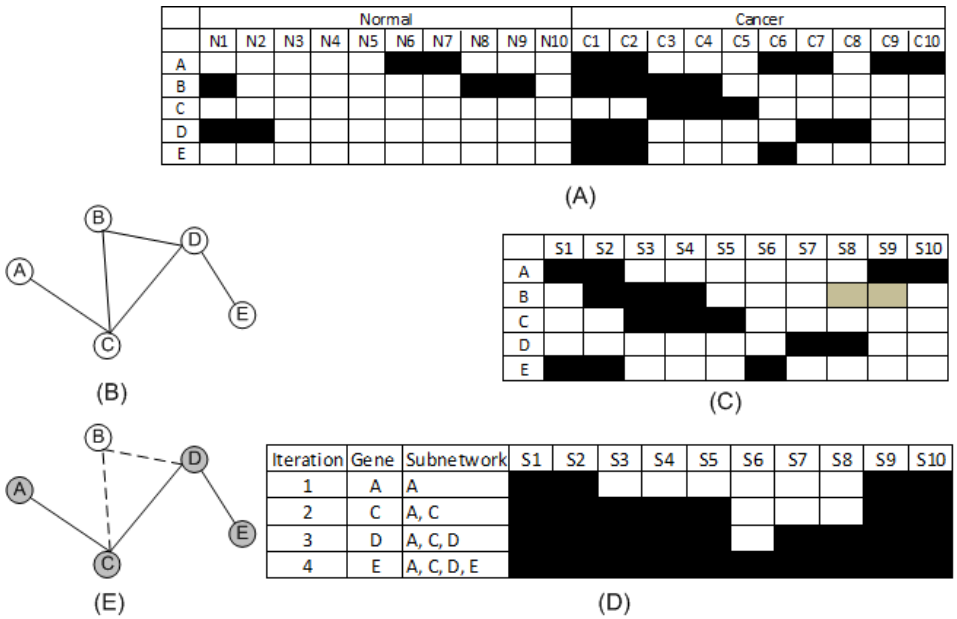


Fig. 1. Illustration of the proposed algorithm,NetCover, for identifying coordinately dysregulated subnetworks. (A) Binarized mRNA expression data with 10 paired samples and 5 genes. A light/dark square indicates that the respective gene is "expressed"/"not expressed" in the respective sample. (B) Part of human PPI network showing the interactions among the products of these genes. (C) *Differential expression profile* for each gene. A black/grey box indicates that the respective gene is up-regulated/down-regulated in the phenotype (with respect to control) for the respective paired sample (the gene *covers* the sample positively/negatively). (D) Progress of the algorithm showing the gene added to the subnetwork at each iteration and the set of samples covered by the corresponding subnetwork. (E) The resulting subnetwork, comprising of the highlighted genes and solid edges, covers all samples positively.

approaches are useful in relating individual genes that are differentially expressed, they do not necessarily capture the coordination or synergy in the dysregulation of multiple genes (e.g., genes that do not exhibit significant differential expression when considered individually, but exhibit significant differential expression when considered together). Chuang et al.[16] address this problem by considering *subnetwork activity*, defined as the aggregate expression of gene products in the subnetwork in each sample. Assessment of differential expression with respect to subnetwork activity enables identification of subnetworks that are *coordinately dysregulated*; i.e., groups of interacting proteins with collective mRNA-level differential expression. As compared to single gene markers, such subnetwork markers are shown to provide better classification performance in the prediction of disease progression in breast cancer.[16] Similarly, the concept of synergistic differential expression captures the collective differential expression of a group of genes that are not individually dysregulated.[17]

The concept of coordinate dysregulation is quite promising in generating novel insights into the network dynamics of complex phenotypes. However, the problem of identifying subnetworks with significant coordinate dysregulation is intractable. Furthermore, since the objective function associated with this problem is combinatorial in nature,[17] bottom-up heuristics that grow subnetworks to greedily maximize the objective function may seriously lack global awareness. Motivated by these considerations, we formulate this problem as a variation of the well-known set-cover problem. The proposed approach is illustrated in Figure 1. As seen in the figure, we first quantize a gene expression dataset with paired samples into binary expression levels. Then, for each gene, we identify individual samples that are *covered* (can be discriminated as phenotype or control) by the expression level of that gene. Subsequently, we search the human protein-protein interaction (PPI) network for subnetworks composed of genes that together cover all samples in the dataset. Since the genes in such a subnetwork complement each other in discriminating phenotype and control, we expect that these subnetworks may have a modular role in the manifestation of phenotype.

In the next section, we formally introduce the proposed framework. We argue that this formulation better suits to the notion of coordinate dysregulation from a biological perspective, in that it captures the coordination between multiple genes at a sample-specific resolution. Furthermore, we theoretically establish the relationship between set-cover and information-theoretic formulation of coordinate dysregulation. We then adapt state-of-

the-art approximation algorithms for set-cover to the problem of identifying minimal subnetworks that cover all samples in a dataset. In Section 3, we evaluate the biological relevance of identified subnetworks in the context of diagnosis and prognosis of human colorectal cancer (CRC). Comprehensive experimental results show that, subnetworks identified by the proposed algorithm, NetCover, outperform subnetworks identified by existing algorithms in terms of accurate classification of tumorigenic and metastatic samples.

2. Methods

In this section, we first introduce the notion of coordinate dysregulation and provide the motivation for the proposed algorithmic approach. Then, based on an information-theoretic formulation of coordinate dysregulation, we demonstrate the relationship between the problem of identifying coordinately dysregulated subnetworks and the well-known set-cover problem. Finally, we propose a cover-based algorithm for the identification of coordinately dysregulated subnetworks and discuss how the subnetworks identified by our algorithm can be used for the diagnosis and prognosis of complex diseases.

2.1. *Coordinately Dysregulated Subnetworks*

In the context of a specific phenotype, a group of genes that exhibit significant differential expression and whose products are connected to each other through physical and functional interactions may be useful in understanding the network dynamics of the phenotype. This is because, the patterns of (i) collective differential expression and (ii) connectivity in PPI network are derived from orthogonal sources (sample-specific mRNA expression and generic protein-protein interactions, respectively). Thus, they provide corroborating evidence indicating that the corresponding subnetwork of the PPI network may play an important role in the manifestation of phenotype. In this paper, we refer to the collective differential expression of a group of genes as *coordinate dysregulation*. A group of coordinately dysregulated genes that induce a connected subnetwork in a PPI network is thus called a *coordinately dysregulated subnetwork*.

Dysregulation of a gene with respect to a phenotype. For a set $\mathcal{V}$ of genes and $\mathcal{U}$ of samples, let $E_i \in R^{|\mathcal{U}|}$ denote the properly normalized[18] gene expression vector for gene $g_i \in \mathcal{V}$, where $E_i(j)$ denotes the relative expression of g_i in sample $s_j \in \mathcal{U}$. Assume that the phenotype vector C annotates each sample as phenotype or control, such that $C_j = 1$ indicates that sample s_j is associated with the phenotype (e.g., taken from a tumor tissue) and $C_j = 0$ indicates that s_j is a control sample (e.g., taken from a normal tissue). Then, the mutual information $I(E_i; C) = H(C) - H(C|E_i)$ of E_i and C is a measure of the reduction of uncertainty about phenotype C due to the knowledge of the expression level of gene g_i. Here, $H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$ denotes the Shannon entropy of discrete random variable X with support $\mathcal{X}$. The entropy $H(E_i)$ of the expression profile of gene g_i is computed by quantizing E_i properly. Clearly, $I(E_i; C)$ provides a reasonable measure of the dysregulation of g_i, since it quantifies the power of the expression level of g_i in distinguishing phenotype and control samples.

Coordinate dysregulation. Now let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a PPI network where the product of each gene $g_i \in \mathcal{V}$ is represented by a node and each edge $g_i g_j \in \mathcal{E}$ represents an interaction between the products of g_i and g_j. For a given subnetwork of $\mathcal{G}$ with set of nodes $S \subseteq \mathcal{V}$, Chuang *et al.*[16] define the *subnetwork activity* of S as $E_S = \frac{1}{\sqrt{|S|}} \sum_{g_i \in S} E_i$, that is the aggregate expression profile of the genes in S. Then, naturally, the dysregulation of subnetwork S is given by $I(E_S; C)$, which provides a measure of the reduction of uncertainty about phenotype C due to the knowledge of the aggregate expression level of the genes in S. In the following discussion, we refer to $I(E_S; C)$ as the *coordinate dysregulation* of S.

Identification of coordinately dysregulated subnetworks. Clearly, identification of subnetworks with maximal $I(E_S; C)$ is an intractable computational problem. Simple greedy approaches to this problem grow subnetworks by starting from a single protein and adding to the subnetwork the proteins in its network neighborhood. At each step, the protein in the neighborhood that maximally increases $I(E_S; C)$ is added to the subnetwork. While such algorithms are useful in identifying subnetworks with reasonably high coordinate dysregulation, they are biased toward identifying subnetworks with very few genes that exhibit significant

individual dysregulation. Consider, for example, a group of genes that are marginally dysregulated with respect to the phenotype, when considered individually. Assume that these genes exhibit significant dysregulation when considered together (i.e., have a large $I(E_S; C)$). This group of genes is not likely to be identified by such an algorithm since the individual contribution of each gene to the dysregulation of the subnetwork will not be apparent at any stage of the algorithm until all genes are added to the subnetwork. Indeed, in our experiments on a gene expression dataset (*GSE8671*) with 8987 genes and 32 samples of colorectal adenomas paired with those of normal mucosa, such an algorithm assigns 84% of the genes to subnetworks composed of at most two genes. However, for effective investigation of the systems biology of complex phenotypes, larger subnetworks with weaker individual, but stronger coordinate dysregulation are very interesting since they offer insights beyond what single gene markers can provide. In the following discussion, we propose a novel framework that utilizes biological insights into the dysregulation of genes at a sample-specific resolution, to develop algorithms for more effective discovery of such coordinately dysregulated subnetworks.

2.2. *Coordinate Dysregulation and Set Cover*

We now show that the problem of identifying coordinately dysregulated subnetworks can be formulated as a variation of the set-cover problem. For this purpose, consider a binary representation of the gene expression data of interest. Binary representation of gene expression is commonly utilized for several reasons, including removal of noise, algorithmic considerations, and tractable biological interpretation of identified patterns. Such approaches are shown to be effective in the context various problems, ranging from genetic network inference[19] to clustering[20] and classification.[21] There are also many algorithms for effective binarization of gene expression data.[22] For our purposes, let $\hat{E}_i$ denote the binarized expression profile of gene g_i (we discuss how we binarize a gene expression dataset in our experiments in Section 3). We say that gene g_i is *expressed* (or "on") in sample s_j if $\hat{E}_i(j) = 1$ and *not expressed* (or "off") if $\hat{E}_i(j) = 0$.

In order to illustrate the relationship between coordinate dysregulation and set-cover, we introduce various concepts that provide insights into the dysregulation of genes at a sample-specific resolution. For this purpose, we assume that the gene expression data is paired; that is, there is one-to-one correspondence between phenotype and control samples. This is indeed the case for many available gene expression datasets that monitor complex phenotypes, e.g., for each sample taken from a cancerous tissue of a patient, a control sample is also taken from the part of the tissue without the lesion. This approach controls for the noise and bias that might be introduced by the biological variability among different tissues or individuals. Formally, we assume that $|\mathcal{U}| = 2n$ is even, and for each $1 \leq j \leq n$, the pair of samples s_j and s_{j+n} are phenotype and control samples that are associated with each other (i.e., they come from the same individual or tissue). A sample instance of binary gene expression data with paired samples is shown in Figure 1(A). We can now define the positive and negative cover for a gene.

Definition 1. Positive and Negative Cover Set of a Gene. A gene g_i is said to cover a sample s_j *positively/negatively* if it is up-regulated/down-regulated in the phenotype sample with respect to control ($\hat{E}_i(j) = 1$ and $\hat{E}_i(j+n) = 0$/ $\hat{E}_i(j) = 0$ and $\hat{E}_i(j+n) = 1$). The set of samples that are covered positively/negatively by g_i is called the *positive/negative cover set* of g_i and denoted $\mathcal{P}_i = \mathcal{P}(g_i)/\mathcal{N}_i = \mathcal{N}(g_i)$.

Note here that the notion of up- or down-regulation of a gene with respect to a sample depends on the procedure for binarization of expression levels. For this reason, we systematically evaluate the effect of binarization on the performance of our algorithms in Section 3.

The positive and negative cover sets of the genes in Figure 1(A) are shown in Figure 1(C). As seen in the figure, gene B covers S2 positively since it is expressed in C2 while it is not expressed in N2. Overall, the positive cover set of gene B is given by $\mathcal{P}(B) = \{S2, S3, S4\}$ and its negative cover set is given by $\mathcal{N}(B) = \{S8, S9\}$. Observe that, since gene B is dysregulated with respect to samples S2, S3, and S4, it can be used to distinguish phenotype and control samples based on its expression in a given sample. However, clearly, the statistical power (or reliability) of gene B in distinguishing phenotype and control samples depends on the number of samples that it covers. Moreover, the samples S8 and S9, which are covered negatively by gene B

interfere with its power in distinguishing phenotype and control samples, since the signals provided by these two groups of samples are conflicting with each other.

Based on these observations, we postulate that genes that can distinguish different sets of samples may be complementary of each other in the manifestation of phenotype. In other words, the dysregulation of genes involved in similar processes may have similar effects on phenotype, and since such genes are expected to be functionally related, they are likely to be in close proximity of each other in a network of interactions. Consequently, if we can identify subnetworks composed of genes that together cover all samples consistently (i.e., either positively or negatively), then the products of these genes may indeed have a coordinate effect on the manifestation of the phenotype. These subnetworks may be useful as features for classification of phenotype and they may reveal targets for theurapetic intervention. Motivated by these considerations, we formulate the problem of identifying coordinately dysregulated subnetworks as one of identifying minimal groups of interacting genes that cover all samples either positively or negatively. To provide a theoretical foundation for this approach, we first show that the cardinality of the cover set of a gene is related to a sound measure of class separability, namely the mutual information between the expression profile of the gene and the phenotype vector.

Theorem 1. *For any two genes $g_i, g_j \in V$, if $|\,|\mathcal{P}_i| - |\mathcal{N}_i|\,| > |\,|\mathcal{P}_j| - |\mathcal{N}_j|\,|$, then $I(\hat{E}_i; C) > I(\hat{E}_j; C)$.*

Proof. By definition of mutual information, we have $I(\hat{E}_i; C) - I(\hat{E}_j; C) = H(C|\hat{E}_j) - H(C|\hat{E}_i)$. Therefore, it will suffice to show that $|\,|\mathcal{P}_i| - |\mathcal{N}_i|\,| > |\,|\mathcal{P}_j| - |\mathcal{N}_j|\,|$ implies $H(C|\hat{E}_i) < H(C|\hat{E}_j)$. First note that

$$H(C|\hat{E}_i) = P(\hat{E}_i = 0)H(C|\hat{E}_i = 0) + P(\hat{E}_i = 1)H(C|\hat{E}_i = 1). \tag{1}$$

We will show that $H(C|\hat{E}_i = 0)$ and $H(C|\hat{E}_i = 1)$ both decline with growing $|\,|\mathcal{P}_i| - |\mathcal{N}_i|\,|$, to conclude that $H(C|\hat{E}_i)$ also declines with growing $|\,|\mathcal{P}_i| - |\mathcal{N}_i|\,|$, since $P(\hat{E}_i = 0)$ and $P(\hat{E}_i = 1)$ are both positive.

Now, for $x, y \in \{0, 1\}$, let $n_i^{(x,y)}$ denote the number of samples with phenotype x in which the binary expression of g_i is y; e.g., $n_i^{(1,0)}$ is the number of phenotype samples in which gene g_i is not expressed. Then, clearly

$$H(C|\hat{E}_i = 0) = h(p_i^{(0)}) \text{ and } H(C|\hat{E}_i = 1) = h(p_i^{(1)}). \tag{2}$$

Here, $h(p) = -p \log p - (1 - p) \log(1 - p)$ denotes the entropy of a Bernoulli random variable with success probability p, $p_i^{(0)} = \frac{n_i^{(0,0)}}{n_i^{(0,0)} + n_i^{(1,0)}}$, and $p_i^{(1)} = \frac{n_i^{(0,1)}}{n_i^{(0,1)} + n_i^{(1,1)}}$. Let $n_i^+ = |\mathcal{P}_i|$, $n_i^- = |\mathcal{N}_i|$ and let m_i^+ / m_i^- denote the number of samples in which g_i is expressed/not expressed in both phenotype and control. Then, we can write:

$$n_i^{(0,0)} = n_i^+ + m_i^-,\ n_i^{(0,1)} = n_i^- + m_i^+,\ n_i^{(1,0)} = n_i^- + m_i^-,\ \text{and } n_i^{(1,1)} = n_i^+ + m_i^+. \tag{3}$$

Consequently,

$$p_i^{(0)} = \frac{n_i^+ + m_i^-}{n_i^+ + n_i^- + 2m_i^-}, \tag{4}$$

and therefore we have $p_i^{(0)} - 1/2 = \frac{n_i^+ - n_i^-}{2(n_i^+ + n_i^- + 2m_i^-)}$. Since $p_i^{(0)} - 1/2$ assumes its zero at $n_i^+ = n_i^-$ and its derivative with respect to n_i^+ is always positive, $|p_i^{(0)} - 1/2|$ grows with growing $|n_i^+ - n_i^-|$. Since the entropy function $h(p_i^{(0)})$ is maximized at $p_i^{(0)} = 1/2$ and declines with growing $|p_i^{(0)} - 1/2|$, we conclude that $h(p_i^{(0)})$ (similary, $h(p_i^{(1)})$) declines with growing $|\,|\mathcal{P}_i| - |\mathcal{N}_i|\,|$, completing the proof. □

This theorem establishes that the number of paired samples for which a gene is consistently up- or down-regulated is directly associated with the information its expression profile provides on the phenotype. It can be seen from the construction of the proof that this result can also be generalized to the aggregate expression profile E_S of a subnetwork S (where E_S is quantized properly). Motivated by this observation, we generalize the notion of positive and negative cover sets to subnetworks and conjecture that a subnetwork with a larger consistently positive or negative cover set provides more information on the phenotype.

Definition 2. Positive and Negative Cover Set of a Subnetwork. For a given subnetwork $S \subseteq \mathcal{V}$, the *positive* and *negative* cover sets of S are respectively defined as $\mathcal{P}(S) = \bigcup_{g_i \in S} \mathcal{P}_i$ and $\mathcal{N}(S) = \bigcup_{g_i \in S} \mathcal{N}_i$.

2.3. *Minimal Covering Subnetwork Problem*

We now formulate the coordinately dysregulated subnetwork identification problem as one of identifying minimal subnetworks that cover all samples either positively or negatively. Similar set-cover based approaches are also shown to be effective in feature selection.[23] Here, rather than searching for the "best" subnetwork in the entire network, we look for the "best" subnetwork associated with a given gene. This is because, from a biological perspective, it is not necessarily true that a single subnetwork that maximizes an objective criterion over the entire network will be the only relevant subnetwork. On the contrary, for various applications including identification of subnetwork markers for classification, multiple subnetworks are useful. Furthermore, in understanding the relationship between the manifestation of phenotype at different levels of cellular control (e.g., genomic sequences, gene expression, protein expression), researchers may be interested in finding subnetworks associated with known genetic markers. Indeed, in Section 3.5, we demonstrate that subnetworks associated with known genetic markers ("driver genes") of colorectal cancer (CRC) are more effective in classification of CRC as compared to subnetworks associated with random genes. Furthermore, proteomic targets that are identified based on differential protein expression, when used as seeds for identification of dysregulated subnetwork markers, are shown to provide significant insights into the systems biology of complex phenotypes.[24]

Definition 3. Minimal Covering Subnetwork Associated with a Gene. For a set of paired samples $\mathcal{U}$, a set of genes $\mathcal{V}$ with binary expression profiles $\hat{E}_i \in \{0,1\}^{|\mathcal{U}|}$, a PPI network $G = (\mathcal{V}, \mathcal{E})$ and a gene $g_i \in \mathcal{V}$, the minimal covering subnetwork associated with g_i is defined as a subnetwork $S_i \subseteq \mathcal{V}$ satisfying the following conditions:

(1) $g_i \in S_i$.
(2) S_i is a local subnetwork, i.e., $\forall\ g_j \in S_i,\ \exists\ g_k \in S_i$ such that $\delta(g_j, g_k) \leq \ell$, where $\delta(g_j, g_k)$ denotes the network distance between g_j and g_k, and ℓ denotes an adjustable threshold that specifies the desired locality of the subnetwork (if $\ell = 1$, the subnetwork is connected).
(3) S_i covers all samples either positively or negatively, i.e., $\mathcal{P}(S_i) = \mathcal{U}$ or $\mathcal{N}(S_i) = \mathcal{U}$.
(4) If $\mathcal{P}(S_i) = \mathcal{U}$ ($\mathcal{N}(S_i) = \mathcal{U}$), then $|\mathcal{N}(S_i)|$ ($|\mathcal{P}(S_i)|$) is minimum over all subnetworks that satisfy the above three conditions.
(5) S_i is minimal, i.e., $\forall g_j \in S_i$, subnetwork $S_i \setminus \{g_j\}$ does not satisfy the above conditions.

Condition (1) ensures that the subnetwork is indeed associated with the gene of interest. Condition (2) ensures that the genes in the subnetwork are functionally associated with each other. Condition (3) ensures that for each paired sample, there is at least one gene in the subnetwork that can distinguish phenotype and control. Condition (4) ensures that the noise introduced by the genes that are dysregulated in the opposite direction is minimal. Finally, condition (5) ensures that there are no redundant genes in S_i.

Note that the minimal covering subnetwork problem is similar to the *minimum connected cover* (MCC) problem introducted by Ulitsky *et al.*[25] However, there is a fundamental conceptual difference between the two problems. MCC searches for subnetworks in which multiple genes are dysregulated in each phenotype sample. On the contrary, minimal covering subnetwork problem explicitly looks for subnetworks composed of genes that are *complementary* of each other in distinguishing phenotype and control samples.

2.4. *Algorithm for the Identification of Minimal Covering Subnetworks*

There is a clear conceptual and mathematical similarity between the minimal covering subnetwork problem and the well-known set-cover problem. An instance of the set-cover problem consists of a finite set X and a family $\mathcal{F}$ of subsets of X, such that the union of all the sets in $\mathcal{F}$ constitute X. The set-cover problem asks for a minimum size subset $\mathcal{C} \subseteq \mathcal{F}$ such that the union of all sets in $\mathcal{C}$ is equal to X. Clearly, the minimal covering subnetwork problem is a special case of this NP-hard problem.[26] Here, the selection of sets in $\mathcal{C}$

(which corresponds to S_i) are further constrained by (i) the network locality of the genes in S_i and (ii) a second collection of associated sets, union of which is to be minimized. We here adapt a polynomial-time approximation algorithm for this well-studied problem, which works by picking, at any stage, the set that covers the maximum number of remaining uncovered elements.[27]

The proposed algorithm for the identification of minimal covering subnetworks, NetCover, is illustrated in Figure 1. For a given gene g_i, assume without loss of generality that $|\mathcal{P}_i| > |\mathcal{N}_i|$. Then, NetCover identifies the minimal covering subnetwork associated with g_i as follows:

(1) Initialize subnetwork: $S_i \leftarrow \{g_i\}$
(2) Initialize set of uncovered samples: $\mathcal{T} \leftarrow \mathcal{U} \setminus \mathcal{P}_i$
(3) Initialize set of neighboring genes: $\mathcal{Q} \leftarrow \{g_j \in \mathcal{V} : \delta(g_i, g_j) \leq \ell\}$
(4) For all genes $g_j \in \mathcal{Q}$, compute $\mathcal{P}'_j \leftarrow \mathcal{P}_j \cap \mathcal{T}$
(5) Find the genes in $\mathcal{Q}$ with maximum $|\mathcal{P}'_j|$ and let g_k be the gene among these genes with minimum $|\mathcal{N}_j|$
(6) Update subnetwork: $S_i \leftarrow S_i \cup \{g_k\}$
(7) Update set of uncovered samples: $\mathcal{T} \leftarrow \mathcal{T} \setminus \mathcal{P}'_k$
(8) Update set of neighboring genes: $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{g_j \in \mathcal{V} : \delta(g_k, g_j) \leq \ell\} \setminus \{g_k\}$
(9) If $\mathcal{T} = \emptyset$ or $\mathcal{Q} = \emptyset$, return S_i; otherwise, go to step (4)

Since the cardinality of $\mathcal{Q}$ is bounded by $|\mathcal{V}|$ and the loop (4)-(9) can be repeated at most $|\mathcal{V}|$ times, the worst-case running time of this algorithm is $O(|\mathcal{V}|^2)$. However, in our experiments on several colorectal cancer datasets, we observe that the algorithm successfully identifies covering subnetworks that are fairly small (i.e., the loop exists with $\mathcal{T} = \emptyset$ after a few iterations).

2.5. *Using Minimal Covering Subnetworks for Classification*

To quantify the discriminative potential (hence, relevance to the manifestation of phenotype) of discovered subnetworks, we use these subnetworks to build classifiers for diagnosis and prognosis of the phenotype. For this purpose, for a given gene expression dataset, we first discover the minimal covering subnetwork for all genes in the human PPI network. Then, since the minimal covering subnetworks for more than one gene might be identical, we eliminate the subnetworks that are redundant. Subsequently, we score each of the remaining subnetworks according to their coordinate dysregulation ($I(E_S, C)$). Then we select the K non-redundant subnetworks with maximum $I(E_S, C)$. Here, we consider a subnetwork redundant if it shares a gene with a subnetwork that is already selected. The number of selected subnetwork features, K, is designated as an adjustable parameter. Finally, we use the aggregate expression profiles (E_S) of these selected subnetworks to compute feature vectors for each sample. We then use these feature vectors to train and test classifiers for the prognosis and diagnosis of the phenotype of interest, as we discuss in the next section.

3. Results and Discussion

In this section, we comprehensively evaluate the performance of the proposed algorithm in the context of human colorectal cancer (CRC) and compare its performance with the greedy algorithm by Chuang *et al.*,[16] which aims to directly maximize the additive mutual information by greedily growing subnetworks. We then investigate the biological relevance of coordinately dysregulated subnetworks in the network neighborhood of genes that are implicated in CRC according to gene association studies.

3.1. *Human Colorectal Cancer (CRC)*

Colorectal Cancer (CRC) is the third most common cancer and second leading cause responsible for cancer related death in the western world.[28] CRC generally starts with a simple, benign tumor. Most often, these growths go undetected, even for years, before they develop into malignancies. Therefore, until the symptoms of cancer are developed, diagnosis of cancer is rather difficult. Monitoring of differential gene expression is therefore useful for diagnosis of cancer at early stages, as well as prognosis of the development of the tumor and therapeutic outcome. Furthermore, since most deaths are related to diagnosis of CRC in late stages, understanding of the network dynamics of CRC thorough its various stages may be very useful in development

of more effective therapeutic intervention strategies. To this end, if coordinately dysregulated subnetworks that are identified with respect to dysregulation at a particular stage can successfully classify disease at another stage, such subnetworks may be identified as those that are important in the development of disease. For this reason, in our experimental studies, we use a particular dataset to identify coordinately dysregulated subnetworks and use these subnetworks to develop classifiers for the diagnosis and prognosis of CRC with respect to other datasets.

3.2. *Datasets*

In our experiments, we use three CRC-related microarray datasets obtained from GEO (Gene Expression Omnibus, `http://www.ncbi.nlm.nih.gov/geo/index.cgi`). These datasets, which are referenced here by their accession number in the GEO database, are the following:

- *GSE8671* contains the expression profiles of 8,987 genes across 32 prospectively collected adenomas paired with those of normal mucosa.[29]
- *GSE10950* contains the expression profiles of 18,171 genes across 24 normal and tumor pairs.[30]
- *GSE6988* contains the expression profiles of 17,104 genes across the following tissue samples: paired tissues of 25 normal colorectal mucosa, 27 primary colorectal tumors, 13 normal liver and 27 liver metastasis, and 20 primary colorectal tumors without liver metastasis.[31]

The human protein-protein interaction data used in our experiments is obtained from the Human Protein Reference Database (HPRD `http://www.hprd.org`). This dataset contains 35023 binary interactions among 9299 proteins, as well as 1060 protein complexes consisting of 2146 proteins. The binary interactions contain *in vivo*, as well as *in vitro* interactions obtained via high-throughput screening. We integrate the binary interactions and protein complexes using a matrix model (i.e., each complex is represented as a clique of the proteins in the complex), to obtain a PPI network composed of 42781 binary interactions among 9442 proteins.

In order to reduce the effect of systematic experimental bias in high-throughput microarray experiments, the expression profile of each gene in each dataset is normalized to have a mean value of zero and standard deviation of one across all the samples. For the identification of subnetworks, these normalized expression values are then binarized by setting the largest $\alpha\%$ of the normalized expression values to 1 and all of the rest to 0. Here, α is an adjustable parameter that specifies the fraction of "expressed" genes in the dataset. This tunable binarization scheme is chosen with a view to investigating the effect of binarization on the performance of NetCover (related experimental results are provided at the end of this section). In the experimental results reported in the following discussion, α is set to 25% since this value is found to be optimal for the performance of NetCover in all experiments. Note here that binary expression levels are used only for the identification of dysregulated subnetworks via NetCover. On the other hand, coordinate dsyregulation of subnetworks is computed by quantizing the aggregate expression profiles into eight bins, since this value is found to be optimal for the performance of Chuang *et al.*'s greedy algorithm. Therefore, the performance methods other than NetCover does not depend on α.

3.3. *Experimental Design*

NetCover is implemented in Matlab. Using this implementation, two different sets of coordinately dysregulated subnetworks are generated based on the datasets *GSE8671* and *GSE10950*. In order to evaluate the classification performance of these subnetworks, extensive experiments are performed by using various types of classifiers and different classification problems. Three classification problems are considered for this purpose:

- *Diagnosis of samples in GSE10950.* Subnetworks discovered using *GSE8671* are used to predict the class (cancerous *vs.* non-cancerous) of samples in *GSE10950*.
- *Diagnosis of samples in GSE6988.* Sets of subnetworks discovered using *GSE8671* and *GSE10950* are used to predict the class (cancerous *vs.* non-cancerous) of samples in *GSE6988*.
- *Prognosis of samples in GSE6988.* Subnetworks that are discovered using *GSE8671* and *GSE10950* are used to classify the 27 colorectal tumors with liver metastasis and 20 without liver metastasis in *GSE6988* into metastatic *vs.* non-metastatic classes.

For each of these problems, two types of classification tests are performed:

- *Cross-classification* (CC): The classifier is trained on one dataset and tested on another dataset (note that this is not applicable to the prognosis of samples in *GSE6988*, since other datasets do not contain metastasis information).
- *Leave-one-out cross-validation* (LOOCV): For each sample in a dataset, one sample is left out and the classifier is trained using the remaining samples in that dataset, which is then used to classify the corresponding sample. The performance of the classifier is evaluated by repeating this procedure for all samples.

For each instance, two types of classifiers are used: (i) a quadratic regression model, provided by Matlab's `classify` function and (ii) a support vector machine (SVM), provided by Matlab's `svmclassify` function.

3.4. *Classification Performance in Diagnosis and Prognosis of Colorectal Cancer*

In this section, we report systematic experimental results on the diagnostic and prognostic performance of the subnetwork markers identified on *GSE10950* and *GSE8671* datasets on classifying samples in the *GSE10950* and *GSE6988* datasets. The performance of each of these sets of subnetwork markers is compared against (i) the subnetwork markers identified by Chuang *et al.*'s greedy approach[16] (also implemented in Matlab for comparison purposes), and (ii) single gene markers. NetCover takes about 10 minutes for the identification of subnetworks associated with all genes in each of *GSE10950* and *GSE8671*, while the greedy algorithm takes about 30 minutes to complete the same task. The subnetwork markers that are identified by Chuang *et al.*'s algorithm are ranked and used for classification in the same way as the subnetwork markers identified by NetCover. Similar to subnetwork markers, single gene markers are ranked based on their individual mutual information with the phenotype and the top k markers are used for classification for $1 \leq k \leq K$.

The number of (subnetwork or single gene) markers, denoted K, is used as a free parameter for building, training, and testing the performance of classifiers. For each $1 \leq K \leq 50$, the performance of the classification is measured using 'Area Under ROC Curve'(AUC) criterion. AUC is a measure of the overall performance of a classifier, which accounts for the trade-off between the precision (selectivity) and recall (sensitivity) achieved by the classifier. Here, precision is defined as the fraction of true positives among all samples classified as phenotype by the classifier, while recall is defined as the fraction of true positives among all true phenotype samples. AUC is a measure of the average precision across varying values of recall (or vice versa) and an AUC of 1.0 indicates that the classifier provides perfect precision without sacrificing recall. Therefore, a value of AUC closer to 1.0 indicates better performance of a classifier. Note that, when prediction of the classifier does not depend on an adjustable threshold (i.e., there is only a single point on the ROC curve), the AUC value returned by Matlab is equal to the arithmetic mean of precision and recall. In those cases, other measures such as the harmonic mean of precision and recall (known as F-measure) are considered more reliable; however, in our experiments, AUC and F-measure provided similar results while comparing different classifiers.

The overall performance of the subnetwork and single gene markers is evaluated in Figure 2. In this figure, to systematically evaluate the performance of the subnetwork and single gene markers across different values of K, we report (i) average AUC across all values of K, ranging from 1 to 50 (average performance of the classifier) and (ii) maximum AUC across this range of K (optimal performance of the classifier that can be obtained by adjusting the number of markers accurately). As seen in the figure, in all three experiments (diagnosis of *GSE10950*, diagnosis and prognosis of *GSE6988*), subnetworks identified by NetCover demonstrate better classification performance compared to subnetworks discovered by greedy algorithm and single gene markers for both classification procedures and classifier types used. This observation indicates that NetCover discovers subnetworks that are more relevant in terms of the network dynamics of the progression of CRC, in that subnetworks identified in one dataset can distinguish samples in another data set better, as compared to single gene markers or subnetworks identified by another state-of-the-art algorithm.

3.5. *Coordinately Dysregulated Subnetworks Associated with Hallmarks of CRC*

In recent years, comparative genomic studies of CRC have revealed many genes with mutations that may be associated with colorectal cancer. These "Hallmarks of CRC" include *APC, CTNNB1, KRAS, HRAS,*

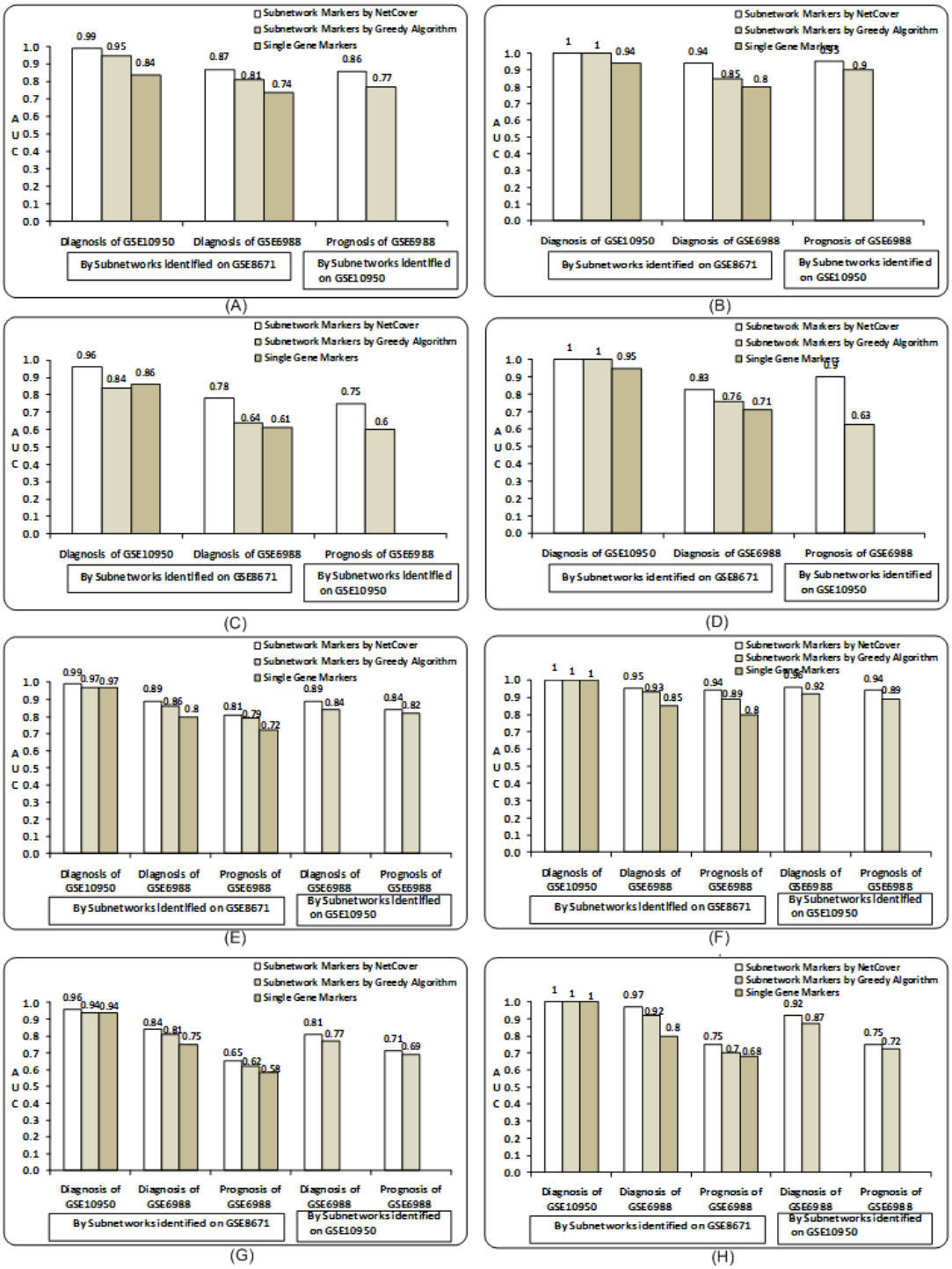


Fig. 2. Comparison of the classification performance of subnetwork markers identified by NetCover against subnetwork markers identified by Chuang *et al.*'s algorithm and single gene markers. In each figure, the AUC for diagnosis and prognosis of samples in *GSE8671*, *GSE10950*, and *GSE6988* datasets is shown for different combinations of classification and performance evaluation methods. For each configuration, average and maximum AUC (i.e., the best classification that can be obtained by accurately adjusting the number of features) are measured across number of markers ranging from 1 to 50. (A) Average, (B) maximum AUC for cross-classification (CC) using support vector machines (SVM). (C) Average, (D) maximum AUC for CC using Quadratic Regression (QR). (E) Average, (F) maximum AUC for leave-one-out cross-validation (LOOCV) using SVM. (G) Average, (H) maximum AUC for LOOCV using QR.

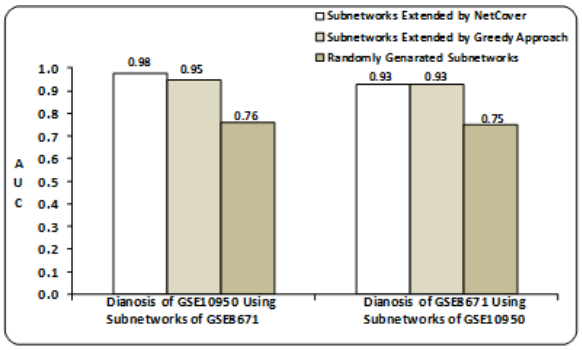


Fig. 3. Classification performance of coordinately dysregulated subnetworks associated with genes that are known to be susceptible for CRC, as compared to subnetworks associated with randomly selected genes.

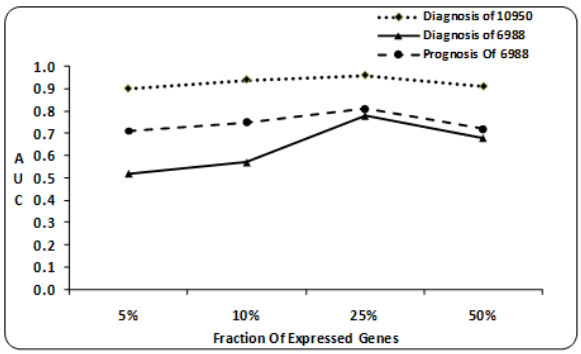


Fig. 4. Effect of binarization on the performance of NetCover. The graph shows the area under ROC curve for three instances with respect to fraction (α) of genes that are considered as "expressed".

SMAD4, TGFBR2, MYC, SRC and *DCC*.[28] To investigate whether the neighborhood of these genes contain coordinately dysregulated subnetworks that may be relevant in the manifestation of CRC, we discover coordinately dysregulated subnetworks associated with these genes using both NetCover and Chuang *et al.*'s greedy algorithm, on datasets *GSE8671* and *GSE10950*. Then, we use the subnetworks identified on each dataset to classify the samples in the other dataset. In Figure 3, the AUC provided by these classifiers is compared against the AUC provided by the same number of subnetworks associated with randomly selected genes. As can be seen in the figure, the subnetworks associated with the hallmarks of CRC demonstrate very high classification accuracy for both datasets. Furthermore, NetCover identifies larger subnetworks with better classification performance, as compared to the greedy algorithm. These results indicate that the coordinately dysregulated subnetworks associated with the CRC driver genes can be used to understand the effect of the mutations in the driver genes on the dysregulation of the other genes within the context of network dynamics. These subnetworks may also be useful in discovering proteins that are related to disease-causing genes but may not exhibit significant individual dysregulation.[32]

3.6. *Effect of Binarization*

In order to investigate the effect of binarization on the performance of the proposed algorithm and to choose the cutoff level at which to binarize the gene expression data, we fix α, which specifies the fraction of "expressed" genes in the dataset, at values of 5%, 10%, 25% and 50%. Then we use NetCover to discover the dysregulated subnetworks at each level of α and perform systematic experiments whose results are reported in Figure 4. It can be observed from the figure that the subnetworks discovered at $\alpha = 25\%$ show best performance for all instances. For this reason, we choose the cutoff level at which 25% of the genes are "expressed" to binarize the gene expression data.

4. Conclusion

In this paper, we propose a novel computational approach for identifying coordinately dysregulated subnetworks in a complex phenotype like cancer. Our algorithm integrates protein-protein interaction data with clinical gene expression data to capture the coordinated dysregulation of multiple interacting genes. Application of our algorithm on human colorectal cancer (CRC) datasets shows its potential in identifying subnetworks with high relevance to disease progression. However, our current algorithm is defined in terms of paired gene expression data; that is, datasets where there is a one-to-one relationship between control and phenotype samples. Extension of this approach to unpaired datasets will broaden the application of the proposed framework. Furthermore, the subnetwork identification algorithm implemented here is based on an approximation algorithm for the set-cover problem. Development of algorithms for more effective search of the subnetwork space may lead to identification of more biologically relevant subnetworks. Finally, detailed investigation of the subnetworks identified by NetCover may shed light into the network dynamics of CRC.

Acknowledgments

We would like to thank Rod Nibbe and Mark Chance for useful discussions on the systems biology of colorectal cancer. This work was supported, in part, by the National Institutes of Health Grant, UL1-RR024989 Supplement, from the National Center for Research Resources (Clinical and Translational Science Awards).

References

1. A. M. Glazier, J. H. Nadeau and T. J. Aitman, *Science* **298**, 2345(Dec 2002).
2. T. R. Golub, D. K. Slonim, P. Tamayo, M. G. C. Huard, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield and E. S. Lander, *Science* **286**, 531(Oct 1999).
3. C. M. Perou, T. Srlie, M. B. Eisen, M. van de Rijn, S. S. Jeffrey, C. A. Rees, J. R. Pollack, D. T. Ross, H. Johnsen, L. A. Akslen, ystein Fluge, A. Pergamenschikov, C. Williams, S. X. Zhu, P. E. Lnning, A.-L. Brresen-Dale, P. O. Brown and D. Botstein, *Nature* **406**, 747(Aug 2000).
4. D. G. Beer, S. L. Kardia, C. C. Huang, T. J. Giordano, A. M. Levin, D. E. Misek, L. Lin, G. Chen, T. G. Gharib, D. G. T. Michelle, L. Lizyness, R. Kuick, S. Hayasaka, J. M. Taylor, M. D. Iannettoni, M. B. Orringer and S. Hanash, *Nat. Med.* **8**, 816 (2002).
5. J. Lapointe, C. Li, J. P. Higgins, M. van de Rijn, E. Bair, K. Montgomery, M. Ferrari, L. Egevad, W. Rayford, U. Bergerheim, P. Ekman, A. M. DeMarzo, R. Tibshirani, D. Botstein, P. O. Brown, J. D. Brooks and J. R. Pollack, *PNAS* **101**, 811(Jan 2004).
6. S. M. Dhanasekaran, T. R. Barrette, D. Ghosh, R. Shah, S. Varambally, K. Kurachi, K. J. Pienta, M. A. Rubin and A. M. Chinnaiyan, *Nature* **412**, 822 (2001).
7. D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, E. S. Lander, M. Loda, P. W. Kantoff, T. R. Golub and W. R. Sellers, *Cancer Cell* **1**, 203(Mar 2002).
8. E. LaTulippe, J. Satagopan, A. Smith, H. Scher, P. Scardino, V. Reuter and W. L. Gerald, *Cancer Res.* **62**, 4499(Aug 2002).
9. S. Ramaswamy, K. N. Ross, E. S. Lander and T. R. Golub, *Nat. Genet.* **33**, 49(Jan 2002).
10. R. Jansen, D. Greenbaum and M. Gerstein, *Genome Res.* **12**, 37 (2002).
11. E. Segal, H. Wang and D. Koller, *Bioinformatics* **19**, I264 (2003).
12. T. Ideker, O. Ozier, B. Schwikowski and A. F. Siegel, *Bioinformatics* **18**, 233 (2002).
13. D. Rajagopalan and P. Agarwal, *Bioinformatics* **21**, 788 (2005).
14. L. Cabusora, E. Sutton, A. Fulmer and C. Forst, *Bioinformatics* **21**, 2898 (2005).
15. S. Nacu, R. Critchley-Thorne, P. Lee and S. Holmes, *Bioinformatics* **23**, 850 (2007).
16. H. Y. Chuang, E. Lee, Y. T. Liu, D. Lee and T. Ideker, *Molecular Systems Biology* **3** (2007).
17. D. Anastassiou, *Molecular Systems Biology* **3** (2007).
18. J. Quackenbush, *Nat Genet* **32 Suppl**, 496(December 2002).
19. T. Akutsu, S. Miyano and S. Kuhara, *Pacific Symposium on Biocomputing.* , 17 (1999).
20. M. Koyutürk, W. Szpankowski and A. Grama, Biclustering gene-feature matrices for statistically significant dense patterns, in *in: IEEE Computer Society Bioinformatics Conf* , 2004.
21. T. Akutsu and S. Miyano, Selecting informative genes for cancer classification using gene expression data, in *Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing*, 2001.
22. I. Shmulevich and W. Zhang, *Bioinformatics* **18**, 555 (2002).
23. M. Dash, Feature selection via set cover, in *KDEX '97: Proceedings of the 1997 IEEE Knowledge and Data Engineering Exchange Workshop*, (IEEE Computer Society, Washington, DC, USA, 1997).
24. R. K. Nibbe, R. Ewing, L. Myeroff, M. Markowitz and M. Chance, *Mol Cell Prot* **9**, 827 (2009).
25. I. Ulitsky, R. M. Karp and R. Shamir, Detecting disease-specific dysregulated pathways via analysis of clinical expressio profiles, in *Proceedings of 12th Int'l Conf. Research in Comp. Molecular Biology (RECOMB'08)*, 2008.
26. M. R. Garey and D. S. Johnson, in *Computers and Intractability, A Guide to the Theory of NP-Completeness*, (W.H. Freeman, 1979)
27. V. Chvatal, *Mathematics of Operations Research* **4**, 233 (1979).
28. F. Macdonald, C. H. J. Ford and A. G. Casson, Colorectal cancer, in *Molecular Biology of Cancer*, 2005
29. J. Sabates-Bellver, L. G. Van der Flier, M. de Palo, E. Cattaneo, C. Makee, H. Rehrauer, E. Laczko, M. A. Kurowski, J. M. Bujnicki, M. Menigatti, J. Luz, T. V. Ranalli, V. Gomes, A. Pastorelli, R. Faggiani, M. Anti, J. Jiricny, H. Clevers and G. Marra, *Molecular Cancer Res.* **5(12)**, p. 1263(Dec 2007).
30. X. Jiang, J. Tan, J. Li, S. Kivime, X. Yang, L. Zhuang, P. L. Lee, M. T. Chan, L. W. Stanton, E. T. Liu, B. N. Cheyette and Q. Yu, *Cancer Cell* **13(6)**, 529(Jun 2008).
31. D. H. Ki, H. C. Jeung, C. H. Park, S. H. Kang, G. Y. Lee, W. S. Lee, N. K. Kim, H. C. Chung and S. Y. Rha, *International Journal of Cancer* **121**, 2005 (2007).
32. N. Turner, A. Tutt and A. Ashworth, *Nat Rev Cancer* **4**, 814 (2004).

SUBSPACE DIFFERENTIAL COEXPRESSION ANALYSIS: PROBLEM DEFINITION AND A GENERAL APPROACH

GANG FANG, RUI KUANG, GAURAV PANDEY, MICHAEL STEINBACH, CHAD L. MYERS and VIPIN KUMAR

Department of Computer Science, University of Minnesota, Twin Cities
200 Union Street SE, Minneapolis, MN 55455, USA
E-mail: {gangfang, kuang, gaurav, steinbac, cmyers, kumar}@cs.umn.edu

In this paper, we study methods to identify differential coexpression patterns in case-control gene expression data. A differential coexpression pattern consists of a set of genes that have substantially different levels of coherence of their expression profiles across the two sample-classes, i.e., highly coherent in one class, but not in the other. Biologically, a differential coexpression patterns may indicate the disruption of a regulatory mechanism possibly caused by disregulation of pathways or mutations of transcription factors. A common feature of all the existing approaches for differential coexpression analysis is that the coexpression of a set of genes is measured on all the samples in each of the two classes, i.e., over the *full-space* of samples. Hence, these approaches may miss patterns that only cover a subset of samples in each class, i.e., *subspace patterns*, due to the heterogeneity of the subject population and disease causes. In this paper, we extend differential coexpression analysis by defining a subspace differential coexpression pattern, i.e., a set of genes that are coexpressed in a relatively large percent of samples in one class, but in a much smaller percent of samples in the other class. We propose a general approach based upon association analysis framework that allows exhaustive yet efficient discovery of subspace differential coexpression patterns. This approach can be used to adapt a family of biclustering algorithms to obtain their corresponding differential versions that can directly discover differential coexpression patterns. Using a recently developed biclustering algorithm as illustration, we perform experiments on cancer datasets which demonstrates the existence of subspace differential coexpression patterns. Permutation tests demonstrate the statistical significance for a large number of discovered subspace patterns, many of which can not be discovered if they are measured over all the samples in each of the classes. Interestingly, in our experiments, some discovered subspace patterns have significant overlap with known cancer pathways, and some are enriched with the target gene sets of cancer-related microRNA and transcription factors. The source codes and datasets used in this paper are available at http://vk.cs.umn.edu/SDC/.

Keywords: Differential coexpression; differential biclustering; differential network analysis; association analysis

1. Introduction

Diseases are often caused by perturbations in networks of genes or their products that are working together to keep a cell in a healthy state. DNA microarrays are one of the most popular technologies for studying these perturbations and understanding their effect on the expression of genes at a large scale, and eventually linking them to diseases. The genome-wide expression profiles of many types of diseases, particularly tumors, have been analyzed, and several associations have been identified between gene expression profiles and phenotypes corresponding to different stages of cancer.[30] Traditional analysis of gene expression data for this task focuses on the identification of (groups of) genes with substantially different expression values (up- or down-regulated) across sample-classes of interest, commonly known as differentially expressed (DE) genes (or patterns).[8] An example of such a group of differentially expressed genes is shown in Figure 1(a), where these genes have significantly higher expression levels in the disease class than in the control class.

However, given that diseases are often caused by the disruption of a system, or network, of genes, identifying only the individual differentially expressed genes may not be adequate for discovering the underlying mechanisms of all the diseases. An important example of such mechanisms is the dysregulation of signaling pathways in cancer.[14] A complementary view for studying these mechanisms is provided by a differential coexpression pattern (DC),[20,22,27,34] which is defined as a set of genes that have substantially different levels of coherence of their expression profiles in the two sample-classes, i.e., highly coherent in one class, but not in the other. An example of a DC pattern is shown in Figure 1(b), where the constituent genes are either all up-, down-, or neutrally-regulated for each sample in the control group (shown by the vertical streaks), but they do not follow any particular trend in the disease group. Biologically, a differential coexpression pattern may indicate the disruption of a regulatory mechanism possibly caused by the disregulation of a pathway[20] or a mutation of a transcription factor,[16,17] among other mechanism. Figure 1(c) illustrates one of these mechanisms, where the mutation of a regulator causes the disruption of the normal activity of a pathway. Specifically, $G0$ is a dominant regulator of

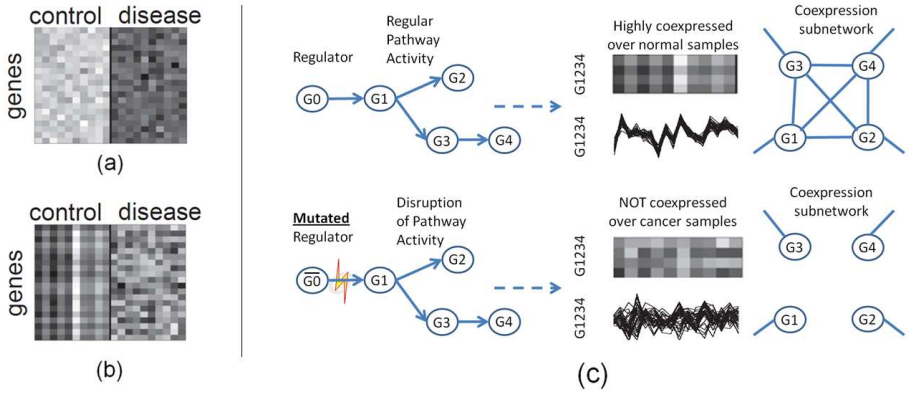


Fig. 1. Illustration of (a) a differential expression (DE) pattern, (b) a differential coexpression (DC) pattern, and (c) a possible mechanism for the occurrence of a DC pattern due to the mutation of a regulator. Note that, while $G0$ is a dominant regulator of $G1$-$G4$, the latter genes are also regulated by other *independent* regulators, that are not shown in the two pathway graphs for simplicity. ((a) and (b) taken from Kostka and Spang (2004).[20])

$G1$-$G4$ that leads to the coordinated and hence coherent expression of all of them. However, once mutated in the disease state, $G0$ is unable to regulate these genes, and their regulation may be taken over by other *independent* regulators that may only be active in the disease state. Now, since the regulation of G1-G4 is independent, they are no longer coordinated, thus leading to the disruption of the coherence of their expression. Therefore, DC patterns can serve as biomarker candidates for some diseases, e.g. cancer and its subtypes,[16,20,47] as well as for differentiating between evolutionarily-related species.[17] Furthermore, at a representation level, a DE pattern can be considered as a connected subgraph of a coexpression network, which is intact in one sample-class but not connected in the other. Such a convenient representation of these patterns can be very useful for their visualization and understanding, and we present some examples of this in Section 3.

Owing to their definition, differential coexpression patterns cannot be discovered via univariate analysis, since the coherence of expression values of a group of genes has to be measured collectively. Corresponding to this need, several techniques for identifying DC patterns have been proposed in the literature, the first ones of which only searched for gene-pairs with sufficiently different correlations (or other statistical measures) between the two classes.[21–23,34] Extending this to larger groups of genes, differential coexpression has also been studied in the context of clustering[17,44] and coexpression networks,[7,11,12,27,47,48] where a cluster or a sub-network of genes is considered differentially coexpressed if they collectively have different pairwise coexpression across the sample-classes of interest. Some algorithms also employ differential coexpression measures collectively for a set of genes,[20,29] instead of only pairwise coexpression measures. Recently, some studies have adopted a related but different perspective and have proposed methodologies for identifying differentially coexpressed gene-pathway pairs[31] and pathway-pathway pairs.[6]

Despite the differences in the methodologies adopted by these approaches for finding DC patterns, a feature common to all of them is that the coexpression of a set of genes is measured over all the samples in each of the two classes, i.e., over the *full space* of samples. For instance, the example shown in Figure 1(b) is a full-space DC pattern. However, as pointed out for the discovery of differentially expressed genes,[39,41,46] the causes of diseases as well as the population affected by them, are often heterogeneous in nature. In such a scenario, full-space approaches may not always be appropriate and may ignore patterns that cover only a subset of the samples in each class, i.e., *subspace patterns*. For instance, a set of genes may only be coexpressed over 60% of the samples in the normal class, and may not be even slightly coexpressed over any of the samples in the disease class, thus qualifying to be a valid subspace pattern. However, this pattern may not be uncovered if the discovery algorithm requires the constituent genes to be coexpressed over all the samples in the normal class. Indeed, even if a pattern can be discovered by both full-space approaches and subspace approaches, the latter can better indicate the subgroup of samples on which the pattern is coexpressed, and thus may allow further study of the different causes of diseases and different demographics among subgroups of samples, which may potentially help personal diagnosis and treatment. These challenges call for the design of new approaches that

can discover patterns that only show differential coexpression over subsets of the samples in the two classes, and can also indicate these subsets as a companion to the patterns. Interestingly, similar challenges faced by traditional clustering approaches have motivated the design of a variety of biclustering algorithms.[5,25]

In this paper, we address these challenges by extending differential coexpression analysis to enable the discovery of subspace DC patterns. We define these patterns as sets of genes that are coexpressed over a relatively large percent of the samples in one class, but in a much smaller percent of samples in the other class. Following this definition, we propose a general approach based upon association analysis framework[1] that allows exhaustive[a] yet efficient discovery of subspace differential coexpression patterns. This approach can be used to adapt a family of biclustering algorithms that have antimonotonicity[24,28,43,49,51] to obtain their corresponding differential versions that can directly discover differential coexpression patterns. Specifically, we illustrate the features of our approach by extending a recently developed biclustering algorithm.[28] Experiments using this approach on lung cancer datasets demonstrate the existence of subspace differential coexpression patterns in real-life data. Permutation tests demonstrate the statistical significance for a large number of discovered subspace patterns, many of which can not be discovered if they are measured over all the samples in each of the classes. Interestingly, some discovered patterns also have a significant overlap with known cancer pathways, and some are enriched with the target gene sets of a cancer-related microRNA and a cancer-related transcription factor. These results suggest that subspace DC patterns may aid in developing new understanding about the mechanisms underlying cancer and other diseases.

2. Proposed Approach

In this section, we first extend differential coexpression analysis to subspace patterns, then we will describe a general approach for the discovery of subspace differential coexpression patterns.

2.1. *Subspace Differential Coexpression Analysis*

A subspace differential coexpression pattern is a set of genes that are highly coexpressed in a relatively large percent (not necessarily all) of samples in one class, but in a much smaller percent of samples in the other class. We formulate the problem of subspace differential coexpression pattern discovery as follows. Let D be a gene expression dataset with a set of p genes, $G = \{g_1, g_2, ..., g_p\}$, and two classes of samples, A and B, which can be considered as cases and controls of size N_A and N_B, respectively, i.e., $A = \{a_1, a_2, ..., a_{N_A}\}$ and $B = \{b_1, b_2, ..., b_{N_B}\}$. Let Ψ be a coexpression measure for a set of genes α ($\alpha \subseteq G$). To illustrate, this measure could be a test as to whether the minimum of the pairwise correlation of the expression profiles of the genes in α is above a particular threshold. We use $A_\Psi(\alpha)$ ($B_\Psi(\alpha)$) to denote the subset of samples in A (B) on which α is coexpressed, i.e., $A_\Psi(\alpha) \subseteq A$ and $B_\Psi(\alpha) \subseteq B$. The two ratios, $\frac{|A_\Psi(\alpha)|}{|A|}$ and $\frac{|B_\Psi(\alpha)|}{|B|}$ are respectively the percentage of samples in A and B on which α is coexpressed. They are denoted as $R_A^\Psi(\alpha)$ and $R_B^\Psi(\alpha)$, respectively. The absolute difference of these two ratios can be used to measure the subspace differential coexpression of α:

Definition 2.1. Subspace Differential Coexpression (SDC)

$$SDC^\Psi(\alpha) = |R_A^\Psi(\alpha) - R_B^\Psi(\alpha)| \tag{1}$$

Given a threshold d, a set of genes α ($\alpha \subseteq G$) is called *$d-$differentially coexpressed if* $SDC^\Psi(\alpha) \geq d$. Then, the problem of subspace differential coexpression pattern discovery with reference to a threshold d can be formulated as discovering all the *$d-$differentially coexpressed* patterns.

We will explain our approach for addressing this problem using Figure 2, which shows a number of types of subspace and full-space, differentiating and non-differentiating, coexpression patterns. Figure 2(a) shows a conceptual example of a differential full-space pattern, while Figure 2(b) shows a conceptual example of a differential subspace pattern. Figures 2(c) and 2(d) are examples of non-differential patterns. Although Figure 2(e) is a differential full-space pattern, it contains a redundant gene, i.e., the dashed curve.

[a]Given a threshold, an exhaustive search guarantees to discover all the patterns w.r.t. that threshold. Different from brute-force search, exhaustive search may avoid exploring the whole search space by pruning a large number of patterns that are guaranteed to disqualify the threshold.

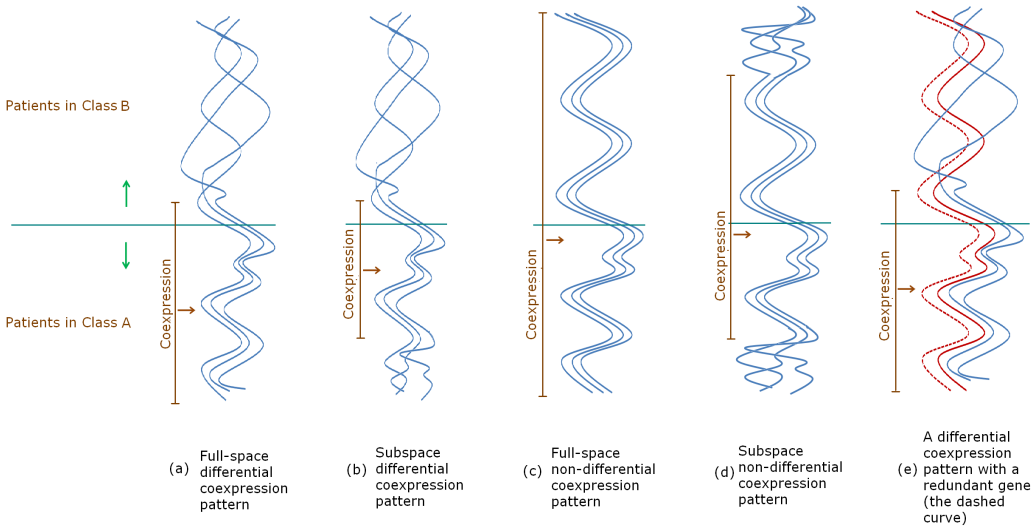


Fig. 2. Different types of full-space and subspace, differential and non-differential coexpression patterns. Each curve denotes the expression values of a gene on all the samples. The horizontal line separates all the samples into class A and B. Five patterns are illustrated, with a brown line indicating the samples on which the set of genes are coexpressed.

Given Definition 2.1, an effective mining algorithm is expected to discover patterns like (a), (b) and (e) in Figure 2, but not the patterns that are equally coexpressed in the two classes (as shown in Figure 2(c) and (d)). However, if we take a further look at pattern (e), we can observe that, although the four genes together have differential coexpression, two genes are coexpressed in both of the two classes (the two red curves). Only one of the two coexpressed genes is enough to form a differential coexpression pattern with the other two genes (the two blue curves). Indeed, including both genes would lead to redundancy that is conceptually unappealing and increases the number of patterns, without improving the SDC measure given in Definition 2.1. We considered the patterns like (e) as redundant ones that can be represented by their subsets. Therefore, for the control of redundant genes as well as for efficient pattern discovery, patterns like (e) will also be pruned together with the non-differential ones like (c) and (d) in the pattern mining process.

A common property of the five patterns in Figure 2 is that, they are all coexpressed in a large percent of samples in class A. In the meanwhile, the common property of the three patterns that are expected to be pruned, namely (c),(d) and (e), is that they all have at least one pair of genes that are coexpressed in a large percent of samples in class B. Motivated by these two observations, we refine the target of subspace differential coexpression pattern mining as those sets of genes that are coexpressed in a relatively large percent of samples in one class, while all of the pairs of genes in the set are coexpressed in a much smaller percent of samples in the other class. Mathematically, we define a measure for this refined criteria as follows:

Definition 2.2. Refined Definition of Subspace Differential Coexpression ($\widetilde{SDC}$) (Assume $R_A^\Psi(\alpha) \geq R_B^\Psi(\alpha)$)

$$\widetilde{SDC}^\Psi(\alpha) = R_A^\Psi(\alpha) - max_{i,j\in\alpha}(R_B^\Psi(\{i,j\})) \tag{2}$$

$\widetilde{SDC}$ is computed as the difference between the percent of samples in class A on which α is coexpressed and the maximal percent of samples in class B on which a size-2 subset of α is coexpressed. A large value for $\widetilde{SDC}$ indicates that a set of genes, α is coexpressed on a much larger percent of samples in class A compared to the coexpression of any size-2 subset of α in class B. Therefore, given a proper threshold, d, $\widetilde{SDC}$ can differentiate interesting subspace differential coexpression patterns like patterns (a) and (b) from uninteresting patterns like patterns $(c) - (e)$.

Mathematically, for some coexpression measures, $\widetilde{SDC}$ has another property called antimonotonicity, which basically means that $\widetilde{SDC}(\alpha)$ is guaranteed to be no less than the $\widetilde{SDC}$ of any superset of α. For $\widetilde{SDC}(\alpha)$ to have the antimonotonicity property, it is sufficient that the coexpression measure used to define $\widetilde{SDC}(\alpha)$ is antimonotonic (A formal proof is given in Fang et al.[9]). Indeed, the coexpression measures used in several

existing association-based and subspace clustering based biclustering algorithms have this property.[24,28,43,49,51] This antimonotonicity property guarantees that, given a threshold, d, $\widetilde{SDC}$ can be used in a systematic yet efficient pattern mining framework, namely Apriori,[1] to discover all and only the patterns with $\widetilde{SDC} \geq d$. We briefly describe the computational algorithm for this approach in Section 2.2.

2.2. *Computation Algorithm*

The Apriori framework is essentially a bottom-up exhaustive combinatorial search framework initially designed for association analysis on binary data. Different from brute-force search, given an antimonotonic measure M and a threshold m, the Apriori search algorithm can avoid exploring the whole search space of all sets of items (genes in our case) by pruning a large number of candidates that are guaranteed to disqualify the threshold based on the antimonotonicity of M.

The process of searching patterns with $\widetilde{SDC} \geq d$ in the Apriori framework can be viewed as the generation of a level-wise pattern tree. Every level of the tree contains patterns with the same number of genes. If the level is increased by one, the pattern size (number of genes in each pattern) is also increased by one. Every pattern has a branch (sub-tree) which contains all the supersets of this pattern. The search is breadth-first. We first check all the patterns at the second level, since the elemental component of differential coexpression analysis is a pair of genes. If a pattern does not satisfy the user-specified $\widetilde{SDC}$ threshold d, the whole branch corresponding to this pattern can be pruned without the need of further checking. This is guaranteed by the antimonotone property of $\widetilde{SDC}$ measures.[9] Following this approach, the pattern tree grows level-by-level until all the qualified patterns have been discovered. This algorithm is systematic yet efficient for handling large-scale datasets. Note that, in Definition 2.2, it is assumed that $R_A^\Psi(\alpha) \geq R_B^\Psi(\alpha)$. In practice, the algorithm will be run twice, one time to find patterns for which $R_A^\Psi(\alpha) \geq R_B^\Psi(\alpha)$, and the other to find patterns for which $R_B^\Psi(\alpha) \geq R_A^\Psi(\alpha)$. Use of the general measure $\widetilde{SDC}$ in the Apriori framework allows the effective pruning of non-differential coexpression patterns like (c) and (d), and also controls gene redundancy in patterns like (e). $\widetilde{SDC}$ also provides the antimonotonicity that allows exhaustive yet efficient discovery of differential coexpression patterns like (a) and (b) (Figure 2) in the Apriori framework. We will use $\widetilde{SDC}$-*Apriori* to denote the the approach of using the general measure $\widetilde{SDC}$ in the Apriori algorithm.

The coexpression measures used in several existing association-analysis-based and subspace-clustering-based biclustering studies have the antimonotonicity property[24,28,43,49,51] and can be adapted to yield their corresponding differential versions that can directly discover differential coexpression patterns. Because of the complementarity of biclustering algorithms (i.e. they may discover patterns in common with each other, as well as some unique to their formulation), their corresponding differential versions are also complementary to each other.

2.3. *DiffRange: an illustration of* $\widetilde{SDC}$

In this paper, we shall use a specific instance of this approach based on a recently proposed antimonotonic coexpression measure, namely range-support.[28] This measure is intended for the discovery of constant-row biclusters[b][25] in the Apriori framework. Conceptually, a range-support pattern is a set of genes that are coexpressed (the expression value of the set of genes fall within a close range) over a set of conditions in a gene expression data matrix. Let $RangeSup_A^r(\alpha)$ denote the range-support of α in class A (an instantiation of $R_A^\Psi(\alpha)$), i.e. the percentage of samples in class A that fall within the predefined range threshold r. From Definition 2.2($\widetilde{SDC}$), the corresponding differential range-support measure *DiffRange* (Differential Range-support) can be adapted:

Definition 2.3. Given a range threshold r, the *DiffRange* of a subset of genes α ($\alpha \subseteq G$) on class A and B

$$DiffRange(\alpha) = RangeSup_A^r(\alpha) - max_{i,j \in \alpha}(RangeSup_B^r(\{i,j\})) \quad (3)$$

3. Experimental Results

In this section, we describe the experimental design for the analysis of the subspace differential coexpression patterns discovered by *DiffRange*. The, we present experimental results which demonstrate that the proposed

[b]In a constant-row bicluster, the set of genes have similar expression values on each condition/sample.

general approach discovers statistically significant and biologically relevant subspace differential coexpression patterns in real-life data. .

3.1. *Datasets and Preprocessing*

In the experiments, three lung cancer datasets[3,35,36] are used, which are all generated with Affymetrix microarrays.[c] To have a larger sample size for better illustration of the existence of subspace patterns and of their statistical significance[d], we combined the three datasets resulting in 102 samples with lung cancer and 67 normal samples (total 169 samples). Across the three datasets, 8787 genes are common. We preprocessed the three datasets with RMA-normalization.[18] Additional cross-platform normalization algorithms[2,32] were also tested and gave similar results, so only RMA normalized results are included here. The effect of different normalization methods on differential coexpression pattern mining will be studied in future work.

Instead of normalized gene expression data, we used rank-converted values, i.e., the expression values are converted to expression ranks ranging from 1 to 169 (number of samples) separately for each gene (similar as used in *Spearman's rank correlation*). Our analysis shows that rank-conversion can allow the discovery of patterns containing genes with different ranges of expression values but still showing differential coexpression. Thus, we focus on the analysis of the patterns discovered on rank-converted data, on which more patterns are discovered. The patterns discovered on the data with expression value are presented on our website. Note that, rank-transformation is especially useful for *DiffRange*, since it is based on the biclustering algorithm[28] designed to find constant-row patterns.[25] Such rank-transformation may not be required in the $\widetilde{SDC}$*-Apriori* framework for other biclustering algorithms that are able to find coherent additive, coherent multiplicative or coherent evolution biclusters.[25]

Higgins et al.[15e] collected a list of genes that are shown to be related to cancer. Out of the 8787 genes in the dataset, 1975 are on the cancer gene list. In the following experiments, we analyze the subspace patterns discovered on these 1975 genes (rank-converted data, and denoted as dataset D'), because cancer genes are more likely to have disregulated patterns and based on the existing knowledge of these cancer genes, the evaluation on the patterns discovered from these genes can better illustrate the biological relevance of subspace differential coexpression patterns. Note that although these 1975 genes are known to be related to cancer, the subspace differential coexpression patterns discovered on them can provide new insights about their relationship with cancer, e.g., by identifying the interactions among individual cancer genes.

3.2. *Pattern Discovery*

With parameters $r = 0.2$[f], and $d = 0.2$, *DiffRange* is used in the Apriori framework to discover subspace DC patterns on D'. Most patterns are of size-2 (gene-pairs), but there are also size-3 and size-4 patterns (no larger size patterns are discovered for the selected parameters). To control the redundancy of genes among size-3 and size-4 patterns,[g] we order them by decreasing SDC value and sequentially select a subset of the patterns in which none of the pairs of patterns have greater than 25% overlap of genes. This compact set has 95 patterns (88 size-3 patterns and 7 size-4 patterns). Figure 3(a) shows the size and SDC value for each discovered DC pattern.

3.3. *Are the discovered subspace differential coexpression patterns statistically significant?*

Due to the issues of low sample size and high-dimensionality for data sets used for problems such as biomarker discovery, many patterns may be falsely associated with the class label by random chance, especially when a large number of combinations of genes are searched. This raises the multiple-hypothesis testing problem.[33] In this paper, a permutation test is used to evaluate the statistical significance of the discovered subspace DC patterns. Specifically, the original class labels are randomly shuffled 1000 times. For each random labeling, the same

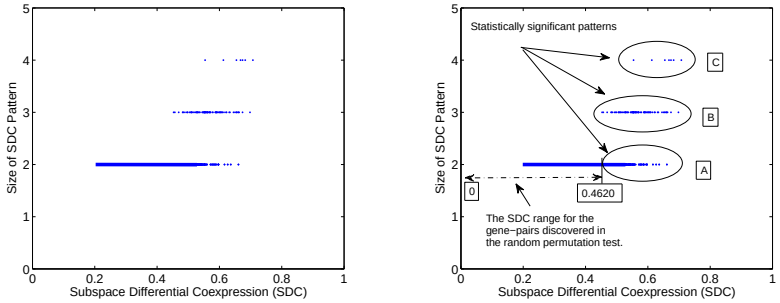
[c]The first two use platform HG-U95A, while the other uses platform HG-U133A

[d]The patterns discovered from the three datasets separately are not statistically significant in the permutation test (refer to Section 3.3 for details), due to the low sample size of each individual datasets.

[e]In this paper, we union the two lists respectively downloaded in October 2008 and June 2009, with a total of 2622 genes

[f]In the rank-converted data, this means k genes have coherent expression if the rank difference of their expression is less than 20% of the 169 samples, i.e. 33.

[g]Note that in the discussion of pattern (e) in Fig. 2, the redundancy is within a pattern rather than among the patterns like here.



(a) Size and SDC for the DC patterns discovered with the original class labels (Each dot is a pattern).

(b) Statistically significant DC patterns according to the permutation test.

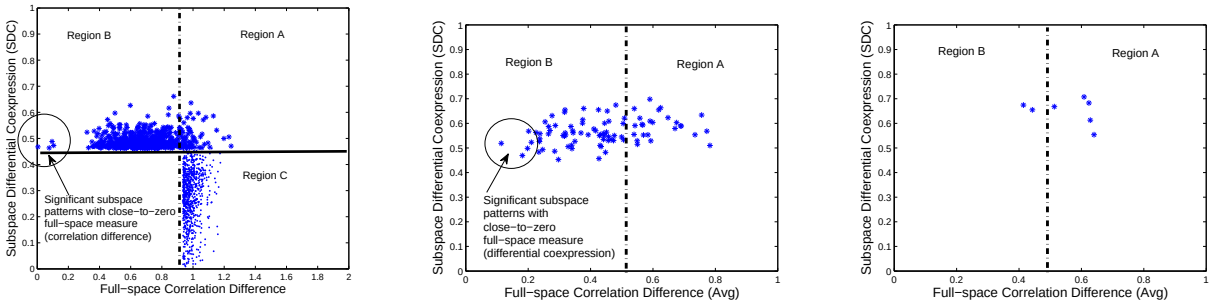
Fig. 3. Patterns discovered with the original class labels (a) and indication of the statistically significant ones (b).

DiffRange parameters ($r = 0.2$ and $d = 0.2$) are used to discover a set of patterns. With the 1000 randomized labels, only size-2 patterns are discovered, with SDC values in the range of $[0, 0.4620]$. Figure 3(b) indicates this range by overlaying a double-arrow on top of the patterns discovered with the original class labels (as shown in Figure 3(a)). Considering 0.4620 as a statistical significance cutoff, Ellipse A indicates the 560 statistically significant gene-pairs whose SDC value was never exceeded by any random pattern. In addition, since there are no size-3 and size-4 patterns discovered with randomized labels in the permutation test, the 88 size-3 patterns (Ellipse B) and 7 size-4 pattern (Ellipse C) are also considered statistically significant. Note that, although a differential coexpression pattern can be highly coexpressed in either the cancer class or the normal class,[21,22] all the statistically significant patterns in the three ellipses are highly coexpressed in the normal class while less coexpressed in the cancer class.

3.4. *How differentially coexpressed are the discovered subspace patterns when measured over full-space?*

In this experiment, we measure the full-space differential coexpression for the statistically significant subspace patterns selected based on the above permutation test, i.e., the 560 gene-pair patterns and the 88 size-3 patterns and the 7 size-4 patterns. We will show that there are subspace patterns that have close-to-random differential coexpression when considered as full-space patterns. A variety of full-space differential coexpression measures are proposed in existing work as discussed in section 1. As used in several studies,[20,34,44] we will use the correlation difference of a pattern between the two classes for illustration purpose. For a gene-pair pattern, correlation difference is just the difference of the two correlations respectively in the two classes. For a pattern of size greater than 2, we compute the difference between the average pair-wise correlation in the normal and cancer class to measure the correlation difference.

The three subfigures in Figure 4 plot the correlation difference and SDC for the statistically significant size-2, size-3, and size-4 patterns, respectively. The three dashed lines indicate the statistical significance cutoff of correlation difference for size-2, size-3 and size-4 patterns (0.9361, 0.5176 and 0.4953), respectively, which is also decided via permutation test. For the gene-pair patterns (Figure 4(a)), several observations can be made: (i) some patterns are considered statistically significant by both correlation difference and SDC (region A); (ii) some gene-pairs are considered significant only by SDC but not by correlation difference (region B). Among these patterns, several pairs have close-to-zero correlation difference(within the circle), which means they show very little differential coexpression when considered as full-space patterns; and (iii) there are also 801 gene-pairs that are only considered significant in terms of correlation difference but not by SDC (region C). This is as expected since many factors can affect the discovery of DC patterns, e.g. different coexpression measures, different mining algorithms, and the parameters used in the algorithms. Our highlight is the existence of subspace differential coexpression patterns that show close-to-random differential coexpression when considered as full-space patterns. Similar observation can also be made in Figures 4(b) and 4(c) which respectively plot the correlation difference for the size-3 and size-4 patterns.



(a) The 560 gene-pair (stars). (b) The 88 size-3 patterns (c) The 7 size-4 patterns.

Fig. 4. Illustration of the full-space differential coexpression (correlation difference) for the discovered statistically significant subspace differential coexpression patterns. The dashed lines in $(a) - (c)$ indicate the statistical significance cutoffs for correlation difference for size-2, size-3 and size-4 patterns respectively. The solid line in (a) is the statistical significance cutoff for SDC (0.4620). There are no corresponding lines in (b) and (c) because all the patterns of size 3 and 4 are statistically significant in terms of SDC as discussed in section 3.3. Region A contains patterns that are considered significant by both correlation difference and SDC; Region B has patterns that are not significant as full-space patterns, several of which have close-to-zero correlation difference (within the circle); and Region C shows the significant full-space patterns that are not discovered by SDC.

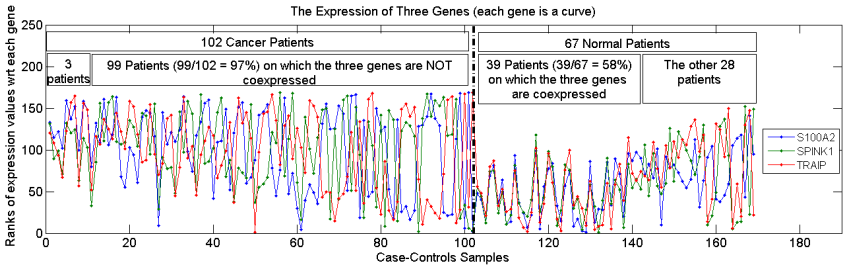


Fig. 5. A statistically significant subspace differential coexpression pattern, for which the minimal pairwise correlation in the normal class is only 0.28. For better visualization, samples are sorted by increasing range of expression ranks separately in the two classes (similar for Figures 6(a), 6(b) and 7(a)).

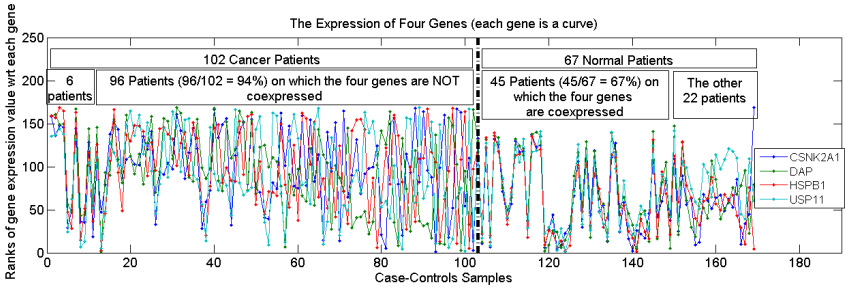
In Figure 5, we illustrate a subspace DC pattern with very small correlation difference (0.19). This pattern is coexpressed in only 58% of the normal samples [h], and the minimal pairwise correlation of the genes in this pattern over all the normal samples is only 0.28. For this pattern, it is not reasonable to assume that the genes are coexpressed on all the normal samples. Furthermore, discovering the pattern as a subspace DC pattern can explicitly show the subgroup of samples on which the three gene show coexpression, i.e., the 39 normal samples and the 3 cancer samples in the cancer class. This allows further analysis of the difference between the 39 normal samples with the pattern and the 28 without it (e.g., different demographic characteristics), which may help personalized diagnosis and treatment.

The existence of subspace patterns that show small and insignificant differential coexpression when considered as full-space patterns demonstrates the potential usefulness of subspace differential coexpression analysis. Next, we will evaluate the biological relevance of the discovered subspace patterns.

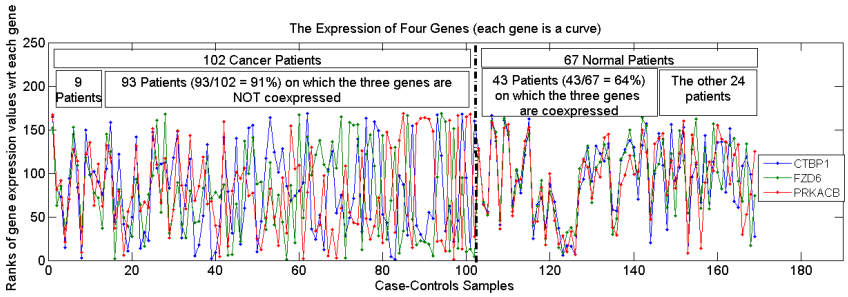
3.5. *Are the discovered subspace differential coexpression patterns biologically relevant?*

Quantitatively, two enrichment experiments are used to evaluate the biological relevance of the discovered subspace differential coexpression patterns: (i) enrichment with ten known cancer-related signaling pathways [i],[15] (ii) enrichment with the 5452 gene sets in the Molecular Signature database (MSigDB)[j].[37] Since patterns of size-2

[h]This is with respect to the *DiffRange* parameters used to discover the patterns, $r = 0.2$. Similar for Figures 6(a) and 6(b)
[i]http://cbio.mskcc.org/CancerGenes/Select.action
[j]Specifically, MSigDB contains 386 positional gene sets, 1892 curated gene sets, 837 motif gene sets, 883 computational gene sets, and 1454 GO gene sets. http://www.broadinstitute.org/gsea/msigdb/



(a) The pattern that is enriched with the $TNF\alpha/NF\kappa B$ signaling pathway (enrichment p-value 0.0011).



(b) The pattern that is enriched with the WNT signaling pathway (enrichment p-value 0.0042).

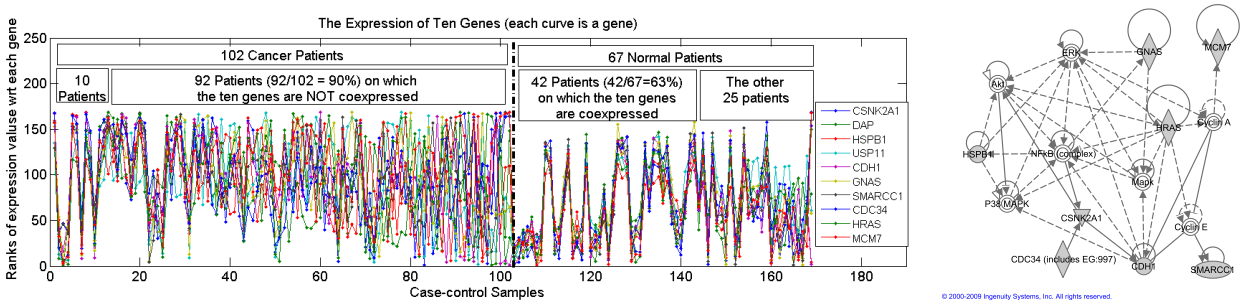
Fig. 6. Two patterns that are respectively enriched with the $TNF\alpha/NF\kappa B$ and the WNT signaling pathway.

are difficult to assess in terms of enrichment, we perform the two biological evaluations only for the 95 patterns of size 3 or 4. Briefly, (i) six patterns have an overlap of 2 or more genes with one of the ten known cancer-related pathways, (ii) in the MSigDB enrichment, 40 patterns have enrichment p-value less than 0.001, among which five have p-value less than 0.0001. Detailed enrichment results can be found on the paper website.

Note that, due to the limited knowledge about differentially coexpressed patterns, the current stage of differential coexpression pattern mining is still hypothesis generation rather than hypothesis verification, as discussed in Kostka and Spang.[20] Indeed, since all the 95 patterns are statistically significant in the permutation test, and all the genes contained in the 95 patterns are known cancer-related genes[15] (as described in section 3.1), they can be considered as hypotheses that may lead to new understanding of the interactions among them, and of the relationship between differential coexpression and cancer mechanism. Therefore, in addition to the above standard enrichment analyses, we will illustrate and discuss several interesting patterns that are enriched with known cancer pathways, or target sets of cancer-related microRNAs and transcription factors.

Figure 6 displays two patterns that are enriched with the $TNF\alpha/NF\kappa B$ signaling pathway and the WNT signaling pathway respectively. Several observations can be made from these two figures. Firstly, they both show strong differential coexpression, i.e. they are both highly coexpressed in the normal class, and much less coexpressed in the cancer class. Secondly, both patterns are subspace differential coexpression patterns, i.e., they show coexpression in only 67% and 64% of the normal samples respectively. Similar to the pattern shown in Figure 5, these two patterns are coexpressed in only about two-third of the normal samples. Discovering them as subspace patterns also points out the subgroup of samples covered by them. This allows further study of the different causes of diseases and the different demographics among subgroups of samples. Finally, both the $TNF\alpha/NF\kappa B$ and WNT signaling pathways have been shown to be related to lung cancer.[19,40] Discovering the differential coexpression patterns enriched with these pathways may shed new light on the understanding of the two pathways and their relationships to cancer mechanism.

Among the six patterns that are enriched with at least one cancer pathway, three are enriched with the $TNF\alpha/NF\kappa B$ pathway. In Figure 7(a), the union of the three patterns, containing ten genes, are plotted. All the ten genes are known cancer-related genes.[15] Out of the ten genes, six overlap with the $TNF\alpha/NF\kappa B$



(a) Visualization of the pattern. (b) Enriched IPA subnetwork.

Fig. 7. The union of the three patterns that are all enriched with the $TNF\alpha/NF\kappa B$ signaling pathway. There are ten genes in this combined pattern (all are known cancer related genes), out of which six are included in the $TNF\alpha/NF\kappa B$ signaling pathway (enrichment p-value 1.4024×10^{-5}).

pathway (enrichment p-value 1.4024×10^{-5}). This may suggest that the other four genes may also participate in or interact with this cancer pathway.[31] Figure 7(b) shows the enriched IPA[k] subnetwork, containing 8 of the ten genes (the 8 genes are shaded). Interestingly, IPA also shows that the connecting components (*NFκB complex, ERK, Mapk and Cyclin E*) are also known to be related to cancer.

Specifically among those patterns that are not considered significant by correlation difference (Region *B*s in Figure 4), some are enriched with the target gene sets of cancer-related microRNAs and transcription factors. For example, the first two genes in the pattern (*PIK3C2B,TSC22D1,AKAP12*) are among the set of target genes of miR-101 (p-value 0.001), a small non-coding RNA that regulates gene expression. miR-101 was shown to be down-regulated in cancer.[10] This agrees with the loss of coexpression of its target genes (*PIK3C2B,TSC22D1*) in the cancer class. Furthermore, the miR-101 targets are enriched for several signaling pathways, and the third gene *AKAP12* is a known regular of protein kinase A (PKA), a central signaling pathway involved in cell growth and proliferation. This may lead to the differential coexpression of (*PIK3C2B,TSC22D1*) and *AKAP12* together as a DC pattern. In another example, the first two genes in the pattern (*ETV4,PTHLH,CBX5*) are among the set of genes with promoter regions $[-2kb, 2kb]$ around the transcription start site containing the motif *TTACGTAA* which matches the binding site for the transcription factor ATF2 (p-value 2.5119×10^{-4}). Mutations of ATF2 was shown to be related to cancer,[45] which agrees with the loss of coexpression of its target genes (*ETV4,PTHLH*) in the cancer class. In addition, the ATF2 targets show enrichment for transcription regulation (repression), and *CBX5* is component of heterochromatin, an epigenetic factor in the regulation of gene expression. This may lead to the differential coexpression of (*ETV4,PTHLH*) and *CBX5* together as a DC pattern.

4. Conclusions

In this paper, we studied methods to identify disease-related change of coexpression subnetworks, i.e. differential coexpression analysis. Specifically, we extended differential coexpression analysis to subspace patterns and proposed an approach based upon association analysis framework[1] that allows exhaustive yet efficient discovery of subspace differential coexpression patterns. This approach can be used to adapt a family of biclustering algorithms to obtain their corresponding differential versions that can directly discover differential coexpression patterns. We illustrated the general approach on a recently-developed biclustering algorithm, and presented the results of experiments on lung cancer datasets using this algorithm. The results showed the existence of meaningful subspace differential coexpression patterns in real-life data. Permutation tests demonstrated the statistical significance for a large number of discovered patterns, many of which can not be discovered if they are measured over all the samples in each of the classes. Interestingly, some discovered patterns also have a significant overlap with known cancer pathways, and some are enriched with the target gene sets of a cancer-related microRNA and

[k]Ingenuity Pathway Analysis: http://www.ingenuity.com/

a cancer-related transcription factor. These results suggest that subspace DC patterns may aid in developing new understanding about the mechanisms underlying cancer and other diseases.

5. Limitations and Future Work

In this section, we discuss several limitations of the proposed approach, possible solutions and future work.

(1) **Size of patterns**: Due to the fixed thresholds imposed on $\widetilde{SDC}$ in the Apriori framework, there may be some larger patterns that do not satisfy the thresholds and are split into smaller ones. This limitation of association analysis is usually addressed by pattern summarization,[13] in which smaller size patterns are merged into larger ones under some criteria. For example, the size-10 pattern in Figure 7 is obtained by merging three smaller patterns as described in Section 3.5. More sophisticated summarization approaches[13] can be exploited in future work.

(2) **Enhancing scalability**: The scalability of the approach depends on the mining algorithm, as well as the permutation test. Generally, the algorithm itself takes about ten minutes for 2000 genes, several hours for 4000 genes and more than a day for all the 8787 genes[l], which is acceptable. However, the real challenge comes from the permutation test in which the mining algorithm is called 1000 times, the total time of which is unacceptable on all the 8787 genes. Thus, to have a comprehensive evaluation of the discovered patterns, we limited the pattern discovery and the follow-up statistical and biological analysis to the subset of genes that are known to be related to cancer. In future work, for the efficiency of the mining algorithm, more effective pruning schemes should be studied together with preprocessing procedures such as standard deviation based gene filtering[m]. For the scalability in the context of permutation test, efficiency could possibly be improved by reusing the calculation over the large number of permutations as studied by Zhang et al.[50]

(3) **Modifying other biclustering algorithms**: In this paper, *DiffRange* is presented as an illustration of the general approach, $\widetilde{SDC}$*-Apriori*, for modifying a biclusteirng algorithm to its differential version. As discussed in Section 2.1, $\widetilde{SDC}$*-Apriori* can also be applied to modify other biclustering algorithms[24,43,49,51] with the antimonotonicity property, and their corresponding differential versions are expected to complement *DiffRange* for discovering differential coexpression patterns.

(4) **Differential biclustering**: Differential coexpression patterns can essentially be considered as biclusters that exist mostly in one class but not in the other. Indeed, such type of biclusters have already been observed in several studies,[26,38,51] where a set of biclusters are discovered in the first step and then the ones that are unique to a single class are selected in the second step. Such a two-step approach can also be used to discover differential coexpression patterns. However, the general approach proposed in this paper, $\widetilde{SDC}$*-Apriori*, can be considered as an initial effort towards a more general *differential biclustering* problem, where more efficient discovery of differential biclusters are possible by making use of class labels within the biclustering process. Similar problems can also be formulated as differential/discriminative co-clustering and differential/discriminative subspace clustering in the data mining community.

(5) **Pattern-based classification**: Since a subspace differential coexpression pattern explicitly captures the subgroups of samples it covers, it will also be interesting to investigate the predictive power of subspace differential coexpression patterns in a pattern-based classification framework,[4,42] where the combination of traditional differentially expressed genes and subspace differential coexpression patterns may provide more accurate disease diagnosis.

Acknowledgments

The authors would like to thank Ba Ryun Hwang for dataset preprocessing, and thank the anonymous reviewers for the constructive comments. This work was supported by NSF grants #CRI-0551551, #IIS-0308264, and #ITR-0325949. Access to computing facilities was provided by the Minnesota Supercomputing Institute.

[l]The experiments presented here were run on a Linux machine with Intel(R) Xeon(R) CPU (E5310 @ 1.60GHz) and 16GB memory
[m]A gene with small variation across samples is less likely to constitute a differential coexpression pattern.

References

1. R. Agrawal and R. Srikant. In *Proc. Very Large Data Bases*, pages 487–499, 1994.
2. M. Benito, J. Parker, Q. Du, J. Wu, D. Xiang, C. Perou, and J. Marron. *Bioinformatics*, 20(1):105–114, 2004.
3. A. Bhattacharjee, W. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, et al. *PNAS*, 21(15):3301–3307, 2005.
4. H. Cheng, X. Yan, J. Han, and C.-W. Hsu. In *Proceedings. Intl Conf. Data Engineering*, pages 716–725, 2007.
5. Y. Cheng and G. Church. In *Proceedings of ISMB*, pages 8:93–103, 2000.
6. S. Cho, J. Kim, and J. Kim. *BMC Bioinformatics*, 10(1):109, 2009.
7. J. Choi, U. Yu, O. Yoo, and S. Kim. *Bioinformatics*, 21(24):4348–4355, 2005.
8. X. Cui and G. Churchill. *Genome Biology*, 4(4):210, 2003.
9. G. Fang, G. Pandey, M. Gupta, M. Steinbach, and V. Kumar. Tech Report 09-011, Department of Computer Science, University of Minnesota, 2009.
10. J. Friedman, G. Liang, C. Liu, E. Wolff, Y. Tsai, W. Ye, X. Zhou, and P. Jones. *Cancer Research*, 69(6):2623, 2009.
11. T. Fuller, A. Ghazalpour, J. Aten, T. Drake, A. Lusis, and S. Horvath. *Mammalian Genome*, 18(6):463–472, 2007.
12. P. Gargalovic, M. Imura, B. Zhang, N. Gharavi, M. Clark, J. Pagnon, W. Yang, A. He, A. Truong, S. Patel, et al. *PNAS*, 103(34):12741, 2006.
13. J. Han, H. Cheng, D. Xin, and X. Yan. *Data Mining and Knowledge Discovery*, 15:55–86, 2007.
14. D. Hanahan and R. Weinberg. *Cell*, 100(1):57–70, 2000.
15. M. E. Higgins, M. Claremont, J. E. Major, C. Sander, and A. E. Lash. *Nucl. Acids Res.*, 35(suppl 1):D721–726, 2007.
16. N. Hudson, A. Reverter, and B. Dalrymple. *PLoS Computational Biology*, 5(5), 2009.
17. J. Ihmels, S. Bergmann, J. Berman, N. Barkai, and L. Kruglyak. *PLoS Genetics*, 1(3):e39, 2005.
18. R. Irizarry, B. Hobbs, F. Collin, Y. Beazer-Barclay, K. Antonellis, et al. *Biostatistics*, 4(2):249–264, 2003.
19. D. Kim, S. Koo, K. Jeon, M. Kim, J. Lee, C. Hong, and S. Jeong. *Cancer Research*, 63:621-626, 2003.
20. D. Kostka and R. Spang. *Bioinformatics*, 20(1):194–199, 2004.
21. Y. Lai, B. Wu, L. Chen, and H. Zhao. *Bioinformatics*, 20(17):3146–3155, 2004.
22. K. Li. *PNAS*, 99(26):16875–16880, 2002.
23. K. Li, C. Liu, W. Sun, S. Yuan, and T. Yu. *PNAS*, 101(44):15561–15566, 2004.
24. G. Liu, J. Li, K. Sim, and L. Wong. In *Proc. Intl Conf. Data Engineering*, 1250–1254, 2007.
25. S. Madeira and A. Oliveira. *IEEE/ACM Trans on Compu Bio and Bioinfo*, 1(1):24–45, 2004.
26. T. Murali and S. Kasif. In *Proc. Pacific Symposium on Biocomputing 8:77-88*, 2003.
27. M. Oldham, S. Horvath, and D. Geschwind. *PNAS*, 103(47):17973, 2006.
28. G. Pandey, G. Atluri, M. Steinbach, C. L. Myers, and V. Kumar. In *Proc. ACM Conf. on Kownlege Discovery and Data Mining*, pages 677-686, 2009.
29. C. Prieto, M. Rivas, J. Sanchez, J. Lopez-Fidalgo, and J. De Las Rivas. *Bioinformatics*, 22(9):1103–1110, 2006.
30. J. Quackenbush. *The New England journal of medicine*, 354(23):2463, 2006.
31. B. Rosemary, C. Leslie, and P. Giovanni. *BMC Bioinformatics*, 9:488, 2008.
32. A. Shabalin, H. Tjelmeland, C. Fan, C. Perou, and A. Nobel. *Bioinformatics*, 24(9):1154, 2008.
33. J. Shaffer. *Annual Review of Psychology*, 46(1):561–584, 1995.
34. C. Silva, M. Silva, L. Faccioli, R. Pietro, S. Cortez, et al. *Clinical and Experimental Immunology*, 101(2):314, 1995.
35. R. Stearman, L. Dwyer-Nield, L. Zerbe, S. Blaine, Z. Chan, P. Bunn, G. Johnson, F. Hirsch, D. Merrick, W. Franklin, et al. *Am J Pathol.*, 167(6):1763–75, 2005.
36. L. Su, C. Chang, Y. Wu, K. Chen, C. Lin, et al. *BMC Genomics*, 8(1):140, 2007.
37. A. Subramanian, P. Tamayo, V. Mootha, S. Mukherjee, B. Ebert, M. Gillette, A. Paulovich, S. Pomeroy, T. Golub, E. Lander, et al. *PNAS*, 102(43):15545–15550, 2005.
38. A. Tanay, R. Sharan, M. Kupiec, and R. Shamir. *PNAS*, 101(9):2981–2986, 2004.
39. S. Tomlins, D. Rhodes, S. Perner, S. Dhanasekaran, R. Mehra, X. Sun, S. Varambally, X. Cao, J. Tchinda, R. Kuefer, et al. *Science*, 310(5748):644–648, 2005.
40. K. Uematsu, B. He, L. You, Z. Xu, F. McCormick, and D. Jablons. *Oncogene*, 22(46):7218–7221, 2003.
41. I. Ulitsky, R. Karp, and R. Shamir. In *Proc. RECOMB*, 4955:347, 2008.
42. M. van Vliet, C. Klijn, L. Wessels, and M. Reinders. *PLoS ONE*, 2(10):1047, 2007.
43. H. Wang, W. Wang, J. Yang, and P. Yu. In *Proc. ACM Conf. on Management of Data*, pages 394–405. 2002.
44. M. Watson. *BMC Bioinformatics*, 7(1):509, 2006.
45. I. Woo, T. Kohno, K. Inoue, S. Ishii, and J. Yokota. *International Journal of Oncology*, 20(3):527–531, 2002.
46. B. Wu. *Biostatistics*, 8(3):566, 2007.
47. M. Xu, M. Kao, J. Nunez-Iglesias, J. Nevins, M. West, and X. Zhou. *BMC genomics*, 9(Suppl 1):S12, 2008.
48. B. Zhang and S. Horvath. *Stat Appl in Genet and Mol Bio*, 4(1):1128, 2005.
49. X. Zhang, F. Pan, and W. Wang. In *Proc. Intl Conf. Data Engineering*, 130-139, 2008.
50. X. Zhang, F. Pan, Y. Xie, F. Zou, and W. Wang. In *Proceeding of RECOMB*, volume 5541, pages 253–269, 2009.
51. L. Zhao and M. Zaki. *IEEE Intelligent Systems*, 20(6):40–49, 2005.

ESTIMATION OF PROTEIN AND DOMAIN INTERACTIONS IN THE SWITCHING MOTILITY SYSTEM OF *MYXOCOCCUS XANTHUS*

F. MORCOS[1], M. SIKORA[2], M. ALBER[3], D. KAISER[4] and J. A. IZAGUIRRE[1]

[1]*Department of Computer Science & Engineering,*
[2]*Department of Electrical Engineering,* [3]*Department of Mathematics,*
University of Notre Dame, Notre Dame, IN 46556, USA
E-mail: amorcosg@nd.edu, msikora@ieee.org, malber@nd.edu, izaguirr@nd.edu

[4]*Departments of Biochemistry and Developmental Biology,*
Stanford University, Stanford, CA
E-mail: adkaiser@stanford.edu

The gram-negative myxobacterium *Myxococcus xanthus* is equipped with an interesting motility system that allows it to reverse direction on average every 8 minutes by switching the construction of two motility engines at the ends of this rod-shaped bacterium. While the mechanisms responsible for timing and engine construction/deconstruction are relatively well understood, there are several competing hypotheses as to how they are coupled together. In this paper we examine the evidence for protein interactions underlying these possible couplings using a novel framework consisting of a probabilistic model describing protein and domain interactions and a belief propagation inference algorithm. When provided with large amount of indirect pieces of information, such as high-throughput experiment results, and protein structures, we can reliably determine the relative likelihoods of these hypotheses, even though each individual piece of evidence by itself has very limited reliability. The same framework can be used to map large protein and domain interaction networks in myxobacteria and other organisms.

Keywords: PPI; DDI; M. xanthus; motility reversals; inference; Sum-Product Algorithm

1. Introduction

The availability of large quantities of experimental data, protein information, crystal structures of proteins and domains stimulated research on quantitative methods to estimate protein-protein (PPI) and domain-domain (DDI) interaction likelihoods. At the present time, most of these efforts have been focused on proving the effectiveness of these algorithms on known protein datasets. However, fewer efforts have been put into using such predictive tools to understand and study the dynamics of protein networks of actual interest in biology from an *in silico* perspective. In this work, we aim to do this by concentrating in a particular pathway related to motility in myxobacteria. Specifically, our study focuses on the study of *Myxococcus xanthus*, a gram-negative myxobacterium which under extreme conditions of starvation stops swarming and starts a new developmental phase to coordinate their motion cooperatively to aggregate and ultimately form fruiting bodies.[1] This multicellular cooperation in an unicellular organism makes *Myxococcus xanthus* an important model to understand regulatory pathways where many cells work together to achieve a common functional role.

To achieve this we use a PPI/DDI inference framework to provide a set of possible interactions underlying signaling pathways in *M. xanthus*. *M. xanthus* cells reverse the direction of gliding, which leads to efficient swarming, by controlling the assembly and disassembly of motility engines. For these reasons, *M. xanthus* is an interesting system to perform PPI predictions, since many of its proteins and domains have homology with chemosensory and other regulatory proteins in other well studied proteobacteria like *Escherichia coli*, *Salmonella typhimurium* and other sequenced δ-proteobacteria. Structures of domains in *E. coli* proteins are available, a feature required for the structural scoring. We also include structural complexes of domains obtained from the iPfam database to enhance the predictive nature of our methodology.

2. Methods

2.1. *Sum-Product Algorithm for DDI/PPI Inference*

Posterior probabilities for domain and protein interaction pairs can be obtained by calculating the marginals of a joint probability distribution that is a function of protein interaction experiments, domain interaction evidence and the relationship between domain composition of proteins and the fact that individual independent domains establish physical contacts leading to protein interactions. We calculate this using the Sum-Product Algorithm (SPA),[2] an efficient method to compute marginals of multivariate functions that factor into products of simpler functions. The SPA uses a graph representation of this joint probability distribution, called the *factor graph*, and obtains marginal values by iteratively exchanging messages along the graph edges. One of the advantages of SPA is the ability to improve prediction accuracy against other known methods like maximum likelihood estimation, while at the same time correct experimental errors. SPA will not only estimate the probability of potential interaction but also will re-score the experimental input data and correct putative experimental errors. We present a more in-depth derivation and validation of the use of SPA for PPI and DDI inference in.[3] In this work, we include a brief overview of the method and focus on its application to protein networks in *M. xanthus*.

We denote by $A_{i,j} = 1$ a hypothesis that proteins i and j interact, by $B_{x,y} = 1$ a hypothesis that domains x and y interact, and use $M_{i,j}$ and $N_{x,y}$ to quantify the results of interaction measurements or experimental evidence performed on the protein pair (i, j) and domain pair (x, y), respectively. According to the model, the joint probability distribution $P(\mathbf{A}, \mathbf{B}, \mathbf{M}, \mathbf{N}|\mathbf{H})$, where $\mathbf{H}$ denotes the set of the domain architectures of proteins, factors into a product

$$P(\mathbf{A}, \mathbf{B}, \mathbf{M}, \mathbf{N}|\mathbf{H}) = \prod_{(i,j)} P(M_{i,j}|A_{i,j}) \prod_{(x,y)} P(N_{x,y}|B_{x,y}) P(B_{x,y}) P(\mathbf{A}|\mathbf{B}, \mathbf{H}). \quad (1)$$

The terms $P(M_{i,j}|A_{i,j})$ and $P(N_{x,y}|B_{x,y})$ represent the probability that an experiment produces a positive result given that proteins i and j actually interact and the probability of a positive measurement given that domain x interacts with y, respectively. These terms serve as the main information input points of the model. The term $P(B_{x,y})$ represents the a priori probability of a domain-domain interactions and is set for all x and y to an estimated probability that two randomly selected domains would interact. The central part of our model is described through deterministic relations of the form

$$A_{i,j} = \bigvee_{(x,y) \in \mathcal{B}_{i,j}(\mathbf{H})} B_{x,y} \text{ for all } (i, j), \quad (2)$$

where $\mathcal{B}_{i,j}(\mathbf{H})$ is the set of domain pairs, such that one domain is present in protein i and the other in j. Relations (2) state that a protein pair interacts if and only if at least one of its domain pairs interacts. This set of equations must be satisfied by all $A_{i,j}$ and $B_{x,y}$. Consequently, the probability $P(\mathbf{A}|\mathbf{B}, \mathbf{H})$ can be factored into a product of individual indicator functions for each (i, j). The factor graph illustrating the

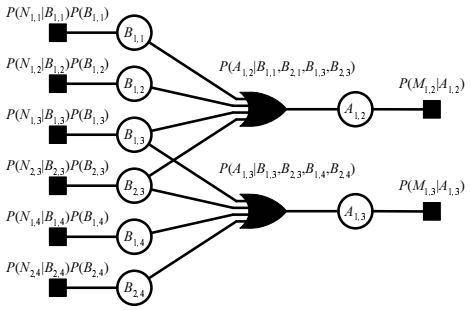


Fig. 1. **Factor graph representation of the joint probability distribution** $P(\mathbf{A}, \mathbf{B}, \mathbf{M}, \mathbf{N}|\mathbf{H})$.

complete factorization is presented in Figure 1. The SPA iteratively recomputes messages along each edge in

the graph according to equations specific to the type of the node from which the message originates. After completing a predefined number of iterations, the algorithm obtains the *a posteriori* probabilities of PPI, $P(A_{i,j} = 1|\mathbf{M}, \mathbf{N}, \mathbf{H})$, and DDI, $P(B_{x,y} = 1|\mathbf{M}, \mathbf{N}, \mathbf{H})$, which are the final result of the inference task. They are obtained from all messages incoming into their corresponding variable nodes, i.e.,

$$P(A_{i,j} = 1|\mathbf{M}, \mathbf{N}, \mathbf{H}) = \left(1 + \prod_{k=1}^{K^{A_{i,j}}} e^{\alpha_k^{A_{i,j}}}\right)^{-1}, \tag{3}$$

$$P(B_{x,y} = 1|\mathbf{M}, \mathbf{N}, \mathbf{H}) = \left(1 + \prod_{k=1}^{K^{B_{x,y}}} e^{\alpha_k^{B_{x,y}}}\right)^{-1}, \tag{4}$$

generated in the final iteration. Where $\alpha_k^{A_{i,j}}$ is the log-ratio message entering node $A_{i,j}$ along the branch k in the factor graph of Figure 1. In this model, the domain pair architecture as well as potential protein pairs are connected via a factor graph. Hence, protein and domain pair probability estimates depend on the shared domain pair architecture. Since protein domain architectures define the topology of the factor graphs they directly affect the computation of probabilities via message passing.

2.2. *Data Sets*

Several sources were used to extract relevant information for PPI inference. Protein information was retrieved from Uniprot[4] and 3-D protein structures from the Protein Data Bank.[5] Domain composition of each protein was extracted from Pfam 22.[6] PPI data were obtained from two main sources: Database of Interacting Protein (DIP),[7] where most of the interaction pairs were obtained using yeast two-hybrid assays and IntAct,[8] which contains a higher number of interactions with different reliability levels. In[3] we use a high quality binary dataset of PPI in *Saccharomyces cerevisiae*, compiled by Yu et al.[9] for performance evaluation and validation. The algorithm also uses 6,081 iPfam domain interaction pairs as input.[10]

Table 1. Data sets used by our inference *in silico* methodology

Data Source	**Description**
Uniprot	Protein general information and sequence
DIP Database and literature (May 2007)	**33,234** protein interactions
IntAct	**150,876** protein interactions
Yeast Golden Set (Yu et al.)	**2,581** protein interactions
Pfam version 22	Domain composition of proteins
iPfam	**6,081** Domain-domain interactions with complex
PDB	Protein and Domain three dimensional structure

For the *M. xanthus* system we used a list of 23 proteins that are known to be related to the Frz system. Out of 253 possible pairings, SPA found a list of 66 potential PPI with a probability larger than 0. These interactions contained 18 unique proteins. We discarded protein interactions that do not share the same cellular location to obtain Figure 2. Finally, Table 2 shows 10 protein pairs that either have experimentally confirmed interactions or that contain potential interactions that seem biologically meaningful for the motility reversal model in *M. xanthus*. The discarded protein interactions have a mean score of 0.6634 while the 10 selected interactions have a mean score of 0.8064.

2.3. *SPA run time and scalability*

Run time of this algorithm depends on the size of the factor graph and the number of iterations defined in the message passing algorithm. The factor graph scales quadratically with the number of proteins and domains

analyzed. However, given the nature of the input data, where domains are not shared among all the proteins and the fact that we only have measurements for a smaller subset of protein pairs, a series of disconnected factor graphs are created for the input protein and domain datasets. We process these disconnected factor graphs independently, which allows parallelization and scalability of the inference task. We do predictions using the high performance computing cluster (HPCC) of the University of Notre Dame. Using 2.29 GHz Quad-Core AMD Opteron processors we run SPA with 19 iterations in less than two hours including the largest connected subgraph.

3. Predicting interactors in *M. xanthus* reversal system

We studied how the control of reversals in gliding direction of *M. xanthus* takes place. *M. xanthus* has two different multi-protein engines: S-motility and A-motility; both involve social interactions. S-motility depends on Type IV Pili motors.[11] These pili attach to fibrils surrounding other *M. xanthus* cells and by means of retraction pull the cell forward. A-motility, is associated with slime secretion. One hypothesis states that slime coming out of several molecular nozzles at polar ends of the cell pushes the cell forward[12].[13] Pili should be present in the leading part of the cell while slime nozzles be part of the back of the cell. This configuration allows for pilus retraction and slime propulsion at the same time. A second hypothesis relates A-motility with non-polar cell surface adhesion complexes.[14] *M. xanthus* cells are able to move over surfaces by gliding, they reverse their direction of movement with an average period of approximately 7.2 minutes.[15] This period seems to be optimal to achieve higher swarming rates and cell flux.[16] These reversals occur by switching motility engines from one end of the cell to the other. Under the slime propulsion model, a process of nozzle inactivation takes place when secretion is switched from one pole to the other, and pili are inactivated when they switch from one end to the other. A control circuit is needed for engine switching that would lead to cell reversals. That circuit includes the Frz (frizzy) system which is deemed responsible of controlling the frequency of reversals in *M. xanthus*.[17] This circuit is based on a two-component system that clocks the time to exchanging motility engines in the ends of the cell. Frizzy proteins that control the reversal frequency include FrzCD (uniprot accession: P43500), a methyl-accepting chemosensory protein having similarities with MCP proteins used for flagellar swarming in *E. coli*, but which is cytoplasmic in *M. xanthus* and not a membrane protein as in *E. coli*.[15] Also important is FrzE (P18769), a histidine kinase composed of two domains similar to CheY (P0AE68) and CheA (P07363) proteins in *E. coli*.[18] FrzE has autophosphorylating function induced by a phosphate transfer from the HK domain (CheA) to its response regulator (CheY). It is suggested that FrzE∼P is responsible for signaling reversal of polarity to the motility engine. FrzG (P31758) is a methyltransferase similar to CheB (P07330), which is involved in the methylation of FrzCD, for adaptation. During fruiting body development, the FruA (Q1D7Q3) response regulator triggers FrzCD methylation. During growth there is no FruA, and FrzCD is spontaneously methylated at a low rate.[19]

Igoshin et al.[20] constructed a dynamical model of the Frz system. During reversals the levels of methylated FrzCD and FrzE∼P oscillated consistently out of phase, suggesting that these proteins form a negative feedback loop. The loop describes accurately the frequency of reversals observed in *M. xanthus* when nutrients are scarce and development towards fruiting body formation is in its initial phase. For the model of Igoshin et al. to be valid, it requires one of two potential protein pairs. These hypotheses still remain to be tested experimentally. The two potential interactions are the following: FrzE-FrzF and FrzE-FrzG. To test these hypotheses, we used our predictive methodology on a list of proteins involved in the FruA-Frz network. Since several interactions forming part of the reversal mechanism are understood and their existence has been established, we have a way to assess our predictions. Reproducing interactions that are already studied provides support to our methodology and at the same time shows the potential of *in silico* methodologies to reproduce scientific efforts to elucidate protein interactions in a given pathway. On the other hand, predicting protein interactions in proteins of and related to the Frz-Mgl network let us investigate the plausibility of these competing hypotheses. The prediction algorithm was run using the data sets presented in Methods. Figure 2 presents a PPI network that was the outcome of the algorithm.

In Figure 2, nodes represent signaling proteins while the edge represent the physical interaction pre-

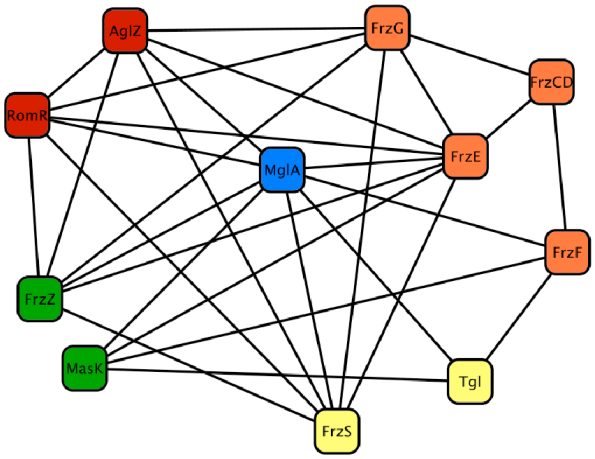


Fig. 2. **Predicted network of interactions in the Frz system of *M. xanthus*.**

diction. Self interactions have been subtracted to facilitate readability. This network has also a color code classification: orange nodes represent proteins in the negative feedback loop of Igoshin's model; red proteins are related to A-motility; yellow nodes play a role in S-motility; green nodes are being identified to have a role in both A and S motility systems and finally MglA is represented by a blue node. This node plays an important role in the connection of the reversal control and the actual motility mechanisms.

Table 2 presents a summary of the most important predicted interactions in the signaling network of *M. xanthus*. It includes literature references and SPA scores to those interactions discuss here, illustrating the correct predictive capabilities of our methodology. The bottom part of the table encompasses those PPI for which we have not found literature describing them, however, our predictions present computational evidence about their interaction and our analysis shows that their biological role might be of importance and thus can be further studied.

Table 2. Predicted Interactions found in Literature and novel interactions for *M. xanthus*

Predicted Interaction	Reference	SPA score
FrzE - FrzG	[20] Hypothesis (high likelihood)	0.96
FrzE - FrzCD	[21]	0.99
FrzE - FrzZ	[22] (suggested)	1.00
FrzF - FrzCD	[21]	0.46
MglA - MasK (Q1DB00)	[23]	1.00
MglA - AglZ	[24]	0.56
MglA - FrzE	Predicted , (motility bridge)	0.639
MglA - FrzF	Predicted	0.95
MglA - RomR	Predicted (A-motility)	0.56
MglA - Tgl	Predicted (S-motility)	0.95

3.1. *Analysis of hypotheses in Igoshin's model*

We turned our discussion to those potential interactions that make possible a negative feedback loop yielding a biochemical oscillator in the signaling circuit of *M. xanthus*. Based on these results, we provide computational support for one of these hypothesis. The hypothesis FrzE-FrzG, was predicted with high score (see

Table 2). The PDB 1a2o provides structural evidence that proteins with these domains might create interacting complexes. Djordjevicj et al.[25] determined the crystal structure of CheB. It has two domains, a response regulator and a CheB methylesterase joined by a linker. They found that in its unphosphorylated state, the RR would bind the CheB_methylest domain and impede its activity. The crystal in 1a2o shows how an unphosphorylated response regulator interacts with the effector domain CheB_methylest with a surface area of 1000 $Å^2$. They also argued that phosphorylation of the response regulator would induce a conformational change that takes both domains apart and free the methylesterase to perform its catalytic function. Phosphorylation induces minor reorientation of regulator helices $\alpha 4$ and $\alpha 5$, resulting in disruption of the inter-domain interaction. Protein CheA phosphorylates the response regulator in CheB.[26] In the regulatory circuit of *M. xanthus*, the FrzG-FrzE interaction is different because FrzG only has the CheB_methylest domain, hence this domain could be active in the absence of a regulator. We hypothesize that the unphosphorylated response regulator in FrzE could help inhibit FrzG function, while an autophosphorylated FrzE could not block FrzG methylesterase and activating it as a consequence. This hypothesis is plausible but unconfirmed. Our results suggest that the biochemical oscillator depends on the FrzE-FrzG interaction.

3.2. *Searching for interactions with MglA, the proposed switch protein*

Although the mechanism to control the frequency of reversals in *M. xanthus* has been studied extensively through the Frz system, and other two-component systems,[27] there are still some missing links needed to uncover the connection between the MglA switch and the actual A and S engines. MglA has been shown experimentally to have an influence in both A and S engines.[28] Cells with mutant *mglA* genes fail to reverse and fail to swarm. They show simultaneous secretion of slime from both ends of the cell;[13] it is suggested, but not experimentally confirmed, that pili are also present from both ends of the cell. Thus MglA may serve as a switch to start disassembly of engines at both ends of the cell.[16] MglA-GTP might pick the end that should lose pili by losing Tgl (P95324) and would also select the end to inactivate slime propulsion by potentially interacting with CglB (O31191).[29]

We included MglA, with a conserved Ras domain, in our list of interactors. We were interested in the interaction patterns of MglA with respect to proteins in the Frz system and other proteins related to motility. We found that MglA has a potential central role in both motility systems and the Frz system given its predicted interactions with different members of these systems. Figure 2 depicts MglA in blue, and presents edges with several proteins. We start our discussion by analyzing the inferred interaction MglA-FrzE, since this interaction could potentially represent the link between the Frz system and the two motility systems in *M. xanthus*.

Our estimation algorithm produced a score for Ras-Response_reg, supporting the plausibility that a potential interaction between MglA and FrzE results from a physical contact between the Ras domain in MglA and the phosphorylated Response_reg in FrzE. The domain pair Ras-Response_reg-P which underlies the FrzE-MglA interaction leads to the prediction of other interactions that seem important to our study. The protein pairs MglA-FrzZ (Q7BU54) and MglA-FrzS (Q1D4U9) are also plausible interactions between a Ras domain pair and response regulators. These results suggest how MglA has an important role for the switching of motility in the Frz system of *M. xanthus*.

3.3. *MglA interacts with Tgl and RomR*

We also investigated a possible interaction between MglA and engine proteins. Tgl is a lipoprotein needed for the assembly of the PilQ (Q9ZFG1) secretin and thus for S-motility. Tgl is a stimulatable protein found in membrane. It allows PilQ to assemble with the help of six TPR (Tetratrico peptide repeat superfamily) repeats. One hypothesis is that three of these repeats interact with one monomer of PilQ, while the three remaining repeats interact with the adjacent PilQ monomer.[29] We investigated if MglA interacts with Tgl. Our prediction algorithm infered that TPR_1, one of the domains present in Tgl, interacts with high probability with the Ras domain in MglA. This domain interaction would potentially trigger the destruction of

Tgl or its removal from PilQ.[29] The inclusion of iPfam evidence in our algorithm allowed us to identify a crystal structure (PDB: `1e96`) showing the Ras domain in complex with TPR_1. When this structure was obtained, Rittinger et al. suggested that this interaction is important for the assembly of protein (enzyme) complexes.[30] This proposition provides more plausibility to the hypothesis that MglA-GTP causes PilQ to disassemble. It is possible that through this interaction, MglA opens Tgl in such a way that it makes it accessible to a protease that would then contribute to the disassemble of PilQ at one end of the rod shaped cell.[31]

RomR is an essential protein for slime engine function. Its switching from the extremes of the cell depends on MglA. Thus, we are interested in the possible interaction MglA-RomR. Søgaard-Andersen et al.[32] report that RomR has a Response_reg (receiver) domain in the N terminal and an output domain in the C terminal of the protein. Søgaard-Andersen et al. proposed that correct RomR polarity depends on the small GTPase MglA. They also state the importance of RomR for A-motility and how this protein relocates synchronously with another Frz protein: FrzS. They concluded that the Receiver domain is involved in dynamic RomR localization and that this is required for reversals, while the output domain is a polar targeting determinant. Their studies also showed that when the receiver domain is not phosphorylated then no reversals are observed. On the other hand, if the receiver domain is always phosphorylated then there is a 1.5 fold increase in reversal frequency. However, the RomR kinase has yet to be identified. With this information and based on our prediction of interaction between Ras-Response_reg-P, we provide computational evidence that indeed MglA could interact with RomR.

4. Discussion

In this work, PPI/DDI prediction methods are combined with the biological understanding of the reversal system of *M. xanthus* as an attempt to improve present knowledge of the players involved in this process. A contribution of this work is to illustrate how this framework is used to study small networks that perform important functions in an organism. The focus in *M. xanthus*' reversal system is due to its importance in development and swarming. Support for this approach is provided by predicting interactions previously reported in the literature. This study provides an assessment of the contribution of interactions FrzE-FrzF and FrzE-FrzG as being part of the biochemical oscillator that controls the reversal frequency. Results suggest that FrzE-FrzG plays an important role in negative feedback circuit. Furthermore, this framework let us reach a more detailed explanation of how the response regulator domain in FrzE possibly interacts with the CheB methylesterase domain in FrzG. This interaction possibly inhibits FrzG demethylation activity. When the response regulator is phosphorylated this interaction is broken allowing FrzG to demethylate FrzCD. This mechanism is needed to support the negative feedback model in the "Frizzilator".

We investigated the role of MglA as a switch to control the construction of motility engines in *M. xanthus*. Predictions showed the central role of this protein, first as a bridge with the negative feedback system by interacting with FrzE and how its interactions with RomR, Tgl, FrzS and AglZ (Q1D823) are important for both A and S engines. With these predictions, along with the present understanding of the reversal switch it is possible to expand the model of this system. This model is illustrated in Figure 3.

The model shows how MglA serves as a link between the signaling pathway involving Frz proteins and the motility engines. MglA receives a signaling message from FrzE that triggers the destruction of old engines in the ends of the cell. Potential interactions of MglA with Tgl and FrzS have an impact in the switching of S-motility engines. Parallel to these interactions, MglA is predicted to be interacting with RomR and AglZ, showing a connection with the A-motility engines. This expanded model could help devise testable hypotheses in the motility system of *M. xanthus*.

Acknowledgments

This work was supported in part by NSF grant CCF-0622940.

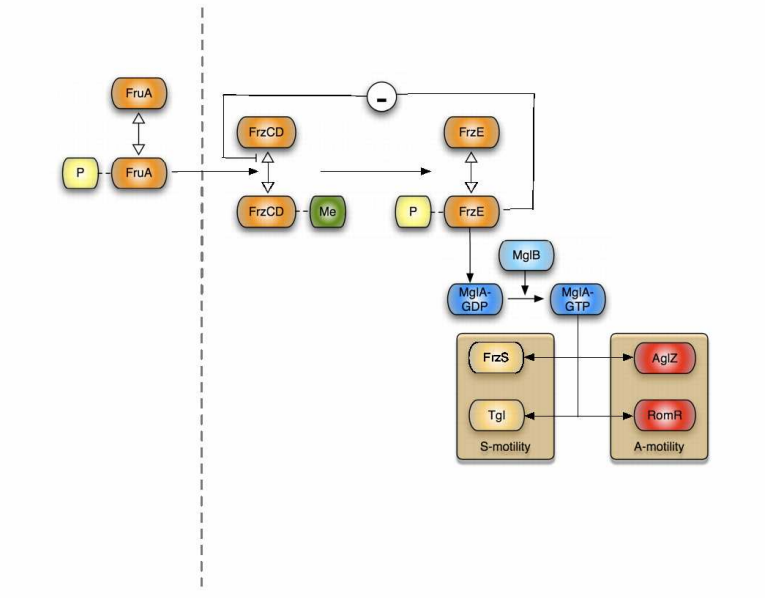


Fig. 3. **Expanded model of switching motility in *M. xanthus***. Switching is triggered by FrzE-MglA. MglA-GTP interacts with A and S motility systems.

References

1. Kaiser, D. *Annu Rev Microbiol* **58**, 75–98 (2004).
2. Kschischang, F., Frey, B., and Loeliger, H.-A. *Information Theory, IEEE Transactions on* **47**(2), 498–519 Feb (2001).
3. Morcos, F., Sikora, M., Alber, M., Kaiser, D., and Izaguirre, J. A. *Information Theory, IEEE Transactions on* **(under review)** (2009).
4. Consortium, U. *Nucleic Acids Res* **35**(Database issue), D193–D197 Jan (2007).
5. Deshpande and et al. *Nucleic Acids Res* **33**(Database issue), D233–D237 Jan (2005).
6. Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L. L., Studholme, D. J., Yeats, C., and Eddy, S. R. *Nucleic Acids Res* **32**(Database issue), D138–D141 Jan (2004).
7. Xenarios, I., Salwnski, L., Duan, X. J., Higney, P., Kim, S.-M., and Eisenberg, D. *Nucleic Acids Res* **30**(1), 303–305 Jan (2002).
8. Kerrien, S., Alam-Faruque, Y., Aranda, B., Bancarz, I., Bridge, A., Derow, C., Dimmer, E., Feuermann, M., Friedrichsen, A., Huntley, R., Kohler, C., Khadake, J., Leroy, C., Liban, A., Lieftink, C., Montecchi-Palazzi, L., Orchard, S., Risse, J., Robbe, K., Roechert, B., Thorneycroft, D., Zhang, Y., Apweiler, R., and Hermjakob, H. *Nucleic Acids Res* **35**(Database issue), D561–D565 Jan (2007).
9. Yu, H., Braun, P., Yildirim, M. A., Lemmens, I., Venkatesan, K., Sahalie, J., Hirozane-Kishikawa, T., Gebreab, F., Li, N., Simonis, N., Hao, T., Rual, J.-F., Dricot, A., Vazquez, A., Murray, R. R., Simon, C., Tardivo, L., Tam, S., Svrzikapa, N., Fan, C., de Smet, A.-S., Motyl, A., Hudson, M. E., Park, J., Xin, X., Cusick, M. E., Moore, T., Boone, C., Snyder, M., Roth, F. P., Barabsi, A.-L., Tavernier, J., Hill, D. E., and Vidal, M. *Science* **322**(5898), 104–110 Oct (2008).
10. Finn, R. D., Marshall, M., and Bateman, A. *Bioinformatics* **21**(3), 410–412 Feb (2005).
11. Spormann, A. M. *Microbiol Mol Biol Rev* **63**(3), 621–641 Sep (1999).
12. Wolgemuth, C., Hoiczyk, E., Kaiser, D., and Oster, G. *Curr Biol* **12**(5), 369–377 Mar (2002).
13. Yu, R. and Kaiser, D. *Mol Microbiol* **63**(2), 454–467 Jan (2007).
14. Sliusarenko, O., Zusman, D. R., and Oster, G. *J Bacteriol* **189**(21), 7920–7921 Nov (2007).
15. Bustamante, V. H., Martnez-Flores, I., Vlamakis, H. C., and Zusman, D. R. *Mol Microbiol* **53**(5), 1501–1513 Sep (2004).
16. Wu, Y., Kaiser, A. D., Jiang, Y., and Alber, M. S. *Proc Natl Acad Sci U S A* **106**(4), 1222–1227 Jan (2009).
17. Zusman, D. R. *J Bacteriol* **150**(3), 1430–1437 Jun (1982).
18. McCleary, W. R. and Zusman, D. R. *J Bacteriol* **172**(12), 6661–6668 Dec (1990).
19. McCleary, W. R., McBride, M. J., and Zusman, D. R. *J Bacteriol* **172**(9), 4877–4887 Sep (1990).
20. Igoshin, O. A., Goldbeter, A., Kaiser, D., and Oster, G. *Proc Natl Acad Sci U S A* **101**(44), 15760–15765 Nov

(2004).
21. McBride, M. J., Khler, T., and Zusman, D. R. *J Bacteriol* **174**(13), 4246–4257 Jul (1992).
22. Li, Y., Bustamante, V. H., Lux, R., Zusman, D., and Shi, W. *J Bacteriol* **187**(5), 1716–1723 Mar (2005).
23. Thomasson, B., Link, J., Stassinopoulos, A. G., Burke, N., Plamann, L., and Hartzell, P. L. *Mol Microbiol* **46**(5), 1399–1413 Dec (2002).
24. Yang, R., Bartle, S., Otto, R., Stassinopoulos, A., Rogers, M., Plamann, L., and Hartzell, P. *J Bacteriol* **186**(18), 6168–6178 Sep (2004).
25. Djordjevic, S., Goudreau, P. N., Xu, Q., Stock, A. M., and West, A. H. *Proc Natl Acad Sci U S A* **95**(4), 1381–1386 Feb (1998).
26. Hess, J. F., Oosawa, K., Kaplan, N., and Simon, M. I. *Cell* **53**(1), 79–87 Apr (1988).
27. Stock, A. M., Robinson, V. L., and Goudreau, P. N. *Annu Rev Biochem* **69**, 183–215 (2000).
28. Stephens, K., Hartzell, P., and Kaiser, D. *J Bacteriol* **171**(2), 819–830 Feb (1989).
29. Nudleman, E., Wall, D., and Kaiser, D. *Mol Microbiol* **60**(1), 16–29 Apr (2006).
30. Lapouge, K., Smith, S. J., Walker, P. A., Gamblin, S. J., Smerdon, S. J., and Rittinger, K. *Mol Cell* **6**(4), 899–907 Oct (2000).
31. Rodriguez-Soto, J. P. and Kaiser, D. *J Bacteriol* **179**(13), 4372–4381 Jul (1997).
32. Leonardy, S., Freymark, G., Hebener, S., Ellehauge, E., and Sgaard-Andersen, L. *EMBO J* Oct (2007).

EXPLORING BIOLOGICAL NETWORK DYNAMICS WITH ENSEMBLES OF GRAPH PARTITIONS

SAKET NAVLAKHA AND CARL KINGSFORD*

Center for Bioinformatics and Computational Biology, and
Department of Computer Science,
University of Maryland, College Park, MD 20742.

Unveiling the modular structure of biological networks can reveal important organizational patterns in the cell. Many graph partitioning algorithms have been proposed towards this end. However, most approaches only consider a single, optimal decomposition of the network. In this work, we make use of the multitude of near-optimal clusterings in order to explore the dynamics of network clusterings and how those dynamics relate to the structure of the underlying network. We recast the modularity optimization problem as an integer linear program with diversity constraints. These constraints produce an ensemble of dissimilar but still highly modular clusterings. We apply our approach to four social and biological networks and show how optimal and near-optimal solutions can be used in conjunction to identify deeper community structure in the network, including inter-community dynamics, communities that are especially resilient to change, and core-and-peripheral community members.

1. Introduction

Many types of biological networks, such as protein interaction networks and metabolic networks, are known to be modular in nature[15]. Modules are typically composed of a set of nodes that are all functionally related. Uncovering such functional building blocks is useful because it can provide us with a systems-level understanding of how the cell is organized. Several graph partitioning algorithms have been recently proposed for this purpose (e.g.[40,2,30,32,34]), but these algorithms typically select only a single solution from the vast space of possible clusterings. The chosen solution is meant to characterize the modular structure of the data, but it ignores the horde of near-optimal solutions.

Near-optimal solutions are crucial in many respects. For example, they can help assess confidence in the optimal partitioning. If a near-optimal solution is nearly as good as the optimal, we may be unsure whether it is the near-optimal or the optimal partitioning that represents the true community structure. This is especially true in the presence of noise, when the true community structure might be obscured and as a result only emerge as some near-optimal solution. More locally, pairs of nodes that are co-clustered in many near-optimal partitionings can be confidently determined to be members of the same community. Equivalence classes of these frequently co-clustered nodes can be considered the "core" members of a community. Others ought to be considered tenuous or "peripheral" members. Thus, unlike single solution approaches that treat each individual as an equivalent community member, near-optimal solutions provide a way to measure the strength of membership of members to each community. Further, understanding inter-and intra-community interactions can be used to quantify how robust or resilient a community is to change. By taking such interactions into account, we transition from treating communities as static, independent blobs to dynamic blobs with varying memberships. Finally, there is also theoretical and empirical evidence suggesting that single point solutions in high-dimensional spaces do not represent the data as well as ensembles of solutions[4]. This is particularly true in machine learning, where ensembles of classifiers have been consistently shown to outperform single models[36,37].

In this article, we look at a broad collection of social and biological networks and show how near-optimal clusterings impart information into community dynamics that would otherwise be missed using single solution approaches. We use the popular modularity partitioning criteria proposed by Newman[34]. Modularity has received mixed reviews regarding its relevance to biological networks. It was shown to perform poorly at

*Corresponding author: carlk@cs.umd.edu.

recovering functional modules from large protein interaction networks[31], but it has been effective at finding modules in smaller metabolic networks[11]. We consider it here for smaller networks, but use it simply as a template to investigate near-optimal solutions. It is likely that other approaches will also reap similar benefits by considering ensembles of solutions.

To explore near-optimal community partitions, we cast modularity optimization as an integer linear program (ILP), as has been done before[1,3], but add diversity constraints so that each subsequent clustering is not only adequately different from all previous solutions, but also has high modularity. This way, we directly optimize for both diversity and quality. The collection of solutions returned constitute a partial "energy landscape" that represents overlaid decompositions of the network.

Several techniques have been proposed for finding ensembles of optimal and near-optimal solutions to similar ILP problems. For example, both randomly perturbing objective function weights by a small amount[29,12], or perturbing the input data itself and re-clustering[16,21], can help explore different regions of the clustering space (though selecting the size of that perturbation can be difficult). Alternatively, a randomized rounding procedure to convert a fractional solution to an integral one can be used[1], yielding a slightly different partitioning each time. In addition, heuristic techniques such as simulated annealing[11,27] can be used instead of ILPs to optimize the modularity. Such approaches explicitly explore the state space, and an ensemble of partitionings can be generated by saving any good solutions observed. But these techniques are all based on the idea of randomization: perturbing the inputs or the outputs randomly, or randomly transitioning between solutions. Such randomized procedures suffer from at least two deficiencies. First, they often yield solutions very similar to the optimal because large deviations are improbable to be generated at random. Secondly, there is no guarantee that the perturbed solutions have high modularity. The randomized procedure may generate many diverse solutions of poor quality. Other approaches vary input parameters, such as the number of clusters to return[22], though there can exist multiple reasonable clusterings that have the same number of clusters. Recently, another approach was proposed that systematically perturbs the input data such that the transformed and original data retain similar properties; an alternative clustering is then found by clustering the transformed data[38]. Here, we take the approach of explicitly constraining for diversity within the clustering process itself. This guarantees that each successive solution is both sufficiently different from previously obtained solutions and achieves the maximum possible modularity attainable under the given diversity criteria.

We explore a broad spectrum of social and biological networks in an attempt to show the types of insights that can be extracted from large collections of near-optimal solutions. We begin with Zachary's karate club social network[43], which documents the fission of a group of university students after an internal dispute over the price of karate lessons. Interestingly, we find that the clustering closest to the actual resulting fission of the club (i.e. the true clustering) does not appear until the 31^{th} near-optimal solution. We also show that exploring near-optimal solutions can help identify fringe members of the two factions.

We next look at the ERK1/ERK2 mitogen-activated protein kinase (MAPK[18]) signal-transduction pathway. We identify functional subunits that correspond well to known submodules of the pathway, and we classify their robustness across the modularity landscape. Two portions of the ERK pathway consistently remain tightly bound, whereas all other components are eventually split. We also identify *gatekeeper* nodes that lie between functional modules in the Integrin signalling pathway[26].

Finally, we consider a network of cortical-cortical connections in the human brain and find 53 of the first 60 near-optimal solutions are within 1% of the optimal modularity. Of these, 12 have a $> 3\%$ advantage in spatial coherence over the optimal clustering, indicating that they might better represent the true modules of the brain. Differentially classified nodes in this case can be used to identify spatial outliers with respect to the topology. The immense number of similar solutions also suggests tremendous uncertainty in the optimal partitioning.

In all four networks, we find insights conveyed by near-optimal partitionings that helps augment our current understanding of community structure and dynamics.

2. Generating A Diverse Ensemble of Partitionings

Below, we describe our procedure for generating an ensemble of distinct, high-modularity clusterings using integer linear programming (ILP). All superscripts used below indicate indices, not exponentiation.

2.1. *Integer Programming for Modularity*

Intuitively, maximizing modularity corresponds to finding communities where the number of edges lying within a cluster is much greater than we would expect by chance (under an Erdös-Renyi null distribution), and the number of edges connecting two different clusters is much less. Formally, the modularity $q(G, \mathcal{C})$ of an undirected, unweighted network G with community decomposition $\mathcal{C}$ is defined as

$$q(G, \mathcal{C}) := \sum_{u,v \in V} \left(A_{uv} - k_u k_v/(2m)\right)(1 - x_{uv}), \tag{1}$$

where A_{uv} is an entry in the adjacency matrix for G (it is 1 if u and v interact and 0 otherwise), k_u is the degree of node u, m is the total number of edges, and the variables x_{uv} describe $\mathcal{C}$ by indicating which vertices are in the same community. More specifically, we have a variable x_{uv} for every pair of nodes $u < v$, with the interpretation that $x_{uv} = 1$ if u and v belong to different clusters, and $x_{uv} = 0$ otherwise. Letting $m_{uv} = A_{uv} - k_u k_v/(2m)$, a pair of nodes u, v in the same cluster contributes m_{uv} to the total modularity (m_{uv} may be negative). Hence, we seek to maximize $\sum_{u,v} m_{uv}(1 - x_{uv})$ by setting the x_{uv} variables appropriately.

To ensure that the nodes identified as co-clustered are consistent with each other, we must enforce the triangle inequality. This leads to the following integer linear program, MOD-ILP:

$$\text{maximize} \quad \sum_{u \in V} \sum_{v \in V} m_{uv}(1 - x_{uv}) \tag{2}$$

subject to

$$x_{uv} + x_{vw} \geq x_{uw} \qquad \text{for all } u, v, w \in V \tag{3}$$

$$x_{uv} \in \{0, 1\} \tag{4}$$

This ILP is identical to the one proposed by Agarwal et al.[1] for modularity maximization and is similar to the ILP proposed for correlation clustering by Charikar et al.[5] Another similar ILP, where instead $x_{uv} = 1$ indicates that u and v are in the same cluster and with consequently modified constraints, was proposed by Brandes et al.[3] Here, we use MOD-ILP as a tool to generate ensembles of diverse community decompositions, as described in the next section. The ILP can be solved to optimality via branch-and-bound using an ILP solver such as `glpk`[25] or CPLEX[17]. There are $\binom{n}{2}$ variables and $3\binom{n}{3}$ constraints, where n is the number of nodes. For large networks solving the ILP to optimality can be time consuming. Hence, a rounding heuristic has been proposed[1] based on an approximation algorithm for correlation clustering[5]. In this approach, the integrality constraints (4) are replaced by constraints requiring $0 \leq x_{uv} \leq 1$ and the fractional solution is rounded, treating the fractional x_{uv} values as pairwise distances between the nodes. In this article, we focus on smaller networks that can be solved to optimality. However, for larger networks the LP-relaxation of MOD-ILP (with subsequent rounding) can be used, along with the same diversity constraints that are discussed below.

2.2. *Diversity Constraints*

A solution to MOD-ILP reveals only one possible partitioning of the network. Suppose X^0 is a $\binom{n}{2}$-vector $\langle x^0_{uv} \rangle$ representing an optimal solution to MOD-ILP, and let $\vec{1}$ be the $\binom{n}{2}$-vector with every component equal to 1. The following constraints require a vector X to be different from vector X^0:

$$X^0 \cdot (\vec{1} - X) \geq d^0_{\text{merge}} \tag{5}$$

$$(\vec{1} - X^0) \cdot X \geq d^0_{\text{split}} \tag{6}$$

Here, $\cdot$ denotes the dot product between the vectors. Considering X^0, d^0_{merge} and d^0_{split} to be constants, the constraints represented in (5) and (6) are linear. By adding them to MOD-ILP and finding a new optimal, the ILP is forced to return a solution X that is different from X^0. The amount of difference is governed by the parameters d^0_{split} and d^0_{merge}. Equation (5) requires that at least d^0_{merge} variables change from 1 to 0, thereby requiring that d^0_{merge} pairs of nodes formerly in separate clusters become co-clustered. Similarly, equation (6) requires that at least d^0_{split} pairs that were co-clustered in X^0 are placed in separate clusters in X. The parameters d^0_{merge} and d^0_{split} can be set to vary the level and type of diversity desired. Both constraints are required to balance between larger-and smaller-sized clusters, respectively. We can avoid setting separate levels of each diversity type by consolidating these constraints:

$$X^0 \cdot (\vec{1} - X) + (\vec{1} - X^0) \cdot X \geq d_{\text{changes}}\,, \tag{7}$$

where the left-hand side is equivalent to the Hamming distance, $\Delta(X, X^0)$, between vectors X and X^0. Re-solving MOD-ILP with constraint (7) added will find an alternative optimal (if one exists) or will find a second-best partitioning.

To speed up the solution of the ILP, we can use a heuristic algorithm to find a reasonable partitioning and then supply that partitioning to the integer programming solver as an initial basis. Here, this was necessary only for the Integrin pathway and the human brain network, where we used the partitioning found by Newman's spectral method[34] as a starting basis. This provided the solver a starting point for the branch-and-bound process and resulted in convergence in minutes (as opposed to hours). Such an initial basis does not alter the optimality of the solution found.

2.3. *Modularity Landscape*

A partial "modularity landscape" of a network can be generated by iteratively solving MOD-ILP including constraint (7) while increasing d_{changes}. If X^i is the solution of the ith iteration, in the $i+1$ interaction, we set

$$d^{i+1}_{\text{changes}} = \Delta(X^0, X^i) + 1. \tag{8}$$

In contrast to repeated sampling using, e.g., simulated annealing[11,27], this approach guarantees that successively obtained partitionings maximize modularity while still being sufficiently different from the optimal, X^0. We call this the *distance-based* method of generating diverse solutions.

An alternative method for generating an ensemble of diverse, high-modularity partitionings is to repeatedly resolve MOD-ILP with the addition of several constraints of the form of Equation 7, one for each previously uncovered solution. In other words, on the ith iteration, for each previous solution X^j $(0 \leq j < i)$, we add a constraint $X^j \cdot (\vec{1} - X) + (\vec{1} - X^j) \cdot X \geq 1$ to MOD-ILP. A new solution will have at least one difference from each previously uncovered solution. We call this the *point-based* method because it is akin to avoiding specific markers on the clusterings space. The point-based method produces clusterings that are finer-grained than the distance-based approach because there can exist many solutions having distance between d^i_{changes} and d^{i+1}_{changes} that the distance-based method would miss. Using the point-based method, the i^{th} solution returned is a provably i^{th} optimal network decomposition in terms of modularity (clusterings with identical modularity will be ordered arbitrarily). The distance-based method more quickly samples a more diverse collection of solutions. By setting $d_{\text{changes}} > 1$, the point-based approach could also be adopted to more rapidly sample the solution space. In the results described below, we experiment with the distance-based approach and the point-based approach with $d_{\text{changes}} = 1$.

2.4. *Determining Core and Peripheral Community Members*

Nodes that travel together across the modularity landscape can be thought of as *core* members of a community. Such nodes remain together despite the additional diversity constraints added, which implies that their cohesion is stronger than that of other pairs of nodes. Nodes whose co-clustered neighbors fluctuate across

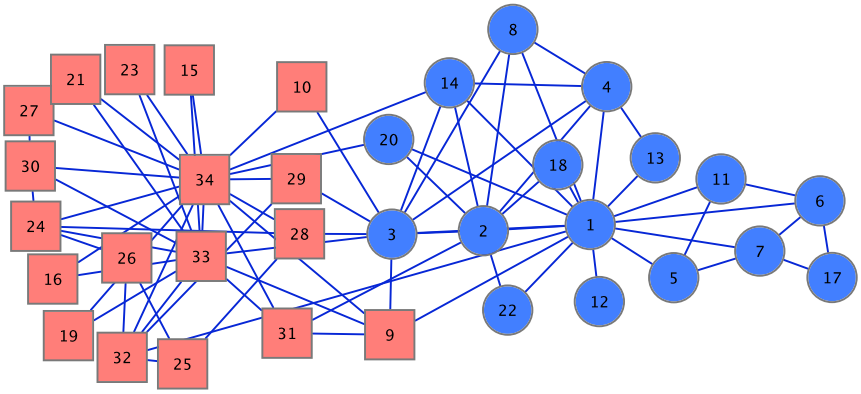


Figure 1. **Zachary's karate club social network**[43]**.** The network consists of 34 nodes and 78 edges. Blue circles correspond to Mr. Hi's faction. Red squares correspond to the officers' faction.

solutions can be considered *peripheral* members that lie on the outskirts of the community. We find core and peripheral members of communities by creating a co-clustering matrix whose entries equals the number of clusterings in the landscape in which nodes u and v are co-clustered. Dense blocks in the matrix correspond to core members; cavities within dense blocks indicate peripheral activity or overlapping modules. Such matrices have been previously investigated in a different context — consensus clustering[28,9] — where the goal is typically to return a centroid clustering that lies centrally amongst a given set of input clusterings. Finding core and peripheral proteins within dense subgraphs in protein interaction networks has also recently been shown to be useful for protein complex identification[8,23,24]. We use the co-clustering matrix as a means to identify inter-and intra-module clustering dynamics.

3. Results

We used MOD-ILP with diversity constraints to produce modularity landscapes for the karate club social network[43], the ERK1/ERK2 MAPK[18] and Integrin[26] metabolic pathways, and a coarse-level human brain network[13]. For each network, we show how exploring ensembles of near-optimal solutions reveals clustering dynamics that would otherwise be missed by single solution approaches.

3.1. *Karate Club Network*

We begin by studying the modularity landscape of Zachary's karate club network[43], shown in Figure 1. This network consists of 34 nodes and 78 social-interaction edges. Due to an internal dispute over the price of karate lessons, the group split into two factions, one corresponding to the club's karate instructor, Mr. Hi, and the other to the club's officers. Although not a network derived from molecular biology, it has the advantage of being small enough to examine by hand and to have hand-curated evidence regarding social interactions and community membership.

The distance-based approach found 82 different clusterings, after which no more feasible clusterings existed. These clusterings had between 1 and 5 communities. Figure 2 shows the modularity landscape produced by MOD-ILP with diversity constraints using the distance-based approach. In each panel, the x-axis gives the solution number. The y-axis in the top panel shows the distance from each solution to the optimal solution; the y-axes of the middle and bottom panels show the solution's modularity and number of communities, respectively. The number of communities does not monotonically decrease with lower modularity. Instead, different components join or break-off as dictated by the resulting modularity and diversity constraints. This implies that our ensembles do not simply correspond to iteratively choosing different levels in the modularity hierarchical tree decomposition[6].

Although the true structure consisted of 2 communities, the optimal modularity solution (with modularity

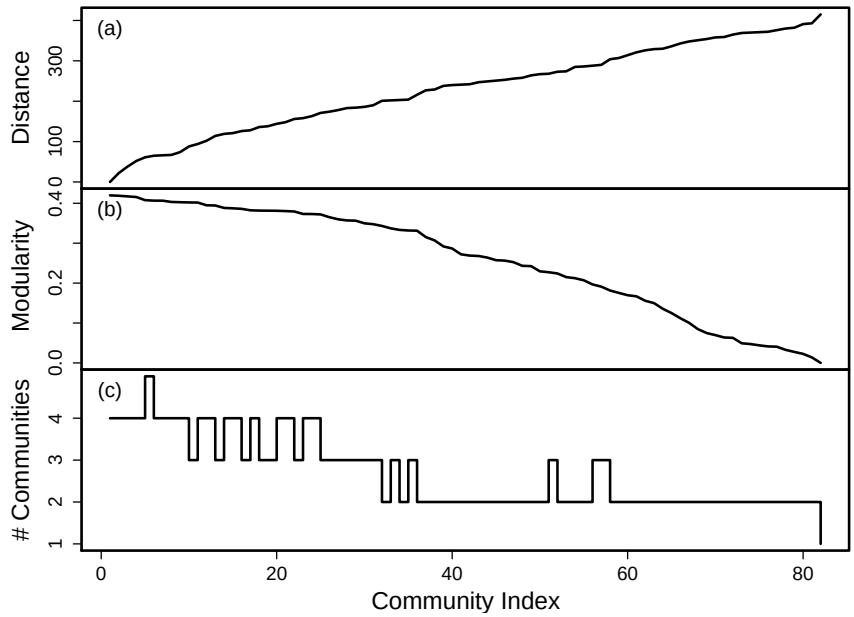


Figure 2. **Modularity landscape of the karate club network.** The x-axis in all panels shows the community index (ordered list of clusterings returned by iterative runs of MOD-ILP with distance-based diversity constraints). The 0^{th} community index corresponds to the optimal modularity clustering. (a) The Hamming distance from each clustering to the optimal modularity clustering. (b) The modularity of each clustering. There are 10 clusterings with modularity > 0.4, and 37 with modularity > 0.3. (c) The number of communities in each clustering.

0.419790) had four clusters (with each faction broken into two communities). The network is not split into the two communities until the 31^{st} solution. This solution has modularity 0.343195 and corresponds closely to the actual groups formed (with the exception of nodes $9, 10, 20$, and 31 — all topologically fringe, three of which were weak supporters of their faction leaders[43]). Such a solution would never be found unless near-optimal solutions were considered. Further, randomized rounding procedures would be unable to generate diverse solutions for this network, because even when the integrality constraints were relaxed, allowing $x_{uv} \in [0, 1]$, an integral solution was returned. This argues for the necessity of a constraint-based approach.

The point-based method, which only constrains each solution to be minimally different from all previous solutions, produced many more finer-grained solutions corresponding to incremental merging and splitting of communities. In fact, the 100^{th} solution of the point-based approach still had a modularity above 0.4. Although this level of detail could be useful for some applications, here we seek to more coarsely characterize the clustering dynamics, and therefore only further consider the distance-based solutions.

Dynamics for individual nodes can be better understood by looking at near-optimal solutions. For example, the solution with the provably second-best modularity, which is also the clustering that is output by Newman's spectral method[34], consists of 4 clusters but with slightly smaller modularity (0.418803) than the optimum. The difference lies in the classification of node 10, which, in the second-best clustering is placed with Mr. Hi and in the optimal clustering is placed with the officer's faction. Zachary measures the strength of friendship between pairs of individuals based on their interactions in other social contexts (for example, academic classes, student pubs, and other karate studios[43]) and finds that node 10 had nearly equal interaction with members from both factions. Node 10 was also not a strong believer in either faction's ideology (although he ultimately chose the officer's club after the fission). Hence, it makes sense that node 10 was the first to jump from one clustering to the other.

Another interesting case occurs for node 20. He lies in Mr. Hi's faction in the optimal clustering, but in subsequent clusterings is co-clustered with members from the officer's faction. According to Zachary, node 20 ultimately chooses Mr. Hi's club, but only weakly supported Mr. Hi's position in the dispute[43]. Looking

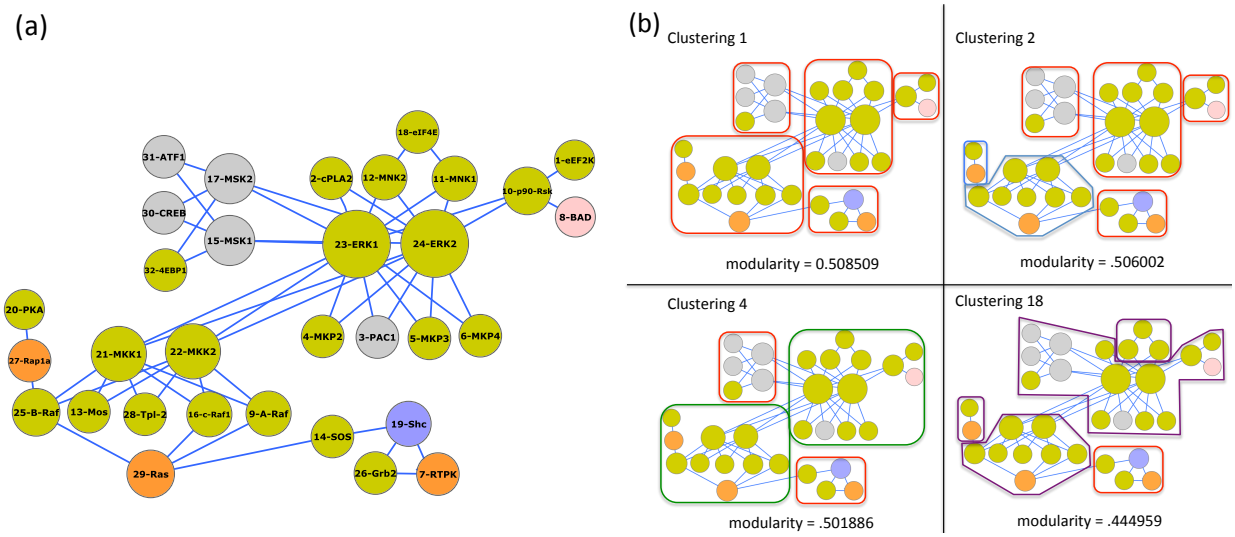


Figure 3. (a) **The ERK1/ERK2 MAPK signalling pathway**[18]**.** The network consists of 32 nodes and 54 edges. The color of the node indicates the subcellular localization of the signalling component (green = cytosol, orange = plasma membrane, gray = nucleus, blue = plasma membrane translocation, and pink = mitochondrion). The network was drawn using Cytoscape[39]. (b) **"Flip-book" showing the clustering dynamics of the ERK1/ERK2 MAPK pathway**. Each of the four blocks corresponds to a clustering produced by MOD-ILP with distance-based diversity constraints. The number of the clustering is shown at the top, and its modularity on the bottom. Each cluster is blocked within a polygonal shape. A variety of near-optimal clusterings provide alternative, legitimate decompositions of the network.

at the network, 20 is connected to both faction leaders, plus an additional supporter of Mr. Hi. Topologically and anecdotally, it seems to make sense then that node 20 is a peripheral member of Mr. Hi's karate club.

Trying to identify core and peripheral nodes by only looking at the neighbors of a node, however, can be misleading. Node 3, for example, is a topologically fringe node with 10 total edges, 5 to members in both factions. But, according to Zachary[43], node 3 was a strong supporter of Mr. Hi, whose club he joined after fission. In our ensemble, we only see node 3 switch from a Mr. Hi-dominant clustering to a clustering dominated by officer members three times. These all occur near the end of the landscape, at clusterings 72, 78, and 80, which have a very low modularity (average = 0.030188). Using only network neighbors to classify a node as core or peripheral is therefore not always sufficient. Further, the landscape also provides a way to confidently say what groups of nodes do not belong together. A static analysis of the optimal clustering will clearly be unable to understand these type of community dynamics.

3.2. *Signalling Networks*

We considered the ERK1/ERK2 mitogen-activated protein kinase (MAPK) pathway[18] shown in Figure 3a. MAPK is a signal-transduction pathway that is highly-conserved across eukaryotes. MAPKs phosphorylate serines and threonines of target proteins and regulate a vast array of cellular functions, including gene expression, mitosis, and metabolism[19]. The extra-cellular signal-regulated kinases (ERKs) play a functional role in cell division, in particular meiosis and mitosis[19]. Identifying functional modules in such pathways is important because modules are often conserved across organisms, and thus can be used to generate new pathways from reference pathways[20,41]. The pathway consists of 32 nodes and 54 edges.

Figure 3b shows four snapshots of the modularity landscape. The optimal modularity (clustering 1) consists of five clusters roughly corresponding to nodes surrounding the Ras activation module, the Raf and MEK kinase modules, and the larger ERK module (split into three) — all known submodules of the pathway. In subsequent clusterings, nearby cores are either split or merged together, corresponding to finer- and coarser-grained functional subunits of the pathway. As in the karate network, we also find that the number of clusters does not simply monotonically decrease (or increase) as the diversity constraint, d_{changes},

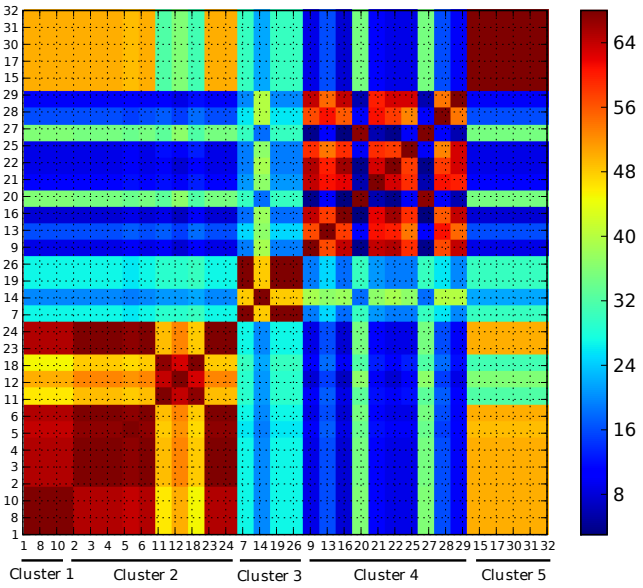


Figure 4. **Co-clustering heatmap for the ERK1/ERK2 MAPK pathway[18].** A broad view of how pairs of nodes traverse the modularity landscape. Each cell (u, v) in the heatmap corresponds to the number of clusterings in which nodes u and v were placed together. The nodes are ordered according to the optimal modularity found by MOD-ILP. Though outlines of the five optimal modules are present, the fluctuation of activity within and between the five blocks reveal interesting inter- and intra-community interactions.

is increased.

Figure 4 shows a global view of how the affiliation between each pair of nodes changes across clusterings. The intensity of cell (u, v) in the heatmap corresponds to the number of clusterings in the landscape in which nodes u and v are co-clustered. A similar picture was obtained by setting the intensity of a cell (u, v) to be the total modularity sum of all clusterings in which u and v were co-clustered. The nodes are ordered based on the clusters from the optimal modularity clustering.

The outlines of the five optimal blocks in Figure 4 provide a basic hint about the modular structure of the pathway, but it does not tell the whole story. For example, nodes 20 (PKA) and 27 (Rap1a) travel together much more than 27 and 13 (Mos), even though all three were placed together in the same optimal module. From the layout shown in Figure 3a, this makes sense — PKA and Rap1a are connected to the core Raf module by only one edge, and are also connected to each other. This suggests that they play a peripheral role in the module in which they were placed, or perhaps that they should be placed together in their own module.

The heatmap also provides a way to measure the confidence in a community by looking at how a group of nodes change their membership with respect to each other. For example, nodes 15, 17, 30, 31, and 32, corresponding to a portion of the ERK module, were co-clustered across all clusterings, as indicated by the solid red block in the upper-right corner of Figure 4. This implies that we are very confident in this module, more so than any other. Other clusters vary greatly with respect to how often their members travel together. An optimal clustering alone would yield a heatmap with solid red blocks for all clusters, which is much less informative of community membership strength.

We also looked at the Integrin signalling pathway[26], known to be vital for cell migration and growth. This pathway is longer and less dense than the ERK/MAPK pathway. The optimal modularity clustering found a reasonable decomposition consisting of modules with long chains of nodes. These long chains are often prefaced by *gatekeeper* nodes that branch off multiple non-overlapping paths. Network centrality measures

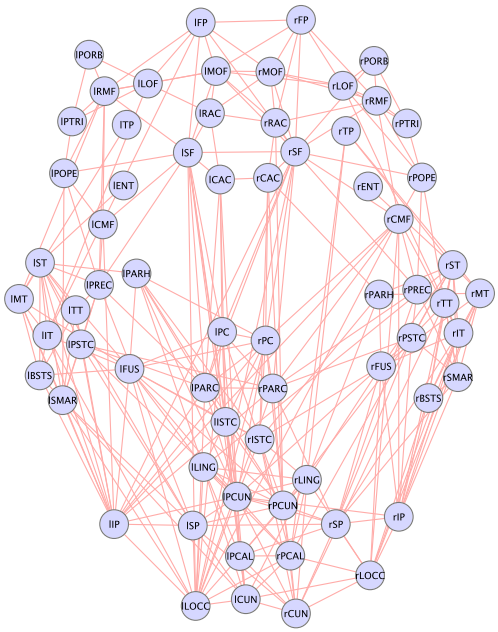


Figure 5. **Anatomical network of the human cerebral cortex**[13]**.** The network consists of 66 nodes (brain regions) and 2,149 multiedges (dense axonal-pathways). The 'r' and 'l' prefixes correspond to the right and left hemispheres of the brain. The remaining portion of the names correspond to cortical regions (e.g. ENT = entorhinal cortex, TP = temporal pole, PC = posterior cingulate cortex, CUN = cuneus, and PARH = parahippocampal cortex). The layout is set to spatially agree with the actual positions of the regions in the brain. Coordinates of the regions were estimated from Figure 6 in Hagmann et al.[13], as well as from the Brede database[35]. The network was drawn using Cytoscape[39].

have also been used to globally identify highly "between" nodes,[42,14] though they do not typically take modules into account. Interestingly, many of the near-optimal clusterings identified these gatekeepers by placing them into different modules corresponding to the various branches. For example, the Cdc42 protein acts as a between-module node that ultimately leads to activation of actin and the c-Jun N-terminal kinases (JNK). It was first placed amongst nodes in the JNK module, but later switches into the actin module. A similar dynamic was seen for branches leading out of the focal adhesion kinase (FAK), which is involved in cellular adhesion and migration.

3.3. *Brain Network*

Lastly, we investigated a network representing the axonal-pathways within the cortex of the human brain. Brain maps have typically been constructed using functional magnetic resonance imaging (fMRI), which measures neural activity via blood flow (e.g.[10,7]). Recently, Hagmann et al.[13] used a technique called diffusion spectrum imaging (DSI) which identifies neuronal fiber trajectories by looking at the diffusion of water molecules in brain tissues. DSI produces a 3D water-flow gradient at specified positions in the brain to which tractography can be applied to recover the underlying neural tracts. Tractography identifies, for each position, the diffusion of water to that position from all other directions. Thus, we can determine the axonal trajectories across white matter, i.e. the connectivity across different regions in the brain. Regions are typically defined manually after white-gray matter segmentation (white = nerve connections, gray = congregations of neurons). The resulting network is composed of nodes (brain regions) and weighted edges, corresponding to the density of the connection between brain regions.

Hagmann et al.[13] applied their technique to generate a brain "connectome" for five human participants. Each connectome consists of 998 regions of interest. They also created a coarser network by condensing the 998 regions into 66 anatomical regions. An edge (u, v) in the anatomical network was weighted by computing the average of all edges that map to (u, v). To handle weighted networks, we used an extended version of

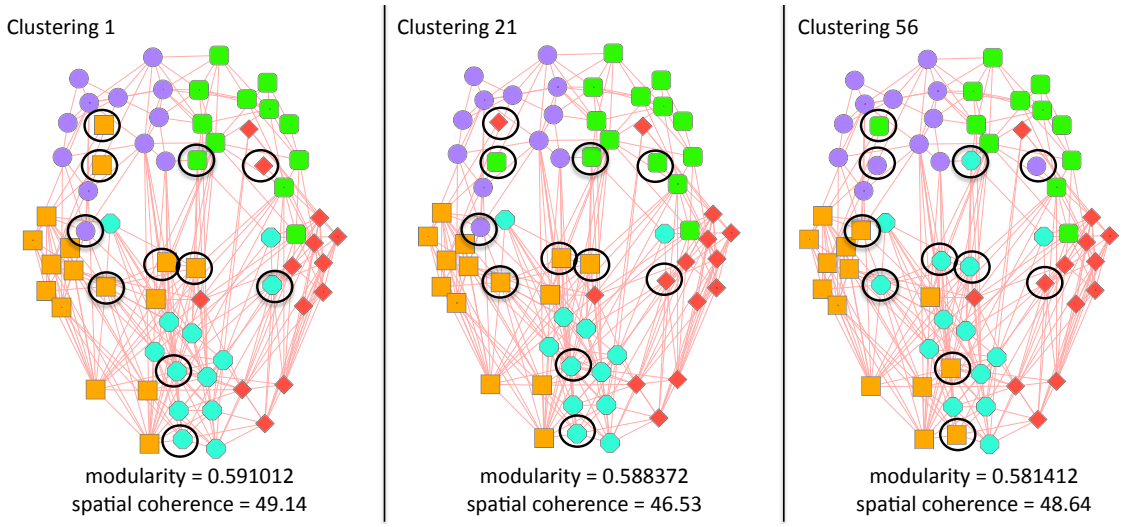


Figure 6. **"Flip-book" showing the clustering dynamics of the anatomical brain network[13].** Each of the three blocks corresponds to a clustering produced by MOD-ILP with distance-based diversity constraints. The number of the clustering is shown at the top, with modularity and spatial coherence on the bottom. Co-clustered nodes share the same color and shape. Black circles highlight nodes whose communities change across clusterings. The optimal modularity clustering does not have the highest coherence with the spatial coordinates of the regions.

modularity that converts weighted edges to unweighted, multi-edges[33]. In particular, in the multi-edged anatomical network we created $\lfloor 1000 \cdot w(u,v) \rfloor$ edges between nodes u and v, where $w(u,v)$ is the weight of edge (u,v) in the weighted anatomical network. The only change required in the definition of modularity is with A_{uv}, which is now the number of edges that go between u and v, instead of just 0 or 1. The final anatomical network contained 66 nodes and $2,149$ multiedges. Hagmann et al.[13] applied modularity to the anatomical network to identify regional hubs.

We ran MOD-ILP with diversity constraints on the first subject's human connectome (Figure 5). The similarity between the modularity values of the near-optimal solutions suggest extreme uncertainty in whether the optimal solution represents the true partitioning. Figure 6 shows the optimal clustering plus two near-optimal clusterings returned by the distance-based approach. The near-optimal clusterings are only slightly less topologically modular. In fact, amongst the first 60 solutions, we find that 53 are within 1% of the optimal modularity.

The brain network is unique among those that we consider because the nodes have a fixed spatial position. Hagmann et al.[13] assigned spatial coordinates to each region corresponding to its center of mass, but because not all spatial coordinates were available, the layout in Figure 5 is a taken from the layout drawn in Hagmann et al.[13] Three-dimensional spatial coordinates were directly available for 23 of the 66 regions. An additional 15 regions were assigned spatial coordinates based on averaging the coordinates from several studies for the relevant region using the Brede neuroimaging database[35].

The spatial coordinates themselves define a rough clustering, which can be used as an additional measure (along with modularity) to evaluate the likelihood of a particular brain network partitioning. We defined the *spatial coherence* of a clustering as the average Euclidean distance between anatomical regions placed in the same cluster. The near-optimal clusterings shown in Figure 6 have a better spatial coherence than the optimal solution at only a tiny decrease in modularity, despite having the same number of clusters. In fact, out of the 60 near-optimal solutions 30 of these solutions have a $> 1\%$ advantage in spatial coherence, 24 have a $> 2\%$ advantage, and 12 have a $> 3\%$ advantage. Naturally, clusters and nodes that do not match what is expected spatially may be the most interesting to investigate further. The nodes that are differentially clustered within the ensemble of solutions (circled in black in Figure 6) are typically such spatial outliers.

4. Conclusions

We investigated the clustering dynamics of four social and biological networks to reveal how these networks are organized. In all four settings, we showed how traversing the modularity landscape by explicitly constraining for diversity can be used to uncover deeper community structure that would otherwise be absent from single-solution or randomization-based procedures. In particular, we used ensembles of near-optimal network decompositions to identify resilient communities, core-peripheral community members, and finer- and coarser-grained community structure. We also found cases where near-optimal solutions corresponded better with known community structure than the optimal solution. We presented mostly anecdotal evidence regarding inter-and intra-module dynamics. Testing these notions on a large scale, such as for the automated identification of core-peripheral proteins in protein complexes[8,23,24], is a potential avenue for future work. It would also be interesting to characterize the relationship between clustering dynamics across the energy landscape and clustering dynamics across time. Nonetheless, we believe the insights provided by near-optimal solutions augment our current understanding of community structure and dynamics, and should not be ignored.

Acknowledgments

This work was partially supported by grants 0849899 and 0812111 from the National Science Foundation.

References

1. G. Agarwal and D. Kempe. Modularity-maximizing graph communities via mathematical programming. *Eur. Phys. J. B*, 66(3):409–418, 2008.
2. G. D. Bader and C. W. V. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4:2, 2003.
3. U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner. On modularity clustering. *IEEE T. Knowl. Data En.*, 20(2):172–188, 2008.
4. L. E. Carvalho and C. E. Lawrence. Centroid estimation in discrete high-dimensional spaces with applications in biology. *Proc. Natl. Acad. Sci. USA*, 105(9):3209–3214, 2008.
5. M. Charikar, V. Guruswami, and A. Wirth. Clustering with qualitative information. In *FOCS '03: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 524–533, 2003.
6. A. Clauset, M. E. Newman, and C. Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70(6 Pt 2):066111, 2004.
7. M. D. Fox, A. Z. Snyder, J. L. Vincent, M. Corbetta, D. C. Van Essen, and M. E. Raichle. The human brain is intrinsically organized into dynamic, anticorrelated functional networks. *Proc. Natl. Acad. Sci. USA*, 102(27):9673–9678, 2005.
8. A. C. Gavin, P. Aloy, P. Grandi, R. Krause, M. Boesche, M. Marzioch, C. Rau, L. J. Jensen, S. Bastuck, B. Dumpelfeld, A. Edelmann, M. A. Heurtier, V. Hoffman, C. Hoefert, K. Klein, M. Hudak, A. M. Michon, M. Schelder, M. Schirle, M. Remor, T. Rudi, S. Hooper, A. Bauer, T. Bouwmeester, G. Casari, G. Drewes, G. Neubauer, J. M. Rick, B. Kuster, P. Bork, R. B. Russell, and G. Superti-Furga. Proteome survey reveals modularity of the yeast cell machinery. *Nature*, 440(7084):631–636, 2006.
9. A. Goder and V. Filkov. Consensus clustering algorithms: Comparison and refinement. In *ALENEX '08: Proceedings of the Workshop on Algorithm Engineering and Experiments*, pages 109–117. SIAM, 2008.
10. M. D. Greicius, B. Krasnow, A. L. Reiss, and V. Menon. Functional connectivity in the resting brain: a network analysis of the default mode hypothesis. *Proc. Natl. Acad. Sci. USA*, 100(1):253–258, 2003.
11. R. Guimerà and L. A. Nunes Amaral. Functional cartography of complex metabolic networks. *Nature*, 433(7028):895–900, 2005.
12. S. T. Hadjitodorov, L. I. Kuncheva, and L. P. Todorova. Moderate diversity for better cluster ensembles. *Inform. Fusion*, 7(3):264–275, 2006.
13. P. Hagmann, L. Cammoun, X. Gigandet, R. Meuli, C. J. Honey, V. J. Wedeen, and O. Sporns. Mapping the structural core of human cerebral cortex. *PLoS Biol.*, 6(7):e159, 2008.
14. M. W. Hahn and A. D. Kern. Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks. *Mol. Biol. Evol.*, 22(4):803–806, 2005.
15. L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–52, 1999.

16. J. Hopcroft, O. Khan, B. Kulis, and B. Selman. Tracking evolving communities in large linked networks. *Proc. Natl. Acad. Sci. USA*, 101 Suppl 1:5249–5253, 2004.
17. IBM Ilog, Inc. Solver CPLEX, 2009. http://www.ilog.com/products/cplex/ (accessed 7 July 2009).
18. G. L. Johnson. ERK1/ERK2 MAPK pathway. *Sci. Signal. Connections Map in the Database of Cell Signaling*, 2009.
19. G. L. Johnson and R. Lapadat. Mitogen-activated protein kinase pathways mediated by ERK, JNK, and p38 protein kinases. *Science*, 298(5600):1911–1912, 2002.
20. M. Kanehisa and S. Goto. KEGG: Kyoto encyclopedia of genes and genomes. *Nuc. Acids Res.*, 28(1):27–30, 2000.
21. B. Karrer, E. Levina, and M. E. J. Newman. Robustness of community structure in networks. *Phys. Rev. E*, 77(4):046119, 2008.
22. L. I. Kuncheva and S. T. Hadjitodorov. Using diversity in cluster ensembles. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 2, pages 1214–1219 vol.2, 2004.
23. H. C. Leung, Q. Xiang, S. M. Yiu, and F. Y. Chin. Predicting protein complexes from PPI data: a core-attachment approach. *J. Comp. Biol.*, 16(2):133–144, 2009.
24. F. Luo, B. Li, X. F. Wan, and R. H. Scheuermann. Core and periphery structures in protein interaction networks. *BMC Bioinformatics*, 10 Suppl 4:S8, 2009.
25. A. Makhorin. *GNU Linear Programming Kit, Version 4.26*. GNU Software Foundation, http://www.gnu.org/software/glpk/glpk.html.
26. K. H. Martin, J. K. Slack, S. A. Boerner, C. C. Martin, and J. T. Parsons. Integrin signaling pathway. *Sci. Signal. Connections Map in the Database of Cell Signaling*, 2009.
27. C. P. Massen and J. P. Doye. Identifying communities within energy landscapes. *Phys. Rev. E*, 71(4 Pt 2):046101, 2005.
28. S. Monti, P. Tamayo, J. P. Mesirov, and T. R. Golub. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning*, 52(1-2):91–118, 2003.
29. E. Nabieva, K. Jim, A. Agarwal, B. Chazelle, and M. Singh. Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. *Bioinformatics*, 21 Suppl 1:i302–i310, 2005.
30. S. Navlakha, R. Rastogi, and N. Shrivastava. Graph summarization with bounded error. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pages 419–432, New York, NY, USA, 2008. ACM.
31. S. Navlakha, M. C. Schatz, and C. Kingsford. Revealing biological modules via graph summarization. *J. Comp. Biol.*, 16(2):253–264, 2009.
32. S. Navlakha, J. White, N. Nagarajan, M. Pop, and C. Kingsford. Finding biologically accurate clusterings in hierarchical tree decompositions using the variation of information. In *RECOMB '09: Proceedings of the 13th Annual International Conference on Research in Computational Molecular Biology*, volume 5541, pages 400–417, 2009.
33. M. E. Newman. Analysis of weighted networks. *Phys. Rev. E*, 70(5 Pt 2):056131, 2004.
34. M. E. J. Newman. Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA*, 103(23):8577–8582, June 2006.
35. F. A. Nielsen. The Brede database: a small database for functional neuroimaging. In *The 9th International Conference on Functional Mapping of the Human Brain*, 2003.
36. D. Opitz and R. Maclin. Popular ensemble methods: an empirical study. *J. Artif. Intell. Res.*, 11:169–198, 1999.
37. R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006.
38. Z. Qi and I. Davidson. A principled and flexible framework for finding alternative clusterings. In *KDD '09: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 717–726, New York, NY, USA, 2009. ACM.
39. P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, 13(11):2498–2504, 2003.
40. S. Van Dongen. Graph clustering via a discrete uncoupling process. *SIAM J. Matrix Anal. A.*, 30(1):121–141, 2008.
41. T. Yamada, S. Goto, and M. Kanehisa. Extraction of phylogenetic network modules from prokayrote metabolic pathways. *Genome Inform.*, 15(1):249–258, 2004.
42. H. Yu, P. M. Kim, E. Sprecher, V. Trifonov, and M. Gerstein. The importance of bottlenecks in protein networks: correlation with gene essentiality and expression dynamics. *PLoS Comput. Biol.*, 3(4):e59, 2007.
43. W. W. Zachary. An information flow model for conflict and fission in small groups. *J. Anthropol. Res.*, 33:452–473, 1977.

GEOMETRIC EVOLUTIONARY DYNAMICS OF PROTEIN INTERACTION NETWORKS

NATAŠA PRŽULJ*, OLEKSII KUCHAIEV, ALEKSANDAR STEVANOVIĆ, and WAYNE HAYES

Department of Computer Science,
University of California, Irvine
CA, 92697-3425, USA
E-mail: natasha@ics.uci.edu

Understanding the evolution and structure of protein-protein interaction (PPI) networks is a central problem of systems biology. Since most processes in the cell are carried out by groups of proteins acting together, a theoretical model of how PPI networks develop based on duplications and mutations is an essential ingredient for understanding the complex wiring of the cell. Many different network models have been proposed, from those that follow power-law degree distributions and those that model complementarity of protein binding domains, to those that have geometric properties. Here, we introduce a new model for PPI network (and thus gene) evolution that produces well-fitting network models for currently available PPI networks. The model integrates geometric network properties with evolutionary dynamics of PPI network evolution.

Keywords: evolutionary dynamics, protein-protein interaction networks, geometric network model

1. Introduction

Understanding protein-protein interactions (PPIs) and the complex networks that they form is a central problem in systems biology. The crucial role that proteins play in most cellular processes justifies intense study of the complex wiring of their interactions. Note that the network of interactions may contain significant information that cannot be extracted solely from a study of protein sequences, because individual proteins rarely function alone. Instead they "cooperate" with other proteins, and the resulting complex networks of protein-protein interactions may provide information that cannot be ascertained from the study of sequences or even sequence comparison. Hence, in the post genomic era, there is considerable scientific value in understanding topological properties of PPI networks and the biological origin and meaning of those properties.

The mathematical foundations for studying PPI networks are graph theory and statistics. A PPI network is modeled as an undirected unweighted *graph* (also called a *network*) $G(V, E)$, where V is a set of nodes (proteins) and E is a set of edges (i.e., interactions between protein pairs). Since self-loops in E significantly complicate the analysis of graphs, we make the simplifying assumption that they are not allowed. Since subgraph isomorphism is NP-complete,[1] exact network comparisons are computationally infeasible. Thus, easily computable heuristics are used for comparing networks. These heuristics are commonly called *network properties* and can historically be divided into two groups: *local* and *global* properties. Global properties include the *degree distribution*, the *clustering coefficient*, the *average shortest path length*, and various forms of *network centralities*.[2,3] Local ones include *network motifs*[4–6] and *graphlets*,[7] both of which relate to the occurrence of small subgraphs in a larger graph. While motifs refer to subgraphs (not necessarily induced) that occur at an unusually high frequency, graphlets refer to *all* induced subgraphs regardless of their frequency. As such, graphlets provide a more thorough description of a network and allow their comparison, such as through *graphlet degree distribution agreement (GDD-agreement)*;[8,9] see section 2.2 for details. By using these properties to compare networks, well-fitting network models for biological networks have been proposed.[7,8,10,11] Also, network properties have been used to suggest protein function and involvement in disease.[12–16] Network models have further been exploited to guide biological experiments and discover new biological features.[17,18]

*Corresponding author.

The first attempts to model real-world networks began with the Erdös-Rényi (ER) random graph model.[19] This model has well-studied mathematical properties, although it is very simplistic. Erdös-Rényi random graphs have only two parameters: the number of nodes in the network and the probability p of an edge between any two nodes. The ER model poorly captures both global and local properties of PPI networks.[7,8] This implies that the structure of PPI networks is not completely random, presumably because evolution has imposed structural patterns that deserve close investigation.

Scale-free networks, popularized by Barabási and Albert, were proposed as a new and better model for early data sets of protein-protein interaction networks.[20] In these networks, degree distributions follow a power-law. The popularity of this model is explained by the fact that in many real-world networks (including PPI networks) a part of the degree distribution obeys a power-law. Barabási and Albert also proposed a *preferential attachment* model, which can generate networks with power-law degree distributions. The scale-free model, however, has several conceptual drawbacks. First, the degree distribution is not a sufficiently discriminative measure, since two networks can have exactly the same degree distribution, but completely different local structure. For example, a graph containing 100 triangles has exactly the same degree distribution as one containing a single 300-node cycle. In fact, the *gene duplication and mutation model* for generating networks with power-law degree distributions, proposed by Vaźquez et al.,[21] fits PPI networks much better than does the preferential attachment model, even though both generate graphs with power-law degree distributions. Furthermore, the systematic study of the fly's PPI network classified it as a duplication-mutation-complementation network instead of a preferential attachment one.[22] Thus, variants of the scale-free model of gene duplications and mutations have been proposed.[23–25] The second major drawback of the scale-free network model for PPI networks comes from the fact that currently available PPI datasets are both noisy and incomplete. It has been shown that subsets of scale-free networks are not scale-free[26,27] and therefore, since current PPI datasets have power-law degree distributions, it is not clear that complete and clean PPI networks are scale-free. Recently, *geometric random graphs*[28] were introduced as a better model of PPI networks; in these graphs, nodes correspond to points in space distributed uniformly and independently at random and edges exist between nodes corresponding to points that are close in space; see section 2.1 for details. These graphs have Poisson degree distributions, but there exist variants of geometric graphs in which this is not the case. In has been shown that geometric random graphs fit PPI networks much better than other commonly used network models despite the fact that parts of degree distributions of PPI networks often follow a power-law.[7,8,11,29]

In this paper we introduce a new model of PPI network evolution that results in networks that provide the best currently known fit to high confidence PPI networks. Since geometric *random* graphs seem to provide the best fit to the currently available PPI networks and since genomes have evolved through gene duplication and mutation events rather than at random, we bridge the concepts of network geometricity with the evolutionary dynamics. We introduce two new network models of *geometric gene duplication and mutation*, which utilize geometric graph principles to model the evolutionary dynamics of PPI networks.

2. Methods

It is important to distinguish between two conceptually different types of network models, which we refer to as *descriptive* and *network-driven* models. Descriptive models describe general properties of all networks of a particular type (e.g., PPI networks). For example, the scale-free model reproduces the power-law degree distribution (regardless of the exponent γ in the power law $P(k) \sim k^{-\gamma}$) which was observed in many, though not all, PPI datasets. Our new geometric gene duplication models and scale-free gene duplication[21] models are also descriptive because they model the principle (gene duplication and mutation) by which all PPI networks have evolved. A network-driven model, in contrast, tries to model a *particular* PPI network instance as well as possible. For example, trained geometric model ("geo-train"),[11] stickiness-index-based model ("sticky")[10] and Erdös-Rényi random graphs with the same degree distribution as data ("er_dd") require a *particular* network example and then try to reproduce its structure.

Since descriptive models do not need to be fitted to the particular network example they do not suffer

from over-fitting. On the other hand, network-driven models must have an example network to learn their parameters and therefore might over-fit the data. Hence, to avoid over-fitting, one should be careful and properly adjust the number of free parameters in such models using, for example, the Bayesian Information Criterion.[11] In this study, we show how the geometric graph framework can be used to create a well-fitting descriptive model of PPI networks. For a geometric network-driven model, see Kuchaiev and Pržulj (2009).[11]

2.1. *Geometric Gene Duplication and Mutation Models*

A *geometric random graph*[28] is a graph $G(V, E)$ with the set of nodes V distributed uniformly at random in some metric space. An edge exists between two nodes u and v if the distance between them $d(u, v)$ is less than ϵ, for some constant ϵ and some appropriate norm. The crucial parameters of this model are: the metric space, its dimensionality, and the distribution of nodes in that space. Intuitively, each protein can be described with its biochemical properties and therefore proteins reside in some multidimensional biochemical space. However, currently, it is hard even to hypothesize about the nature or dimensionality of that space. In this study, we focus on altering the distribution of the nodes in a low-dimensional Euclidean space in a way which simplistically models evolutionary dynamics of protein-protein interaction networks. Euclidean space is chosen just as a proof of concept. Note that highly dimensional spaces are less interesting, since if we allow enough dimensions, we can trivially embed a network into such a space. Thus it is encouraging to discover that only a few dimensions are required to accurately model this space geometrically,[7,8,11,29] indicating that geometricity is an important factor in modeling PPI networks. Furthermore, PPI networks can be directly embedded into a low dimensional space, in the sense that nodes sharing a graph-theoretic neighborhood can be placed close together in a geometric space in such a way that the resulting geometric graph is almost identical to the original.[29]

We introduce two geometric network models that incorporate the principles of gene duplications and mutations. Each of our models determines the principle by which the network is grown from an initial, small seed network. Growth is governed by adding new nodes intended to model gene duplications and mutations, moderated by natural selection as follows. A duplicated gene starts at the same point in biochemical space as its parent, and then "evolutionary optimization" acts either to eliminate one, or cause them to slowly separate in the biochemical space. This means that the child inherits some of the neighbors of its parent while possibly gaining novel connections as well. The further the "child" is moved away from its "parent," the more different their biochemical properties are. The randomness in the direction of the move models which subset of parent's properties (i.e. network interactions) will be preserved.

We refer to our two new models as *GEO-GD expansion* and *GEO-GD with a probability cutoff*. Each GEO-GD model network starts from a small initial *seed* network. For simplicity we use a 5-node clique, although we do not know what effect the exact makeup of the seed network will have on the final structure (this will be tested in a future paper). To create our seed networks, we place 5 nodes uniformly at random inside a sphere of radius $\frac{\epsilon}{2}$ in the embedding space. As a proof of concept, we use 3-dimensional Euclidean space for growing both of our GEO-GD models.

2.1.1. *GEO-GD Expansion Model*

Starting from the seed network, this model adds nodes iteratively, by choosing as the parent an existing node uniformly at random and placing a child node in a random direction at a randomly chosen distance of at most 2ϵ from the parent, where ϵ is the same parameter as was used in the definition of a geometric random graph. The movement at a distance less than ϵ allows the child to keep some of the parent's connections, whereas the movement at a distance of greater than ϵ allows the child to form a completely new set of connections. Thus, the child-node is adjacent to some of the neighbors of the parent-node, while at the same time potentially gains new interactions. We stop once we reach a predetermined number of nodes.

2.1.2. *GEO-GD with a Probability Cutoff Model*

This model is almost identical to the expansion model, except that the child can be duplicated in two different ways, rather than just one. In both cases, the child moves in a randomly chosen direction, but the two cases differ in the distance the child can move: with probability p, the child can move a maximum distance of ϵ, while with probability $1-p$ it can move up to 10ϵ; the second case is meant simply to model a large mutation rather than a small one.

Figure 1 presents some examples of distributions of points on the plane generated by our models. As it follows from this figure, the GEO-GD with a probability cutoff model generates networks with more pronounced clusters than GEO-GD expansion model, especially for higher values of parameter p. We only present points in the figure without edges, since edges would clutter the figure. Note that after n nodes were generated by *GEO-GD Expansion* or *GEO-GD with a Probability Cutoff* model, the resulting number of edges E' in the model network might be different from the number of edges in the data. Hence, in order to tune this number to our desired number of edges (i.e., to the number of edges of the PPI network that we are modeling) we have to rescale our initial ϵ by a small amount. If we denote by E the number of edges in the data, we first calculate the distances between all pairs of nodes in the model network, then we order them from the smallest to the largest and choose the E^{th} smallest distance as the re-scaled value $\epsilon^{'}$. Then we connect two nodes by an edge if they are closer than $\epsilon^{'}$ in the space. Since we rescale the ϵ in order to produce as many edges as in the data, the initial value of ϵ is not important.

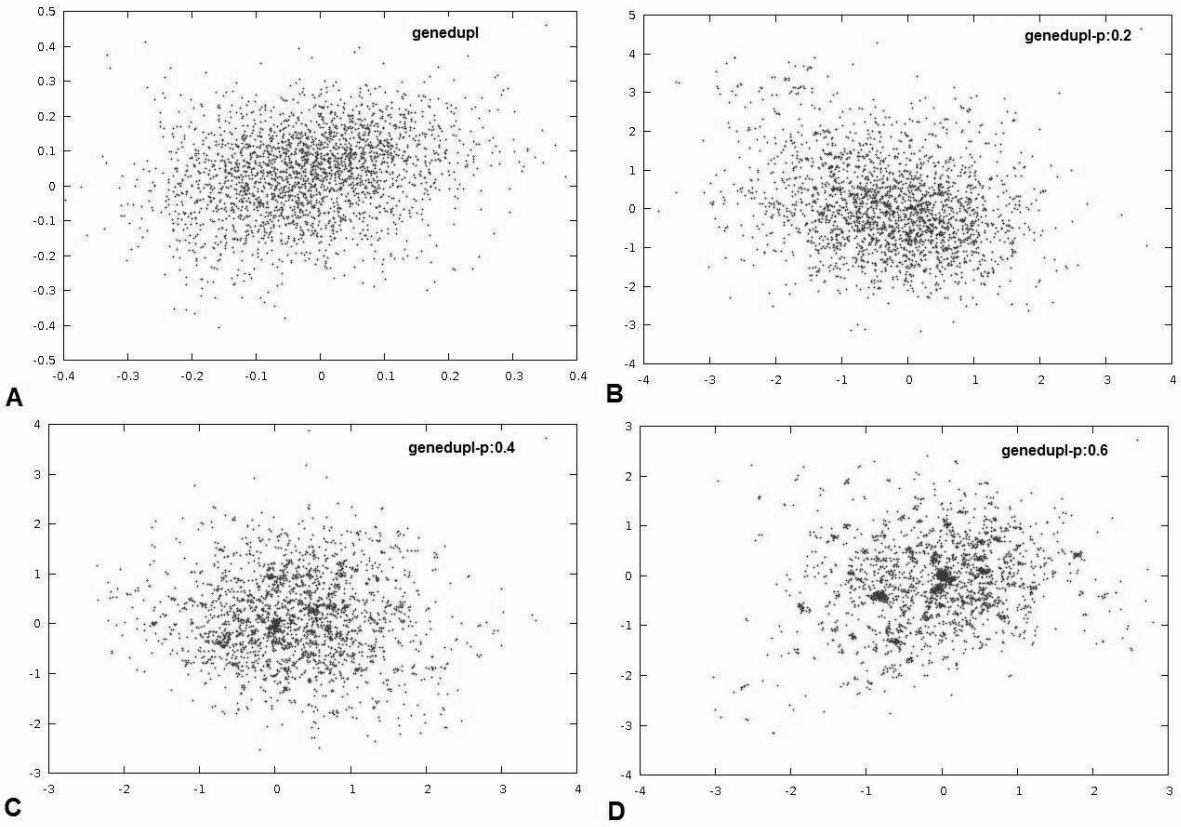


Fig. 1. Example distributions of points in the 2 dimensional Euclidean space, generated by (A) GEO-GD expansion model (genedupl), and GEO-GD with a probability cutoff model where (B) (genedupl-p:0.2) $p = 0.2$, (C) (genedupl-p:0.4) $p = 0.4$, and (D) (genedupl-p:0.6) $p = 0.6$ (genedupl-p:0.6). Here p is the probability of the move within ϵ distance from the parent node and $1 - p$ is the probability of the move within 10ϵ distance.

2.2. *Evaluation of the Models*

To evaluate the fit of the network model to the data, we need to compare the model networks to the PPI networks. As described in the Introduction, since large network comparisons are computationally infeasible, we examine the fit of local and global network properties of the model to the data. Since PPI networks are incompletely explored, global properties of such incomplete data are likely to be biased, or even misleading with respect to the currently unknown complete PPI networks. However, certain parts of these networks are very well studied (e.g., parts relevant for human disease). Thus, since we have detailed knowledge of certain local parts of PPI networks, but the data outside these well-studied parts are currently incomplete, global statistics are likely to provide misleading information about the PPI network as a whole, whereas local statistics are likely to be valid and meaningful. For these reasons, we focus on examining the similarity of networks by using our highly constraining measure of local network similarity, *graphlet degree distribution (GDD) agreement*.[8] Its formal definition is as follows.

First, a *graphlet* is a small, connected, induced subgraph of a network;[7] an *induced* subgraph with a node set $A \subseteq V$ of a graph $G(V, E)$ is obtained by taking A and all edges of G having both endpoints in A. There are 30 possible non-isomorphic graphlets on 2, 3, 4 and 5 nodes[7] (Figure 2; isomorphism is defined below). The GDD-agreement is a generalization of the degree distribution. Since the degree distribution $P(k)$ measures the number of nodes "touching" k edges and since an edge is the only 2-node graphlet, we generalize the degree distribution into the spectrum of distributions measuring the number of nodes "touching" k graphlets, for each of the 30 2-5-node graphlets. Clearly, the degree distribution is the first one in this spectrum.

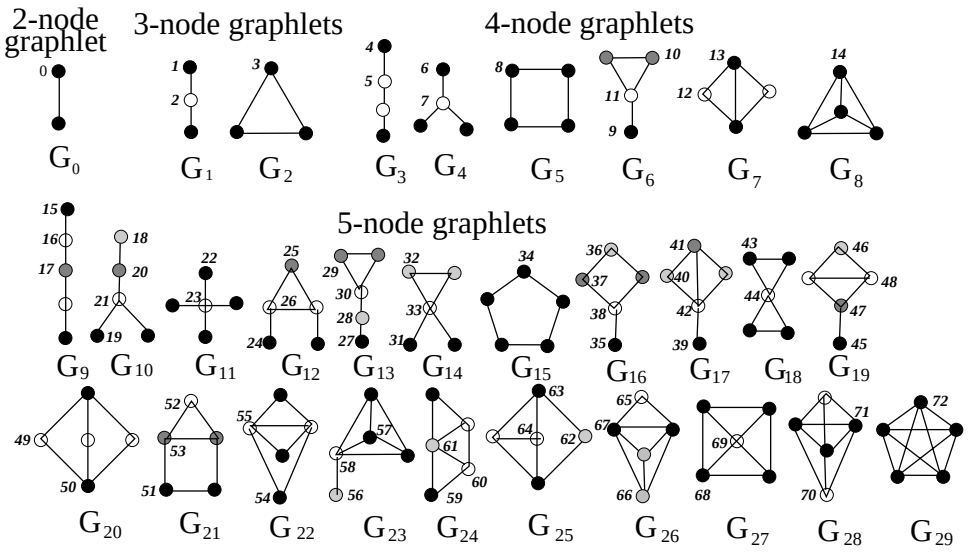


Fig. 2. The 30 2-5-node graphlets $G_0, G_1, G_2, \ldots, G_{29}$. In each graphlet, nodes belonging to a different automorphism orbit are of different shade. The 73 automorphism orbits of the 30 graphlets are labeled from 0 to 72. The figure is taken from Pržulj (2007).[8]

However, when we do this, we notice that while an edge was "symmetric," other graphlets might not be in the following sense. From a topological point of view, it is relevant to distinguish between *automorphism orbits* within each graphlet. For example, in a 3-node linear path (graphlet G_1 in Figure 2), the "end-nodes" are topologically identical, i.e., can be mapped to each other by an *automorphism*, an isomorphism of a graph with itself, where an *isomorphism* between two graphs is a bijection of their node sets that preserves their adjacency; the "middle node" of a 3-node path can only be mapped to itself by an automorphism. Therefore, a 3-node path has two different automorphism orbits. There are 73 automorphism orbits for the 30 2-5-node graphlets (Figure 2).[8] Thus, the *graphlet degree distribution (GDD)* is a 73-component distribution

of a network. Its j^{th} component, $d^j(k)$, is the sample distribution of the number of nodes in the network touching a particular graphlet k times at automorphism orbit j. The *GDD-agreement* is a similarity measure between graphlet degree distributions of two networks. It is a number between 0 and 1, meaning that two networks have similar GDDs if their GDD-agreement is close to 1, and otherwise, their GDDs are different (see Pržulj (2007)[8] for details). Note that GDD-agreement is a very strong measure of structural similarity of small-world networks (those with small diameters), since in these networks 5-node graphlets reach most of the network from each node. Since PPI networks are small-world, as demonstrated in Table 1 below, GDD-agreement based on up to 5-node graphlets is a strong measure of comparing their topologies to each other, or to model networks. We use our GraphCrunch software package[9] to calculate GDD-agreement and other network properties and evaluate the fit of model networks to the data.

3. Results

We test our model on four eukaryotic organisms whose PPI network data were produced using different biotechnologies, as well as deposited into curated databases. The organisms are baker's yeast *Saccharomyces cerevisiae*,[33–35] fruitfly *Drosophila Melanogaster*,[36,37] worm *Caenorhabditis elegans*[38,39] and human.[38,40–42] Table 1 shows the PPI networks that we analyze along with their sizes, some global network properties, and references the data originates from. YH1 and YH2 contain high-confidence parts of Collins *et al.*[33] and von Mering *et al.*[34] data sets that contain both binary interaction and co-complex data (see section 4.3 for details). Similarly, FH1 and FH2 contain high confidence parts of Giot *et al.*[36] and Finley *et al.*[37] fruitfly binary PPI data sets; FH2 is more recent and thus believed to be of higher quality than FH1. WE1 is the worm PPI network downloaded from BioGRID and WH1 is the high confidence binary PPI network from Simonis *et al.*[39] Human PPI network HE1 contains binary interactions from Rual *et al.*,[40] HH1 contains high-quality binary interactions from Venkatesan *et al.*,[42] while HE2 and HE3 are downloaded from BioGRID and HPRD, respectively. All of the networks have short average pathlenghts, i.e., they are small-world networks.

Table 1. PPI Networks that we analyze.

Network	Organism	Number of Nodes	Number of Edges	Avg Path-length	Cluste-ring Coef.	Reference (source of the data)
YH1	Yeast	1,622	9,074	5.53	0.55	Collins *et al.*[33]
YH2	Yeast	988	2,455	5.19	0.34	von Mering *et al.*[34]
YH3	Yeast	2,018	2,705	5.61	0.04	Yu *et al.*[35]
FH1	Fly	4,602	4,637	9.43	0.01	Giot *et al.*[36]
FH2	Fly	1,345	3,112	4.50	0.03	Finley *et al.*[37]
WE1	Worm	2,821	4,470	4.84	0.02	BioGRID (v2.0.51)[38]
WH1	Worm	2,528	3,706	5.32	0.02	Simonis *et al.*[39]
HH1	Human	235	239	4.53	0.00	Venkatesan *et al.*[42]
HE1	Human	1,873	3,463	4.34	0.03	Rual *et al.*[40]
HE2	Human	8,446	25,525	4.63	0.10	BioGRID (v2.0.51)[38]
HE3	Human	9,182	34,119	4.26	0.10	HPRD(v7)[41]

We compare the fit of our new models to PPI networks with the fit of other commonly used network models. The list of network models that we analyze is presented in Table 2. For each model, we evaluate the fit to the data to 30 random networks from the model that are of the same size as the data (have the same number of nodes and edges as the data). We report the averages and standard deviations of their fit in Figures 3, 4, 5, and 6 below. For our GEO-GD model with cutoff probability p, we vary p over all possible values between 0.1 and 0.9 in increments of 0.1 to determine p that yields the best fit. For our trained geometric model,[11] for each of the species, we trained it on the part of the species' PPI network

that is reported to be of high confidence. The scale-free gene duplication model that we analyze[21] (denoted by "vespdd:x," see Table 2) in addition to the number of nodes and edges has two probabilistic parameters p and q representing the probabilities of a child node to keep the parent's interactors and form new ones, respectively. Similar to what we do for our GEO-GD model to generate the best fitting model networks to the data, we vary p for vespdd:x from 0.3 to 0.7 in increments of 0.1 and for each p seek q using a binary search starting from $q = 0.5$ such that the resulting number of edges in the model network is within 1% of the number of edges in the data network (the number of nodes is the same as in the data). Figures 3, 4, 5 and 6 present GDD-agreements between the data and the model networks.

Table 2. Network models that we analyze.

Model Abbreviation	Network Model
er	Erdös-Rényi (ER) random graph model[19]
er_dd	ER model with the same degree distribution as the data[43]
geo	Geometric random graph model[7]
sf	Scale-free Barabási-Albert preferential attachment model[20]
sticky	Stickiness-index-based model[10]
geo-trained	Trained geometric model[11]
vespdd:x	Scale-free gene duplication model[21] with probability $p = x$ of child node keeping parent's interactors
genedupl	GEO-GD expansion model
genedupl-p:x	GEO-GD with a probability cutoff model, with cutoff probability $p = x$

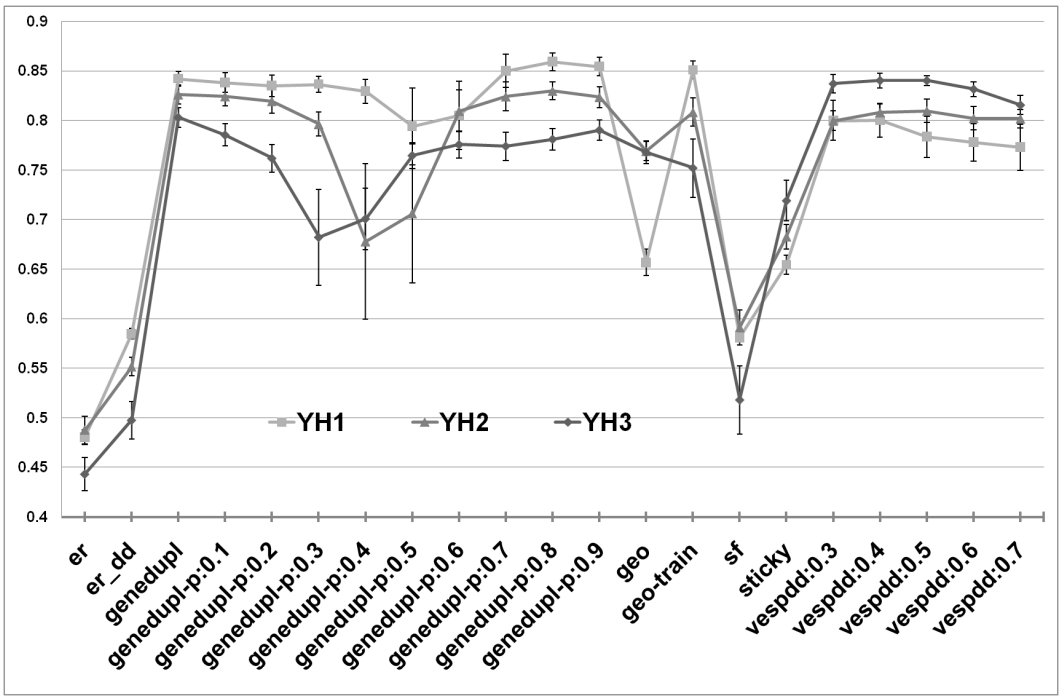


Fig. 3. GDD-agreement between yeast PPI and model networks. $x-$axis presents different model networks (see Table 2), $y-$axis presents the average values of GDD-agreements between the data and 30 model networks from each model. Lines with different point shapes correspond to different PPI networks (see Table 1); the error bar around a point is one standard deviation above and below the mean.

For yeast YH1 and YH2 PPI networks, our new GEO-GD model outperforms any other model, with

the best results being achieved by the GEO-GD model with the probability cutoff $p = 0.8$. For YH3 yeast network, vespdd:x model slightly outperforms any of the geometric models. Note that YH3 network is much sparser than YH1 and YH2 in terms of the number of edges and it has a much smaller clustering coefficient than the other two yeast networks (Table 1). Thus, it is possible that YH3 contains many false negatives, i.e., missing interactions and that vespdd:x model is better for modeling sparser networks. We comment on the types of interactions (binary versus co-complex) of these three networks in the Discussion section. Trained geometric model also provides a good fit to the data, but performs slightly worse than the best geometric and scale-free gene duplication models (Figure 3). Since yeast PPI networks are currently the most complete over all eukaryotic PPI networks, the superior fit of our GEO-GD model suggests that our model successfully captures the evolutionary dynamics of PPI networks.

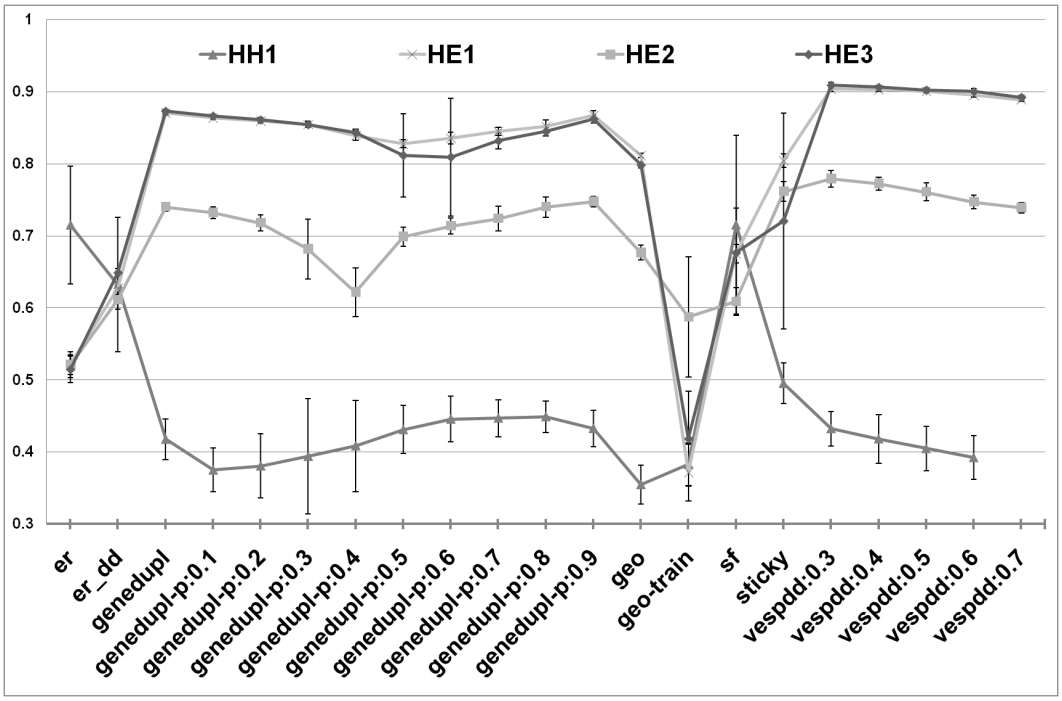


Fig. 4. GDD-agreement between human PPI and model networks. See the legend of Figure 3.

The situation is slightly different for PPI networks of human, fruitfly, and worm where scale-free gene duplication model slightly overperforms the geometric ones (Figures 4, 5 and 6). For human PPI networks, in all cases the best model is scale-free gene duplication model, except for HH1 network for which the best model is scale-free preferential attachment model (sf). Note however, that HH1 human network that is very small and sparse containing only 239 interactions between 235 proteins. Also, as mentioned above, all PPI networks of human are less complete than the PPI networks of yeast. For example, even though HH1 network contains only high-quality binary human PPIs (obtained by yeast-two-hybrid, Y2H, experiments), it is clearly an extremely small sample from the full human interactome. Also, since HH1 is extremely sparse, this indicates that many true interactions are likely to be missing. This is further corroborated by the fact that Erdös-Rényi random graphs provide the best fit to HH1 (Figure 4) and it is widely believed that topology of PPI networks is not random. Furthermore, the scale-free preferential attachment model (sf) provides as good of a fit to HH1 as Erdös-Rényi random graphs. This supports the hypothesis of the scale-free nature of incompleteness in the data.[8,26,27] Geometric gene duplication models (expansion and probability cutoff) have improved their fit over the geometric random graph model, only slightly doing worse than scale-free gene duplication model (we comment on the meaning of this slight underperformance in the Discussion section).

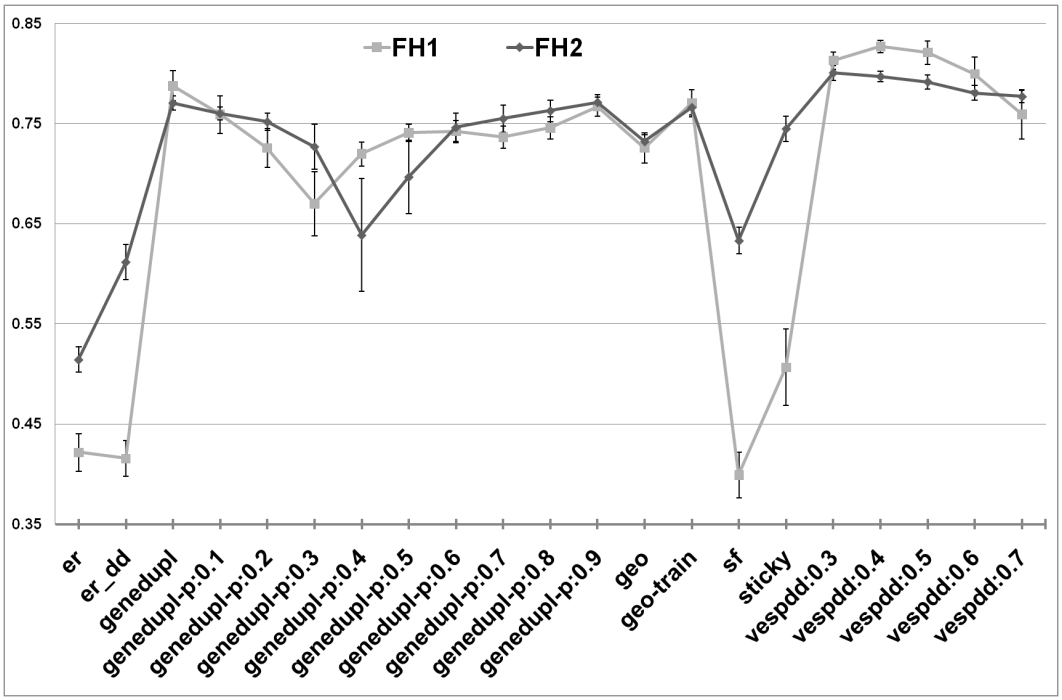


Fig. 5. GDD-agreement between fruitfly PPI and model networks. See the legend of Figure 3.

For fruitfly and worm PPI networks, scale-free gene duplication model slightly overperforms our geometric gene duplication models, while our gene duplication models outperform all other models (Figures 5 and 6).

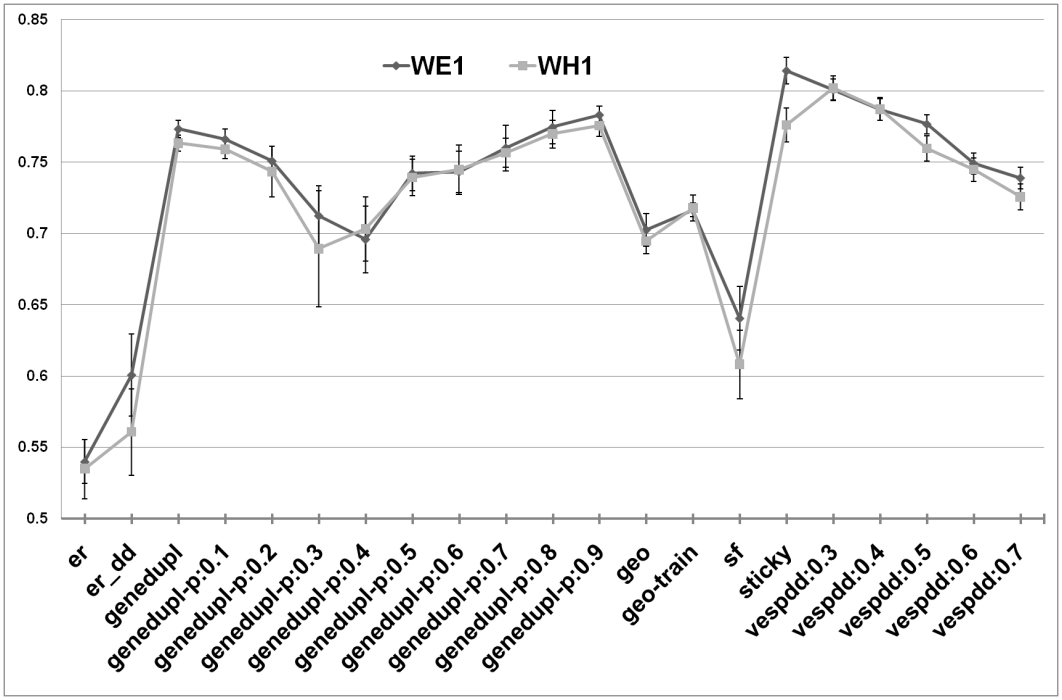


Fig. 6. GDD-agreement between worm PPI and model networks. See the legend of Figure 3.

4. Discussion

4.1. *Sensitivity of GDD-agreement*

Note that model networks generated with the same parameters that are coming from any random graph model that we analyzed have GDD-agreement of 0.86 ± 0.01.[8] Thus, since GDD-agreement of YH3 with our GEO-GD genedupl model (Table 2) is 0.8 and its GDD-agreement with the vespdd:x model is 0.84, the difference in the fit of these two models to the data falls within the sensitivity error of the GDD-agreement measure. Similar holds for the slight overperformace of our GEO-GD over vespdd:x for YH1 and YH2 networks. That is, both GEO-GDD and vespdd:x model provide the best possible fit to the yeast PPI networks that can be measured with the best currently available network comparison tool. Our contribution is not only in proposing an intuitive, geometric paradigm of evolutionary network dynamics, but also in designing such models with fewer parameters than similar scale-free-based models (see below).

4.2. *Parameters of Network Models*

An important characteristic of any model is its number of parameters. For any network model, we need at least two parameters (specified either explicitly or implicitly): the number of nodes and the number of edges in the network. More complicated models can have additional parameters. Clearly, among two models which fit the data equally well, the model with fewer parameters is better, since it provides a simpler description of the phenomena. The scale-free gene duplication model,[21] in addition to the number of nodes and edges, has two parameters p and q which are the probabilities of keeping old and forming new connections, respectively, for the duplicated node. In contrast, our new geometric gene duplication with a probability cutoff model has only one additional parameter, p, whereas geometric gene duplication expansion model does not have any parameters except of the number of nodes and edges in the network. As our experiments show, our new models fit denser high quality yeast PPI networks better than any other network model. Also, they perform on the networks of other species approximately the same as scale-free gene duplication model. The slightly better performance of a scale-free gene duplication model on human, fly and worm networks can be explained by the lower quality of these networks (compared to yeast networks) and the fact that this model has more parameters than our new models. Since our new geometric gene duplication model has fewer parameters, we can conclude that it is a better network model. Of course, one might argue that geometric graphs have an additional parameter, D, the dimensionality of the underlying Euclidean space. However, our previous research shows that the exact value of dimensionality is not important.[7,8,11,29] Instead, the important fact is that the PPI networks are well modeled by *low-dimensional* geometric graphs.

4.3. *Types of PPI Data*

We need to distinguish between PPI networks containing solely binary interactions (obtained by Y2H) and those containing co-complex data (obtained by mass spectrometry of purified complexes). Since the "spoke" and "matrix" models are used for co-complex PPIs, binary interaction networks are believed to have fewer false positives than networks containing co-complex data.[39,42] In the "spoke" model, edges exist between the bait and each of the preys, but not between the preys, while in the "matrix" model, a fully connected graph is formed between the bait and all preys. However, binary data still contain false negatives (missing interactions) due to technological limitations of Y2H (e.g., Y2H is not good at detecting membrane PPIs). The goal of our study is to provide a better model for high confidence PPI networks that are as complete as possible. Our new GEO-GD model outperforms any other model for yeast YH1 and YH2 networks, which are high confidence parts of the yeast interactome containing both binary and co-complex data, although vespdd:x model is a close contestant (Figure 3). YH3 network contains only binary interactions detected by high quality Y2H experiments. However, this network is extremely sparse, containing only 2,705 interactions amongst 2,018 proteins and therefore, it contains many false negatives. Nevertheless, the fit of our GEO-GD to YH3 network is remarkable. It is possible that if we used as the seed network a sparse graph, e.g. a 5-

node path, instead of a 5-node clique for growing our GEO-GD models, we would obtain even better fitting GEO-GD networks for binary interaction data.

Human PPI network HE1 consists solely of binary interactions. HH1 network consists only of high-quality binary interactions, but its level of false negatives (missing interactions) is very high, which potentially explains why simple Erdös-Rényi random graphs (er) and scale-free preferential attachment models (sf) are the best for this data set (Figure 4). HE2 and HE3 human networks were downloaded from BioGRID and HPRD and thus contain both types of PPIs, which makes them more complete, but with higher levels of false positives than in HE1 and HH1. Both of the fly PPI networks contain only binary PPIs and they can both be modeled indistinguishably well by geometric and scale-free gene duplication and mutation models (Figure 5). Worm WH1 network contains solely binary interactions, whereas WE1 contains both types of PPIs. They are both modeled well by geometric as well as by scale-free gene duplication and mutation models (Figure 6).

5. Conclusion

We have shown how the geometric graph framework can be used to model the principle of gene duplications and mutations by which all PPI networks have evolved. We demonstrated that our new descriptive network models of geometric evolutionary dynamics are well-fitting to the currently available PPI networks of eukaryotic organisms. The fact that geometric and scale-free gene duplication and mutation models always perform approximately the same and often outperform other models leads to the conclusion that it is not the power-law degree distribution of the currently available PPI datasets that is important, but the underlying processes of evolutionary dynamics that created these networks.

A mathematical model of any real-world phenomena has two ultimate goals: to provide better understanding of the phenomena and to allow practical applications. The scale-free models were used to describe the data; however their practical applications were limited to simply estimating the size of the interactomes. Geometric framework allows to work with network's nodes as with point in the metric space which is much more convenient from the mathematical point of view and can be used for such important practical applications as, for example, PPI network de-noising.[18] Due to their better fit to the data and mathematical convenience for the practical applications, we believe that geometric random graph model is the most promising framework for working with PPI networks.

Acknowledgments

This project was supported by the NSF CAREER grant IIS-0644424.

References

1. S. A. Cook, *Proc. 3rd Ann. ACM Symp. on Theory of Computing*, 151-158 (1971).
2. M. E. J. Newman, *SIAM Review*, **45**, 167-256 (2003).
3. M. E. J. Newman, in *The New Palgrave Encyclopedia of Economics*, L. E. Blume and S. N. Darlauf (eds.), **2**nd edition (2008).
4. R. Milo, S. S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, *Science*, **298**, 824-827 (2002).
5. S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon, *Nature Genetics*, **31**, 64-68 (2002).
6. R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, and U. Alon, *Science*, **303**, 1538-1542 (2004).
7. N. Pržulj, D. G. Corneil, and I. Jurisica, *Bioinformatics* **20**, 3508-3515 (2004).
8. N. Pržulj, *Proceedings of the 2006 European Conference on Computational Biology (ECCB'06), Bioinformatics*, **23**, e177-e183 (2007).
9. T. Milenković, J. Lai, and N. Pržulj, *BMC Bioinformatics*, **9**, 70 (2008).
10. N. Pržulj and D. J. Higham, *Journal of the Royal Society Interface*, **3**, 10 (2006).
11. O. Kuchaiev and N. Pržulj, *Proceedings of the 2009 Pacific Symposium on Biocomputing*, 39-50 (2009).
12. R. Sharan, I. Ulitsky, I. and R. Shamir, *Molecular Systems Biology*, **3:88** (2007).
13. T. Milenković and N. Pržulj, *Cancer Informatics*, **2008:6**, 257-273 (2008).

14. C. Guerrero, T. and Milenković, N. and Pržulj, J. J. Jones, P. Kaiser, and L. Huang, *PNAS*, **105**, 13333-13338 (2008).
15. T. Milenković, V. Memišević, A. K. Ganesan, and N. Pržulj, *Journal of the Royal Society Interface*, to appear (2009).
16. R. Aragues, C. Sander, and B. Oliva, *BMC Bioinformatics*, **9:172** (2008).
17. M. Lappe and L. Holm, *Nature Biotechnology*, **22**, 98-103 (2004).
18. O. Kuchaiev, M. Rašajski, D. J. Higham, N. Pržulj, *PLoS Computational Biology*, **5(8)**, e1000454 (2009).
19. P. Erdös, and A. Rényi, *Publ. Math.* **6**, 290-297 (1956).
20. A.L. Barabasi and R. Albert, *Science* **286**, 509-512 (1999).
21. A. Vaźquez, A. Flamminia, A Maritana, A. Vespignani, *Complexus*, **1**, 38-44 (2003).
22. Manuel Middendorf, Etay Ziv, Chris H. Wiggins , *PNAS* **102:9**, 3192-3197 (2005).
23. R. V. Sole, R. Pastor-Satorras, E. Smith, T. Kepler, *Adv. Complex Syst.*, **5**, 43-54 (2002).
24. A. Wagner, *Proc. R. Soc. London B* **270**, 457-466 (2003).
25. F. Chung, L. Lu, T. Dewey, D. Galas, *J. Comput. Biol. 10*, **5**, 677-688 (2003).
26. M.P.H. Stumpf, C. Wiuf, and R.M. May, *PNAS* **102**, 4221-4224 (2005).
27. J. D. H. Han, D. Dupuy, M. Bertin, M. E. Cusick, M. and Vidal, *Nature Biotechnology*, **23**, 839-844 (2005).
28. M. Penrose. Geometric Random Graphs, *Oxford University Press* (2003).
29. D. J. Higham, M. Rašajski, and N. Pržulj, *Bioinformatics*, **24**, 8 (2008).
30. R. Albert and A. Barabási, *Reviews of Modern Physics*, **74**, 47-97 (2002).
31. Barabási AL, Oltvai ZN, *Nat Rev Genet.*, **2**, 5 (2004).
32. D.J. Watts, S.H. Strogatz, *Nature*, **393**, **6684**, 440-442 (1998).
33. S. R. Collins, P. Kemmeren, X. C. Zhao, J. F. Greenblatt, F. Spencer, F. C. Holstege, J. S. Weissman, and N. J. Krogan, *Molecular and Cellular Proteomics*, **6(3)**, 439-450 (2007).
34. C. von Mering, R. Krause, B. Snel, M. Cornell, S. G. Oliver, S. Fields and P. Bork, *Nature*, **417**, 399-403 (2002).
35. H. Yu *et al.*, *Science*, **322**, 104-110 (2008).
36. L. Giot *et al.*, *Science*, **302**, 1727-1736 (2003).
37. J. Yu, S. Pacifico, G. Liu, R. L. Finley Jr., *BMC Genomics*, **9**, 461 (2008).
38. C. Stark C, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, amd M. Tyers, *Nucleic Acids Res*, **34**(Database issue), D535-D539 (2006).
39. N. Simonis *et al.*, *Nature Methods*, **6**, 47-54 (2009).
40. J.-F. Rual *et al.*, *Nature*, **437**, 1173-1178 (2005).
41. Prasad et al., *Nucleic Acids Research*, **37**, D767-D772 (2009).
42. K. Venkatesan *et al.*, *Nature Methods*, **6**, 83-90 (2009).
43. M. Molloy and B. Reed, *Random Structures and Algorithms*, **6**, 161-180 (1995).

THE STEADY STATES AND DYNAMICS OF UROKINASE-MEDIATED PLASMIN ACTIVATION

LAKSHMI VENKATRAMAN[1, 2], HANRY YU [2, 3], SOURAV S. BHOWMICK[1, 2], FORBES DEWEY JR.[2, 4], LISA TUCKER-KELLOGG[2, 5, *]

[1]*School of Computer Engineering, Nanyang Technological University, Singapore*
[2]*Singapore-MIT Alliance, E4-04-10, 4 Engineering Drive 3, Singapore 117576*
[3] *Department of Physiology, National University of Singapore, Singapore*
[4]*Fluid Mechanics Laboratory, Department of Mechanical Engineering, M.I.T, Cambridge, MA 02130*
[5]*Department of Computer Science, National University of Singapore, Singapore 117417*
** Correspondence to tucker@comp.nus.edu.sg*

Plasmin and urokinase-type plasminogen activator (uPA) are ubiquitous proteases regulating the extracellular environment. They can activate each other via proteolytic cleavage, suggesting the potential for complex dynamic behaviors that could be elucidated by computational modeling. Ordinary differential equations are constructed to model the activation dynamics of plasminogen into plasmin, and single-chain uPA (scUPA) into two-chain uPA (tcUPA). Computational simulations and phase plane analysis reveal two stable steady states for the activation of each protein. Bifurcation analysis shows the *in silico* system to be bistable. Cell-free experiments verify the system to have ultrasensitive activation behavior, where scUPA is the stimulus and plasmin the output. Furthermore, two significantly different steady states could be seen *in vitro* for the same stimulus levels, depending on the initial activation level of the plasmin. The switch-like dynamics of the uPA-plasmin system could have potential relevance to many normal and disease processes including angiogenesis, migration and metastasis, wound healing and fibrosis.

Keywords: Urokinase-Type Plasminogen Activator, Computational modeling, Nonlinear Dynamics

1. INTRODUCTION

Mathematical modeling of molecular interaction kinetics can give insight into dynamic characteristics and time-dependent functions of molecular networks [1-3]. The nonlinearity inherent in such networks can cause dynamical effects that play a transformative role in converting signals into biological functions. Qualitative changes in the outcome of pathway behavior, called bifurcations, arise from nonlinearity of interaction networks and are dependent on the parameter values. Analysis of transitions in system outcome due to changing parameters is called bifurcation analysis [4, 5]. Bifurcation analysis has been used on biological models like the cell-cycle [10, 11]. A common bifurcation in biology is bistability, or existence of two steady states, which transforms a gradual input change into an "all-or-none" switch. Bistability can explain the switch-like nature of apoptosis [6-9] and cell-cycle progression [10-13].

Bistability is most often studied in reversible pathways, especially those involving phosphatases and kinases [14]. Proteases, well known for their importance in the homeostasis of the extracellular matrix (ECM), are enzymes that catalyze the irreversible cleavage of a protein backbone. Some previous work has examined the systems-level dynamics of protease networks [15]. Plasmin (PLS) is a ubiquitous serine protease activated from the secreted protein plasminogen (PLG). Irreversible conversion of PLG to PLS is facilitated by plasminogen activators (PA) which nick at the Arg^{560}-Val^{561} bond of PLG to release the active PLS protease [16]. Tissue Plasminogen Activator (tPA) mediates PLG activation in connective tissues, while urokinase Plasminogen Activator (uPA) mediates PLG activation in the tissue context [17].

PLS is crucial in haemostasis and blood clotting where it converts inactive fibrinogen to fibrin, causing degradation of clots [18]. In tumor angiogenesis, PLS has been confirmed as a pro-angiogenic activator causing dissolution of ECM components to allow for development of new blood tissues [19]. In wound healing, PLS contributes to remodeling of injured/wounded ECM by activating the proteases that dissolve scars [20]. In addition, PLS can activate TGF-β1 from its inactive latent LTGF-β1 form [7, 8, 14] and TGF-β1 is an essential factor in the production of the ECM.

In this paper, we use mathematical modeling to investigate the dynamics of PLS activation by urokinase, and our modeling shows the system to be bistable. The model and its construction are described in section 2. Simulations and bifurcation analysis of the model follow in section 3. Finally in Section 4 we show experimental results that validate the mathematical predictions, confirming that uPA-mediated PLS activation is ultrasensitive and bistable.

2. MODEL CONSTRUCTION

Although an integral part of many pathways, proteases have not been as extensively modeled and studied as their kinase-phosphatase counterparts. A protease reaction is generally irreversible, as the cleaved fragments of the substrate diffuse apart and/or the substrate is consumed. Proteases are usually broad-spectrum reactors, meaning they are capable of cleaving different protein substrates having similar target sequences. The simple, classical manner of protease activation is auto-activation (Figure 1a), in which an active protease X cleaves its inactive precursor form. An alternative manner of activation (Figure 1b) operates via regulation of an intermediate regulatory enzyme (Y), which also has active and precursor forms. This is a common type of positive feedback imposed by the activated protease and is seen in pathways such as caspase activation, MMP activation and blood clotting. Figure 1c displays a variant in which the "inactive" precursor Y has some low level of catalytic activity which can by itself initiate activation of the protease.

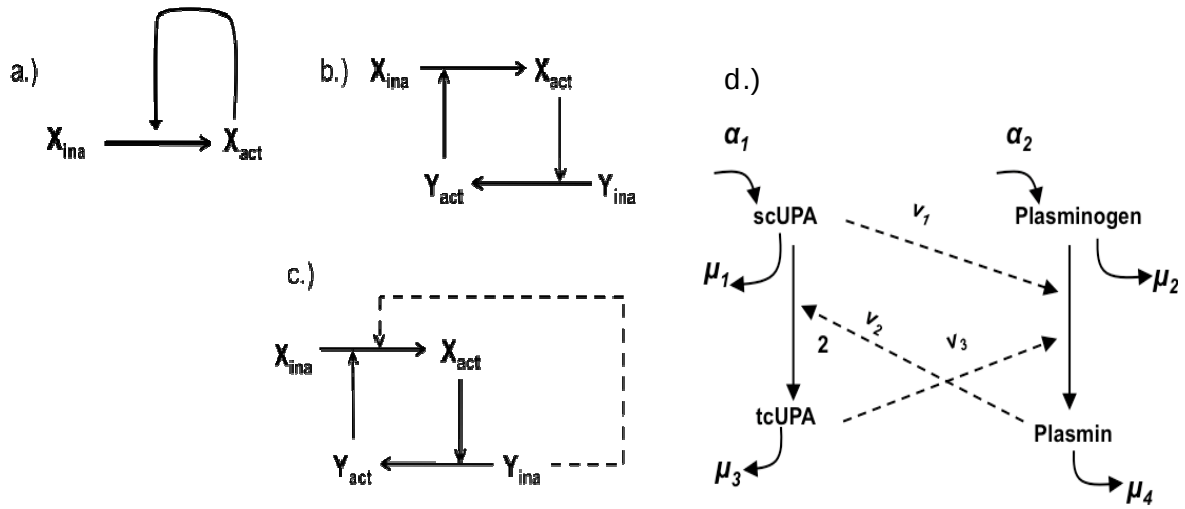


Table 1- Ordinary Differential Equations (ODEs)

ODEs	Reaction Rates
$d[scUPA]/dt=-v_2+\alpha_1-\mu_1*[scUPA]$	$v_1=keff_1*[scUPA]*[PLG]$
$d[PLG]/dt=-v_1-v_3+\alpha_2-\mu_2*[PLG]$	$v_2=keff_2*[scUPA]*[PLS]^n$
$d[PLS]/dt=+v_1+v_3-\mu_4*[PLS]$	$v_3=keff_3*[tcUPA]*[PLG]$
$d[tcUPA]/dt=v_2-\mu_3*[tcUPA]$	

Table 2- Parameter values

Parameters	Values
$keff_1$	0.0017 $\mu M^{-1}min^{-1}$
$keff_2$	1 $\mu M^{-1}min^{-1}$
$keff_3$	0.03 $\mu M^{-1}min^{-1}$
n	3
$\mu_1=\mu_3$	0.0001 min^{-1}
$\mu_2=\mu_4$	0.001 min^{-1}
α_1	0.00009 $\mu M\ min^{-1}$
α_2	0.001 $\mu M\ min^{-1}$

Figure1. Model Construction. a) Active Protease (X_{act}) activating itself from inactive form (X_{ina}). b) Protease feedback through intermediate regulatory enzyme Y. c) Initiator inactive protease (Y_{ina}) activating X. d) Actual model of PLS activation from PLG mediated by uPA. Table 1 showing the ODEs and the reaction rates. μ (1,2,3 and 4) represent degradation of respective species and α (1 and 2) represent production rates of scUPA and PLG respectively. Table 2 lists the normalized parameter values used for modeling.

uPA mediated PLS activation follows the mechanism of Figure 1c with some additional considerations (Figure 1d, and Table 1). UPA is secreted as a single chain form (scUPA) having very little intrinsic activity, but it can cleave the Glu-PLG form of plasminogen to produce PLS[16]. PLS in turn cleaves scUPA into the two-chain form, tcUPA, activating it completely by nicking at the Lys^{158}-Ile^{159} bond. tcUPA has 12-fold greater enzymatic activity for PLG than the scUPA form [16] . tcUPA creates positive feedback (Figure 1d) by cleaving PLG to form more PLS. Being the completely activated form of scUPA, tcUPA has more reactivity to PLG than scUPA. PLS activity has been modeled empirically as a co-operative process with a Hill coefficient (n), similar to reference 21. Substrate competition can be a cause for co-operativity in enzyme action, apart from the traditional mode of allosteric co-operativity [21]. PLS being a broad substrate enzyme could thereby exhibit co-operativity in its activity towards scUPA in the ECM. Production and degradation terms are important in this network (figure 1d) because, for example, if some production process does not occur to balance degradation, then any amount of degradation would eventually cause the system to decay to a single steady state at zero.

Extensive experimental literature on PLS and uPA provides narrow ranges for most rate constants in this model [16, 22-24]. Some parameter uncertainty remains and is examined directly in Sections 3.3 and 3.4. The numerical parameters we chose for closer study, explained in Appendix A and listed in Table 2 of Figure 1, represent a plausible qualitative model, but not an absolute quantification of all phenomena. Our purpose is to use modeling for elucidating possible behaviors, generating hypotheses, and directing experimental design, but we do not draw conclusions from the modeling alone.

3. MODEL SIMULATION

3.1 Steady state behavior of PLS

To understand the behavior of the system, we simulate the PLS-uPA model from Section 2. Using random initial conditions of all the species (Figure 2a), we follow the time progression curves of PLS. Interestingly, we notice that, depending on initial concentrations, PLS can attain two different steady states: a lower steady state at 0.02 μM and a higher one at 0.27μM. Usually reaction systems tend to converge to one steady state, and the presence of two steady states suggests bistability. The steady state behavior of PLS, computed in response to different concentrations of the initiator protease, scUPA, shows a sharp change in PLS steady state levels, caused by very little change in scUPA concentration (between 0.2-0.25 μM, Figure 2b). A system with such high sensitivity to parameter values is called "ultrasensitive" [14].

Since activation is irreversible, we note that turnover would be necessary for the system to be able to switch from activated to inactivated steady states. Indeed, changes in the production rate (which might reasonably occur *in vivo* when different cell types proliferate or die) can cause the system to switch up or down. A time-progression plot (Figure 2c) of PLS, responding to stepwise changes in the scUPA production rate, shows the dependence of PLS steady state on small changes in scUPA production. In this case, scUPA production less than 9.4 μM min^{-1} keeps PLS in its lower steady state; while a production rate of 9.7μM min^{-1} is sufficient to maintain a steady state with higher activation. A variety of other system parameters are capable of causing similar switches in the steady state (not shown).

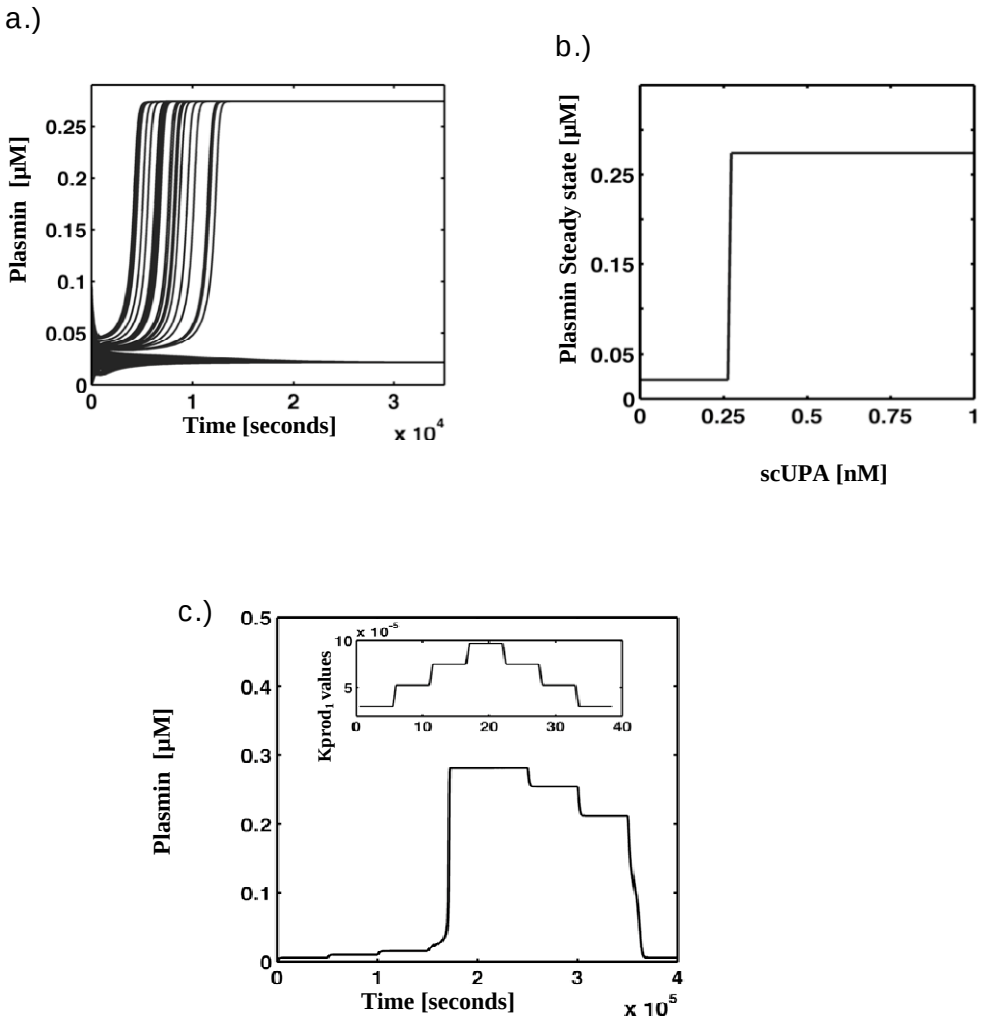


Figure 2. Steady state simulations. a) PLS time progression with random initial concentrations. b) Effect of the production rate of scUPA (α_1) on the time progression of PLS. c) Steady State of PLS for different initial concentrations of scUPA.

3.2 Phase plane Analysis

To investigate the steady states of the system, we used phase plane analysis which projects the full model onto 2 variables, for visualizing the essential dynamics of a minimized representative system. The reduction of the system, shown fully in Appendix B, decreases the number of degrees of freedom in the system by assuming some quantities to have zero derivative, as if at steady state. The sets of points where the remaining species are also at steady state are called *nullclines.* Intersections of nullclines are fixed points (equilibrium points) of the overall system. After the model reduction of Appendix B, the nullclines of the system are as follows:

$$pls\_n = \frac{a * TP}{D_2 + a} \quad (1)$$

$$tcupa\_n = \frac{(pls)^n * keff_2 * TU}{keff_2 * (pls)^n + D_1} \quad (2)$$

The nullclines are plotted in Figure 3, where the tcUPA nullcline is a dotted light grey, and PLS is a solid line. These intersect at three points labeled **A**, **B** and **C**. The solutions of the ODEs are plotted as tiny arrows across the plane, pointing in the direction of time evolution. The trajectory arrows converge towards points **A** and **B,** indicating that **A** and **B** are stable states. **C** is an unstable steady state because trajectories near **C** move the system either towards **A** or **B.** Eigen-value analysis confirms the stability, as points **A** and **B** have real negative Eigen-values while point **C** has positive Eigen-values. The system therefore exhibits two steady states and is "bistable."

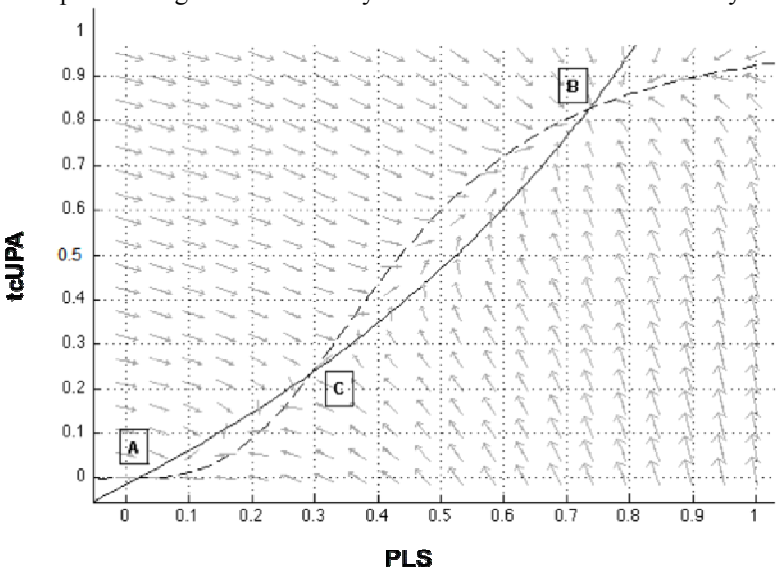


Figure 3: Nullclines for the two species tcUPA (dotted line) and PLS (solid line). The points where nullclines intersect are the steady states of the system. A and B are stable steady states, while C is unstable. Trajectories are shown as smaller arrows

3.3 Bifurcation analysis

The steady state behavior of the model seen in Sections 3.1 and 3.2 is dependent on parameter values. In order to understand how PLS makes the transition from the lower steady state to a higher one, we did bifurcation analysis. Bifurcation analysis relates parameter change with the system response. Figure 4a shows the effect on PLS steady state of changing $Keff_2$, the PLS efficiency parameter. Values of $Keff_2$ between 0.1-1.3 $\mu M^{-1}min^{-1}$ allow PLS to exist in two different steady states (Figure 4a), depending on initial conditions. Starting from an initial higher PLS concentration (grey curve, "Going Down" of Figure 4a), allows the system to achieve the higher steady state over a wider range of $Keff_2$ values before jumping down at $Keff_2 = 0.1\ \mu M^{-1}min^{-1}$.

Starting the system with a lower initial concentration of PLS (black curve, "Going Up", of Figure 4a), shifts into a higher steady state at a different threshold, 1.3 $\mu M^{-1}min^{-1}$ of $Keff_2$. Such systems exhibiting different thresholds for switching between two different steady states are called "hysteretic"[5]. Figure 4b is a bifurcation diagram with $Keff_3$, the tcUPA efficiency rate, as the bifurcation parameter. For $Keff_3 \leq 0.02\ \mu M^{-1}min^{-1}$, the PLS steady state is low, while at $Keff_3$ values higher than 0.053 $\mu M^{-1}min^{-1}$, PLS reaches the higher steady state. The dotted line in Figure 4b describes unstable equilibrium states, to which the system will not converge. Changes in two parameters $Keff_3$ and μ_2 [a degradation parameter, see Appendix B] yield a two-parameter bifurcation diagram (Figure 4c), where the area within the shaded cusp represents configurations with 3 fixed points or bistability, and the areas outside the cusp represent monostable regions.

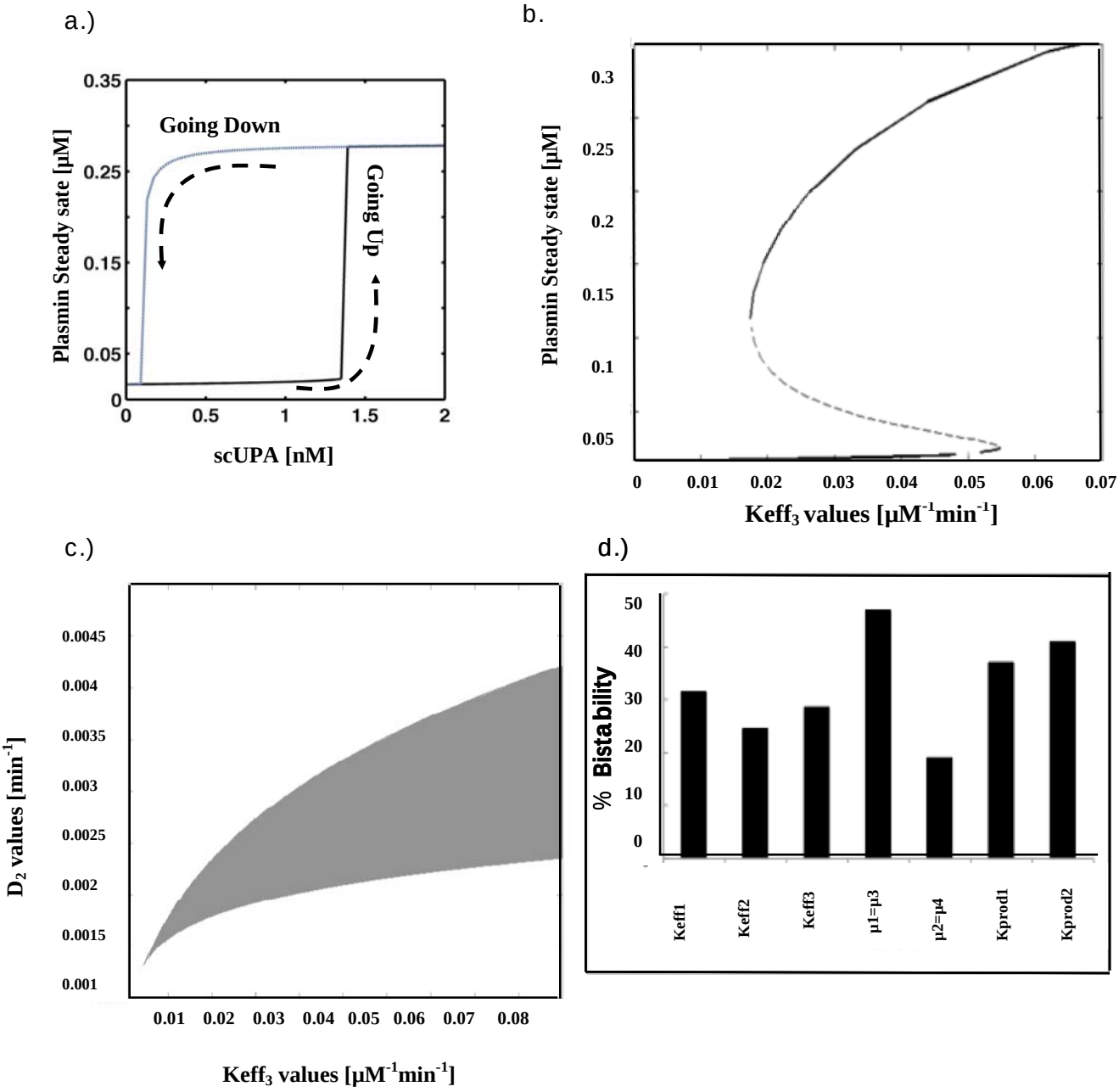


Figure 4: Bifurcation analysis: a) Change in steady state of PLS in μM with change in the $Keff_2$ parameter. b) Bifurcation diagram of PLS with changing the $Keff_3$ parameter. The dashed black line indicates unsteady states. c) 2-Parameter bifurcation diagram with the dark region being bistable. $\mu_2=\mu_3=D_2$. d) Bifurcation parameter robustness.

3.4 Parameter robustness

The bifurcation diagram results indicate a range of parameters over which the system can be bistable. Parameters in a biological system are never known with high confidence, and there are often differences in the same phenomenon between different *in vivo* models. We did a robustness analysis for the presence of bistable behavior in this system over a broad range of all the parameters (20% on either side of reference) [9] and noted the number of parameter sets which are capable of being bistable using a hysteretic technique similar to [25]. Briefly, each parameter was varied +/-20% of its reference value, keeping the other parameters constant, and each was checked for the presence of hysteretic behavior, i.e. a different threshold in the two steady states of the system. As shown in Figure 4d, >=30% of the varied parameter population is still capable of inducing a bistable response in the system.

4. EXPERIMENTAL EVIDENCE

We measured the dynamic behavior of PLS, in an isolated *in-vitro* cell free system. scUPA and PLG are the inputs to the system, which were provided as initial concentrations and also in the form of constant production.

For testing the ultrasensitive nature of the model, we measured PLS steady states after providing different initial concentrations [26]. Variable amounts of scUPA between 0.1nM and 6nM were given, along with a non-variable initial concentration of 1μM PLG. During the progression of the experiments, slight amounts of scUPA and PLG were added, at rates of 50pM/min and 1nM/min, respectively. PLS was monitored using its substrate s-2251, which can be measured as absorbance at 405nm. Figure 5a shows a time profile of PLS activity, which achieves steady state 4 hrs after being initiated with 0.5nM of scUPA. Figure 5b shows the steady state levels of PLS activity in response to variable initial concentrations of scUPA. When initial scUPA levels were 0.9nM or lower, the PLS steady state activity level was low at 0.24 O.D. scUPA levels of 2nM or higher resulted in PLS achieving steady state at a high level (0.95 O.D). Between 0.9 and 2.0nM, the system made a sudden transition in the steady state of PLS. This is a sigmoidal curve, indicative of ultrasensitive behavior, as opposed to a hyperbolic curve typical for Michaelis-Menten reactions [26].

Bistability was verified using the "going-up" and "coming-down" method [27] for observing hysteresis. The "going-up" experiments were initiated with PLS and PLG concentrations in the ratio 60%PLG: 40%PLS (0.6μM PLG: 0.4μM PLS). Varying initial concentrations of scUPA were added, and PLS steady state was monitored. For the "coming-down" experiments, the ratio was reversed (0.4μM PLG: 0.6μM PLS) and parallel values of scUPA were used. Thus we varied the initial activation state of the PLS without varying the absolute amount of the protein. If the system were indeed monostable, then the system would converge to the same steady state of PLS activation, irrespective of the initial activation ratio. As seen in Figure 5c, intermediate concentrations between 0.7nM and 4nM scUPA exhibited two steady states of PLS activation, depending on the initial activation ratio.

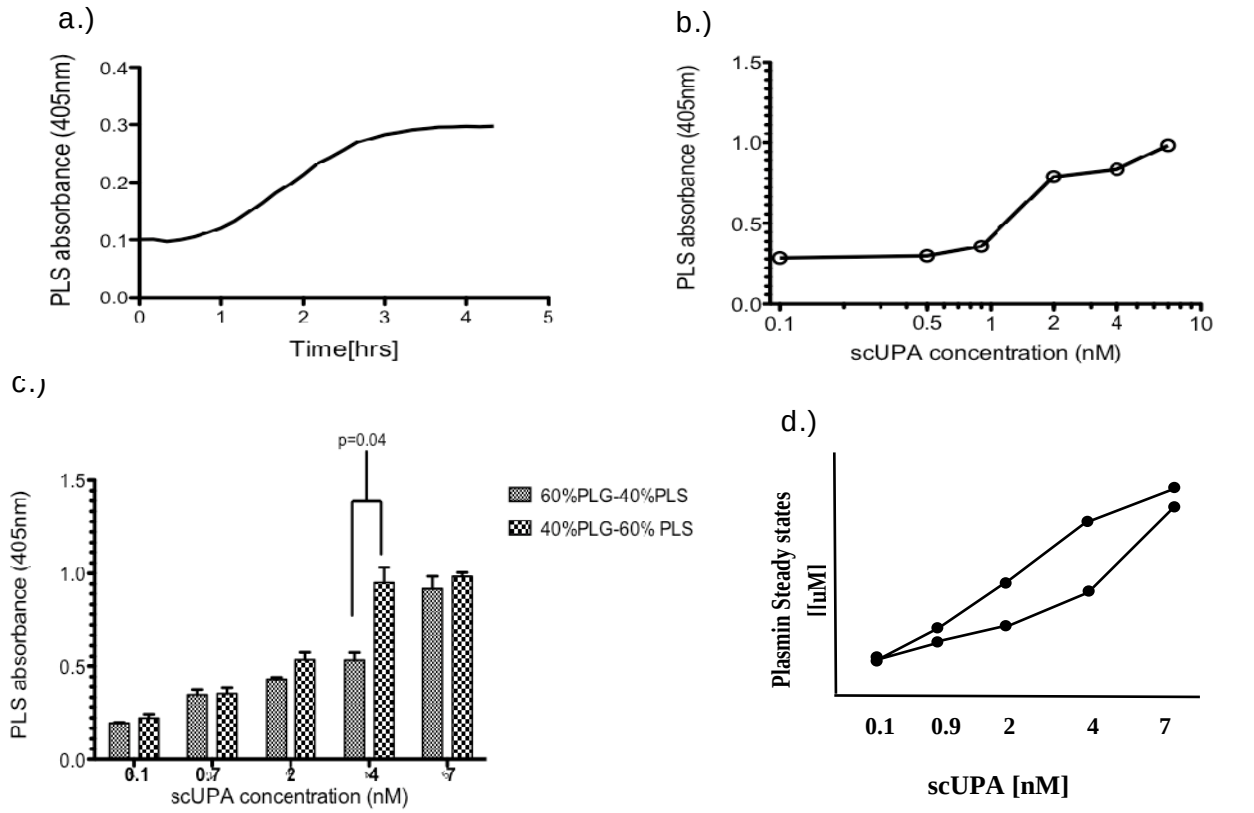


Figure 5-Experimental validation: - a) Time progression curve of PLS achieving steady state after an initial scUPA concentration of 0.5nM .b) Steady state of PLS activity with different scUPA initial concentrations. Steady state levels of PLS activity after initialization with different activation ratios, c) plotted showing the significance of the bistability at 4uM scUPA, and d) plotted in the style of a "going-up" and "coming-down" diagram.

DISCUSSION

In this paper, we employed computational and experimental methods to study the activation behavior of uPA and PLS, a pair of proteases that are crucial to regulating extracellular environments. Our work describes PLS as the system output, but there are significant downstream biological implications for uPA as well, so that uPA activation should also be viewed as an output. Both enzymes are synthesized in relatively inactive forms and both must be cleaved to be activated. They cleave each other in a positive feedback loop, so that any significant accumulation of either enzyme is sufficient to switch both towards their activated forms. Simulations show this arrangement is capable of generating behavior that is bistable as well as ultrasensitive. Simulations also show the importance of turnover. Because degradation causes removal of all forms, but influx affects only the uncleaved forms, the synthesis and degradation rates play an important role in determining the steady state. For example, changes in the production rate are alone sufficient to induce the ultrasensitive change in output (Figure 2b). We verified that the PLS activation dynamics resemble the simulations as follows: small changes in scUPA concentration were capable of causing large absolute increases in PLS activity; and PLS can exhibit two different steady states for the same amount of scUPA protein, depending on the initial activation state of the PLS.

PLS, PLG and urokinase (uPA) are influential in many physiological processes. Our discovery that they can exhibit bistable activation dynamics could have important repercussions for a number of normal and disease processes such as angiogenesis, tumor metastasis, wound healing and fibrosis. Angiogenesis, the process of growing new blood vessels, is tightly regulated by a variety of pro- and anti-angiogenic factors[19]. Angiogenesis occurs as a switch-like decision [28, 29], the mechanism of which is not known. uPA and PLS play crucial roles in angiogenesis [29], and if the PLS-uPA system can exhibit bistable switching, this could contribute to the switch-like behavior of angiogenesis.

Normal cell migration and pathological cell invasion require coordination of proteases and other extracellular factors to remodel the extracellular matrix (ECM). PLS and uPA act directly on the ECM [20], and PLS also regulates many members of the canonical ECM-regulating family, the matrix metalloproteinases (MMPs). The potential of PLS and uPA to switch between two stable levels of activation may have implications for cell migration, such as coordinating diverse, gradual signals into a synchronized cellular function.

Deficiency of uPA or plasminogen has been shown to lead to reduced healing and to promote progression of fibrosis[30]. Fibrosis is often characterized as an overactive wound-healing response [31]. Our observation that uPA and PLS are relatively insensitive to stimuli when they are not near the ultrasensitive threshold might aid in understanding why the wound healing response fails to switch off in some cases. The bistability of the uPA/PLS subsystem may have far-reaching consequences that should be investigated in future work.

MATERIALS AND METHODS

Model simulations were done using the ODE15s stiff solver of MATLAB (www.mathworks.com). Pplane7 was used for Phase plane analysis and XPPAUT (www.math.pitt.edu/~bard/xpp/xpp.html) used for bifurcation analysis.

Human scUPA was purchased from American Diagnostica and Glu-PLG was from Merck. All solutions were prepared in Tris buffer (0.05 M Tris-HCL, 0.10 M NaC1, 0.01% Tween 80, pH 7.4). Chromogenic substrate of PLS, s-2251, with chemical formula H-D-Val-Leu-Lys-pNA.2HCl was purchased from Chromogenix. In the presence of PLS the following reaction occurs, releasing pNA.

H-D-Val-Leu-Lys-pNA + PLS → H-D-Val-Leu-Lys-OH + pNA

The color intensity of pNA can be measured at O.D. of 405nm. A TecanM200 microplate reader was used for measuring changes in absorbance.

ABBREVIATIONS

ECM - Extracellular Matrix; PLG - Plasminogen; PLS - Plasmin; TGF-β1 - Transforming growth factor Beta 1; LTGFβ1 - Latent Transforming growth factor Beta 1;UPA - Urokinase Plasminogen activators; scUPA - single chain Urokinase Plasminogen activator; tcUPA - two chain Urokinase Plasminogen activator; PAI1 - Plasminogen Activator Inhibitor -1;ODE - Ordinary differential equations.

ACKNOWLEDGEMENTS

This work is supported by the Singapore -MIT Alliance Computational and Systems Biology Flagship Project funding to Sourav Bhowmick, Hanry Yu and Forbes Dewey; fellowship and grant R-252-000-342-112 from the Lee Kuan Yew Foundation to Tucker-Kellogg, and by Singapore-MIT Alliance grant C-382-641-004-091 to Tucker-Kellogg.

APPENDIX A

Table 3 –Parameter values and references

Parameters	Parameter values	References	Normalized Values Used (Values/$Keff_2$ *β)
$Keff_1$	0.061 $\mu M^{-1}min^{-1}$	[16]	0.0017 $\mu M^{-1}min^{-1}$
$Keff_2$	35 $\mu M^{-1}min^{-1}$	[8]	1 $\mu M^{-1}min^{-1}$
$Keff_3$	0.978 $\mu M^{-1}min^{-1}$	[16]	0.03 $\mu M^{-1}min^{-1}$
n	1<n<5		3
$\mu_1 = \mu_3$		[32], [6]	0.0001 min^{-1}
$\mu_2=\mu_4$		[33, 34]	0.001 min^{-}
α_1		[35]	0.00009 μM min^{-1}
α_2		[35]	0.001 μM min^{-1}

Table 3 lists the parameter values with references and their normalized values. Although these parameters are consistent with data and highly plausible, they are not necessarily unique. Due to the irreversibility of most of the reactions, production and degradation terms are non-trivial. Degradation (μ_i) and production (α_i) terms have been adjusted based on [32, 35], and they may contain some bias towards values exhibiting bistable behavior. The assumption of equal degradation rates for inactive and active proteases was made for nullclines, similarly to [15]. To avoid numerical errors from the XPPAUT software, the parameters were normalized as follows: all parameter values were divided by $Keff_2$ for rescaling, and multiplied by β for restoring units. β is $1\mu M^{-1}min^{-1}$.

APPENDIX B: REDUCTION METHOD FOR BIFURCATION ANALYSIS

We let TU, the total Urokinase, be defined as scUPA + tcUPA, and TP be defined as PLG + PLS. For analyzing the equilibrium of the system, we assume the following time derivatives to be zero, which would occur if the system is at steady state.

$$TU' = scupa' + tcupa' = 0 \qquad \text{(B.1)}$$

$$TP' = plg' + pls' = 0 \qquad \text{(B.2)}$$

For ease of notation, we assume equal degradation rates of the proteases, $\mu_1=\mu_4=D_1$ and $\mu_2=\mu_3=D_2$. By solving the ODEs at steady state and substituting in (B.1) and (B.2) we get

$$TU' = kprod_1 - D_1(tcupa + scupa) \quad \text{(B.3)}$$

$$TP' = kprod_2 - D_2(pls + plg) \quad \text{(B.4)}$$

Solving (B.3) and (B.5) further at steady state gives

$$TU = \frac{kprod_1}{D_1} = scupa + tcupa \quad \text{(B.5)}$$

$$TP = \frac{kprod_2}{D_2} = plg + pls \quad \text{(B.6)}$$

Substituting PLG=TP-PLS into the ode for PLS and solving for PLS at steady state, we get the PLS nullcine (PLS_n) as

$$pls\_n = \frac{a * TP}{D_2 + a} \quad \text{(B.7)}$$

Where $a = (keff_1 * scupa + keff_3 * tcupa)$ Similarly, the tcUPA nullcline (tcUPA_n) is

$$tcupa\_n = \frac{(pls)^n * keff_2 * TU}{keff_2 * (pls)^n + D_1} \quad \text{(B.8)}$$

REFFERENCES

[1] D. Angeli, Syst Biol (Stevenage) 153 (2006) 61-69.
[2] J.M.B. Bree B. Aldridge, Douglas A. Lauffenburger and Peter K. Sorger, Nat Cell Biol 8 (2006) 1195.
[3] U.S. Bhalla, R. Iyengar, Science 283 (1999) 381-387.
[4] E.M. C. Fall , J Wagner, John Tyson Computational Cell Biology, 2004.
[5] S.H. Strogatz, Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering, 2001.
[6] E.Z. Bagci, Y. Vodovotz, T.R. Billiar, *et al.*, Biophys J 90 (2006) 1546-1559.
[7] S. Legewie, N. Bluthgen, H. Herzel, PLoS Comput Biol 2 (2006) e120.
[8] T. Eissing, S. Waldherr, F. Allgower, *et al.*, Biophys J 92 (2007) 3332-3334.
[9] J. Cui, C. Chen, H. Lu, *et al.*, PLoS ONE 3 (2008) e1469.
[10] A. Ciliberto, B. Novak, J.J. Tyson, Cell Cycle 4 (2005) 488-493.
[11] B. Novak, Z. Pataki, A. Ciliberto, *et al.*, Chaos 11 (2001) 277-286.
[12] B. Novak, J.J. Tyson, B. Gyorffy, *et al.*, Nat Cell Biol 9 (2007) 724-728.
[13] J.J. Tyson, K.C. Chen, B. Novak, Curr Opin Cell Biol 15 (2003) 221-231.
[14] A. Goldbeter, D.E. Koshland, Proc Natl Acad Sci U S A 78 (1981) 6840-6844.
[15] T. Eissing, S. Waldherr, F. Allgower, *et al.*, Biosystems 90 (2007) 591-601.
[16] M.F.S. Vincent Ellis, and Vijay V. Kakkar, J Biol Chem 262 (1987) 14998-15003.
[17] J.H. Chesebro, G. Knatterud, R. Roberts, *et al.*, Circulation 76 (1987) 142-154.
[18] J.M. Stassen, J. Arnout, H. Deckmyn, Curr Med Chem 11 (2004) 2245-2260.
[19] J. Folkman, Journal of Thrombosis and Haemostasis (2003).
[20] H.R. Lijnen, Thromb Haemost 86 (2001) 324-333.
[21] S.Y. Kim, J.E. Ferrell, Cell 128 (2007) 1133-1145.
[22] H.R. Lijnen, B. Van Hoef, F. De Cock, *et al.*, Blood 73 (1989) 1864-1872.
[23] D. Collen, C. Zamarron, H.R. Lijnen, *et al.*, J Biol Chem 261 (1986) 1259-1266.
[24] H.R. Lijnen, B. Van Hoef, L. Nelles, *et al.*, J Biol Chem 265 (1990) 5232-5236.
[25] C. Chen, J. Cui, H. Lu, *et al.*, Biophys J 92 (2007) 4304-4315.
[26] A. Goldbeter, D.E. Koshland, Journal of Biological Chemistry (1984).

[27] J.R. Pomerening, E.D. Sontag, J.E. Ferrell, Nature Cell Biology (2003).
[28] D. Ribatti, B. Nico, E. Crivellato, *et al.*, Leukemia (2007).
[29] V.W. van Hinsbergh, M.A. Engelse, P.H. Quax, Arterioscler Thromb Vasc Biol 26 (2006) 716-728.
[30] W.Y. Li, S.S. Chong, E.Y. Huang, *et al.*, Wound Repair Regen 11 (2003) 239-247.
[31] G.K. Michalopoulos, M. DeFrances, Adv Biochem Eng Biotechnol 93 (2005) 101-134.
[32] H.C. T. Eissing, Journal of Biological chemistry 279 (2004) 36892–36897.
[33] J. Travis, G.S. Salvesen, Annu Rev Biochem 52 (1983) 655-709.
[34] M. Kohler, S. Sen, C. Miyashita, *et al.*, Thromb Res 62 (1991) 75-81.
[35] E.Z. Bagci, Y. Vodovotz, T.R. Billiar, *et al.*, Biophysical Journal (2006).

MULTI-RESOLUTION MODELING OF BIOLOGICAL MACROMOLECULES

SAMUEL FLORES, JULIE BERNAUER, XUHUI HUANG, RUHONG ZHOU, AND SEOKMIN SHIN

1. Introduction

The field of molecular modeling has long recognized modeling the folding, assembly, and long-time dynamics of large macromolecules as its biggest challenge, since it is precisely at large size and long time scales that computational methods are most taxed. In response, many methods have been developed to reduce the computational cost by coarsening the granularity of the problem. Inevitably accuracy remains limited for these methods and so in recent years a consensus has emerged that we should work at more than one level of resolution, often simultaneously, in an approach known as multi-resolution modeling. We propose to organize a session on multi-resolution approaches to predict and analyze macromolecular structure, assembly and dynamics. This session will focus on state-of-the-art methodological developments and applications at different levels of molecular organization. Research directions are based on developments arising in the community field of the organizers, like the ones developed in the NIH Center for Biomedical Computation at Stanford University. In this proposal, the emphasis is made on integrative techniques for analysis of molecular structure and organization. Bridging the gap between computer science and structural biology, we represent an emerging community, presenting new efficient representations for the prediction and analysis of molecular structures and dynamics.

2. Session Summary

.
This session includes an invited talk, seven reviewed oral presentations, three additional accepted papers, a discussion session, and a tutorial. We are pleased to have the distinguished Professor Michael Levitt as our invited keynote speaker. Ron Levy will preside at our discussion session.

3.1 *Oral presentations*

Multi-resolution Modeling of Biological Macromolecules toward understanding allosteric signaling mechanisms in the ATPase domain of molecular chaperones
Authors: Ying Liu and Ivet Bahar

This paper represents a biologically important problem: the allosteric signaling pathways of molecular chaperones, HSP70. In order to investigate this problem, the authors adopt a multi-resolution approach by combining methods at different resolutions: sequence alignment, residue contact map, Gaussian network model, and mutual
information. They show that a subset of central residues located at the interface between the two lobes of the Nucleotide Binding Domains near the nucleotide binding site form a putative communication pathway invariant to structural changes.

Multi-resolution and multi-physics approach for interactively locating functionally linked ion binding sites by steering smal molecules into electrostatic potential maps using a haptic device
Authors: Olivier Delalande, Nicolas Ferey, Benoist Laurent, Marc Geroult, Brigitte Hartmann, and Marc Baaden

In this work a haptic device is cleverly used to find ion binding sites on a protein. The user experiences tactile feedback which provides a quick and intuitive assessment of binding affinity.

3D-BLAST: 3D protein structure alignment , comparison, and classification using spherical polar

Fourier correlations
Authors: Lazaros Mavridis and David Ritchie.

The authors represent protein atomic density using spherical harmonics and Laguerre-Gaussian radial functions and find that the lowest-order terms can be used to quickly scan a database of structures and evaluate shape similarity. The results suggest a structural search tool fast enough to be implemented on a web server is within reach.

Structural prediction of protein-RNA interaction by computational docking with propensity-based statistical potentials
Authors: Laura Pérez-Cano, Albert Solernou, Carles Pons, Juan Fernández-Recio

Relatively little work has been done in protein-RNA docking, possibly due to the paucity of solved complexes. In this work a coarse-grained residue-nucleotide potential is devised which is then used to discriminate the native complex from decoys generated by rigid-body rotations and translations of the molecules.

Multi-resolution Modeling of Biological Macromolecules Multiscale dynamics of macromolecules using Normal Mode Langevin
Authors: Jesus A. Izaguirre, Christopher R. Sweet, and Vijay S. Pande

In this paper, the authors introduce a new approach that has addressed an important issue in multi-scale modeling of biological macromolecules: how to project out fast motions of biological macromolecules in an automatic way and explicitly propagate only slow degrees of freedom. In this way, one can achieve a speedup in simulations without sacrificing accuracy. In order to achieve this goal, the authors use the coarse-grained normal mode analysis to decompose the dynamics of the biological macromolecules into slow and fast modes. Only slowest degrees of freedom are then explicitly propagated by a Langevin integrator. They demonstrate the power of
the new method by folding of the Fip35 mutant of WW domain. This method is promising in multi-scale simulations of biological macromolecules.

Constructing multi-resolution markov state models (MSMS) to elucidate RNA hairpin folding mechanisms
Authors: Xuhui Huang, Yuan Yao, Gregory Bowman, Jian Sun, Leonidas Guibas, Gunnar Carlsson, and Vijay Pande

Most multi-scale modeling approaches are based on multi-scale representation of the molecules. These approaches are normally limited by their accuracy by leaving out important degrees of freedoms. In this paper, the authors introduce a completely different multi-scale approach by generating multi-resolution Markov State Models (MSMs) to analyze dynamic atomistic simulation data. MSMs are a powerful tool to predict long timescale dynamics from many short simulations; however, previous constructions of MSMs are all focused on a single resolution. The key insight of their new algorithm Super-level-set Hierarchical Clustering (SHC) is to generate a set of super levels covering different density regions of phase space, and then cluster each super level separately. The authors demonstrate the power of the new algorithm using the folding of a small RNA hairpin. This new algorithm holds its promise to bridge the timescale gap between simulations and experiments, and also opens up a new generation of algorithms to treat the dynamics at multi-resolutions.

3.2 *Papers without oral presentation*

Predicting RNA structure by multiple template homology modeling
Authors: Samuel Flores, Yaqi Wan, Rick Russell, and Russ Altman.

This is the first published application of RNABuilder, a multi-purpose macromolecular modeling code

which uses the Simbody internal coordinate mechanics library. The authors show that a "Frankenstein" molecule comprised of ribozyme fragments from two different species can be used as a template for modeling the structure of a ribozyme from a third. In regions for which no template is available, base pairing forces are used to enforce known contacts. The results show how threading can be done for highly interconnected RNAs with relatively low sequence similarity to molecules of known structure.

Insights into the intra-ring subunit order of TRIC/CCT: a structural and evolutionary analysis
Authors: Nir Kalisman and Michael Levitt

The TRiC complex in eukaryote is composed of 8 different protein products arranged in a ringed structure. The order of the subunits in the ring is still unknown despite its important functional role. In this work we mapped evolutionary information of TRiC onto a structural hypothesis, which led us to formulate several restrictions on the possible ring arrangements. We conclude that 72 ring arrangements (out of the possible 7!=5040) are consistent with those restrictions.

3.3 *Tutorial*

The tutorial aims to address emerging multi-resolution approaches to predict and analyze structure, assembly and dynamics of biological macromolecules. In particular, the following three topics will be discussed: (1) Multi-scale modeling of RNA structures. (2) Macromolecular assembly prediction and analysis, and (3) Conformational sampling.

Multi-scale modeling for RNA structures. Turning the secondary structures of a RNA molecule into 3-dimensional RNA structure can be done with the assistance of various computer programs. For example, Nucleic Acid Builder (NAB) is a modeling language with utilities to create, analyze and manipulate molecular structures. Visual Molecular Dynamics (VMD) is a popular molecular viewer which permits the user to drag atoms or molecules using a mouse. It is also possible to connect to a molecular dynamics package and apply forces with a mouse, a process called Interactive Molecular Dynamics (IMD). Multi-resolution techniques could potentially improve such tools. For example, the user could be empowered to selectively rigidify parts of the molecule, or to apply forces to specific internal coordinates such as torsion angles, or to enforce geometric constraints between molecular subunits such as nucleotide bases. One such tool, RNABuilder, is introduced in this session.

Macromolecular assembly prediction and analysis. The function of biological macromolecules often relies on their interactions with one or many partners. These interactions tend to deform the macromolecules to better adapt their binding partners. This process involves not only shape complementarities but also chemical interactions. For example, protein-protein docking prediction techniques usually include two steps: finding putative complex conformations and scoring them to keep the most biologically relevant. Finding suitable conformers involves large conformational sampling, and it is difficult to be achieved at atomic level for large assemblies. Thus, multi-resolution approaches are necessary to be introduced to deal with large complexes. Such multi-resolution approaches have already shown to be the promising by successfully predicting individual side-chain contacts at the interface in the last CAPRI, the worldwide competition on docking of protein complexes. Furthermore, in the past few years, new algorithms dealing with multiple scales of representations and multiple levels of experimental data have emerged in the docking community for detecting interfaces, sampling and scoring the exploration results. In the tutorial, we will discuss the challenges on developing new multi-resolution techniques for conformation selection of protein-protein docking. For example, multi-scale scoring functions based on geometric criteria will be discussed.

Conformational sampling. Conformational sampling is one of the major changelings for the multi-scale modeling of the biological macromolecules due to the rugged nature of the free energy landscapes. Without adequate sampling, it is impossible to validate the parameters or force fields or to address phenomena that occur on biologically relevant timescales. Many methods have been developed in an attempt to address the sampling problem such as Generalized Ensemble (GE) algorithms like Replica Exchange Method (REM) and Simulated Tempering (ST), Metadynamics, Transition path sampling, and

Adaptive seeding method. Some sampling algorithms such as Resolution Replica Exchange can even directly integrate simulations running at different resolution and greatly enhanced their ability to sample the conformational space.

3. Acknowledgements

We gratefully acknowledge session funding from INRIA, the Stanford Simbios Center, and IBM Research. Samuel Flores and Xuhui Huang acknowledge support from the NIH Center for Physics-Based Simulation of Biological Structures (Simbios) with NIH Roadmap U54 GM072970. Xuhui Huang also acknowledges support from the Hong Kong University of Science and Technology. The session chairs thank the many anonymous referees who read submissions and made recommendations.

MULTI-RESOLUTION APPROACH FOR INTERACTIVELY LOCATING FUNCTIONALLY LINKED ION BINDING SITES BY STEERING SMALL MOLECULES INTO ELECTROSTATIC POTENTIAL MAPS USING A HAPTIC DEVICE

OLIVIER DELALANDE, NICOLAS FEREY, BENOIST LAURENT, MARC GUEROULT[§]
Laboratoire de Biochimie Théorique, CNRS UPR9080/IBPC, 13 rue Pierre et Marie Curie, F-75005, Paris, France

BRIGITTE HARTMANN
[§] DSIMB team, INTS, 6 rue Alexandre Cabanel, F-75739 Paris Cedex 15, France

MARC BAADEN
Laboratoire de Biochimie Théorique, CNRS UPR9080/IBPC, 13 rue Pierre et Marie Curie, F-75005, Paris, France

Metal ions drive important parts of biology, yet it remains experimentally challenging to locate their binding sites. Here we present an innovative computational approach. We use interactive steering of charged ions or small molecules in an electrostatic potential map in order to identify potential binding sites. The user interacts with a haptic device and experiences tactile feedback related to the strength of binding at a given site. The potential field is the first level of resolution used in this model. Any type of potential field can be used, implicitly taking into account various conditions such as ionic strength, dielectric constants or the presence of a membrane. At a second level, we represent the accessibility of all binding sites by modelling the shape of the target macromolecule *via* non-bonded van der Waals interactions between its static atomic or coarse-grained structure and the probe molecule(s). The third independent level concerns the representation of the molecular probe itself. Ion selectivity can be assessed by using multiple interacting ions as probes. This method was successfully applied to the DNase I enzyme, where we recently identified two new cation binding sites by computationally expensive extended molecular dynamics simulations.

1. Introduction

Metal ions drive important parts of biology, yet it remains experimentally challenging to locate their binding sites. A particular example may concern the mechanisms underlying the formation of non-specific protein-DNA complexes. In this context, we have recently studied the DNase I/DNA system as a representative and rather simple model of a non-specific complex. DNase I is a glycoprotein hydrolyzing DNA phosphodiester linkages in the presence of divalent cations, Ca^{2+} and Mg^{2+}. We demonstrated that Ca^{2+} and Mg^{2+} are crucial for optimizing the electrostatic fit between DNA and enzyme. In particular, extended molecular dynamics simulations at atomic detail allowed us to identify two new cation binding sites that are functionally important [1]. Such high-resolution atomistic methods are computationally expensive and require long simulation times. Could these ion binding sites also have been detected with simpler methods? Maybe this could be achieved at lower resolution and requiring less computational power? In the present manuscript we set out to describe a new method aimed at these goals, using the molecular dynamics results as a reference for assessing our findings. The purpose of this method is to provide a fast user-guided search in order to locate potential binding sites, prior to detailed investigations of these sites using more accurate but computationally expensive approaches.

Recently, we have described the exciting possibilities of interactive molecular simulations for studying biological macromolecules [2]. Here, we demonstrate that this approach can be used for locating ion binding sites. The concept of our method is to interactively explore electrostatic potential fields while being guided towards the binding sites by force feedback. This interactive haptic approach may be traced back to the work of Nagata *et al.* [3] who explored the concept in the context of protein-ligand docking. Their work was somewhat ahead of its time, as may be illustrated by the following observations: "Certain limitations remain; for example, only twenty protein atoms can be used to generate the electrostatic field. Furthermore, the system can only use globular probes, preventing drug molecules or small chemical groups from being simulated. These limitations are the result of our insufficient computer resources". Today, this situation has changed and we did not observe any significant computational limitations. Our approach has been tested on systems as big as a pentameric ligand-gated ion channel comprising 1 500 amino acid residues [4]. Of course, the raw increase in computer speed is not the only reason for such an improvement. Adapted software solutions and improved algorithms further render current approaches more efficient.

Using such an interactive approach for locating ion binding sites is a new idea. It will be illustrated with the DNase I example system. Previously, electrostatic steering has been described for a variety of cases such as

enzyme-ligand binding [5], tRNA binding [6], antibody-antigen association [7] and Cdc42 recognition [8]. Computational methods have been used in several of these studies, but interactive virtual reality (VR) approaches only marginally. Why are such VR approaches useful? VR efficiently introduces a human element in the process and benefits from the user's experience and insight. Enabling the user to sense the electrostatic potential field via tactile feedback is a major advantage. The electrostatic potentials of biomolecules are often complex volume and multi-level data, rendering the visual perception of individual binding sites difficult if not impossible. Adding haptic feedback to their interactive exploration significantly simplifies this task. Here, we use such an approach on a biomolecular system in order to explore potential variations at the DNase surface, to locate favourable binding sites, and to discover far-reaching electrostatic pathways guiding ions to these sites.

The representation used for the simulation system is innovative. It could be summarized as a multi-level simulation combining multiple physics models and is intrinsically multi-resolution. The potential field is the first level of resolution used in this model. At a second level, we represent the accessibility of all binding sites by modelling the shape of the target macromolecule at atomic or coarse-grained resolution. The third independent level concerns the representation of the molecular probe itself. By using multiple scales and resolutions it is possible to focus on the essential degrees of freedom of a complex biological system. This makes the user's interaction with the system more efficient.

The interactive VR-based approach renders the scientific task nice and enjoyable, hence encouraging the user to pursue his investigation. Another example of this kind is the ''Fold It!'' project (http://fold.it) that recently brought interactive simulations to the fore. ''Fold It!'' is a 3D-puzzle desktop game, in which the user's task is to fold proteins interactively and without any knowledge prerequisites. The puzzle is essentially based on an interactive-intuitive learning process. The similarities such applications bear with video games should not delude scientists to underestimate the scientific value of such approaches.

2. Materials and Methods

2.1. *Structural data and model preparation for the DNase I system*

Several experimental structures of the DNase I enzyme from different biological contexts were considered as reference in this study: the enzyme crystallized in its *apo* form (PDB code 3DNI [9]), in a DNase I:Actin complex (PDB code 1ATN [10]) or bound to an oligonucleotide duplex (PDB code 1DNK [11]). Calcium ions are found in some of these structures: *apo* DNase I (3DNI) bears two Ca^{2+} binding sites (described below as sites 1 and 2) and the DNase I:Actin complex reveals three Ca^{2+} binding sites (sites 1, 2 and 3).

In the interactive explorations, a model derived from previous molecular dynamics (MD) simulations is used to represent the enzyme. This model corresponds to the most representative conformer of the whole trajectory as determined by an exhaustive cluster analysis [1]. The detection of ionic binding pockets and their localization in the crystallographic structures have been based on this model. In the MD simulations, the *AMBER* parm99 force field has been used [12]. Consistently, we generated the parameters for the electrostatic potential calculation (*pqr* files) with the *pdb2pqr* program [13] using the *--ff=amber* option, including for ionic probes. The potential was determined with the *APBS* software [14] and standard input was obtained with the *--apbs-input* option. Ionic strength was introduced using a 0.15 M concentration of both +1 and -1 charged ions with a 2.0 Å radius value (*APBS* parameters).

The shape of the DNase I molecule was represented *via* van der Waals parameters from the *AMBER* parm99 force field (atomic resolution) or from the coarse-grained model by Zacharias [15] (low resolution).

Every interactive experiment consisted in at least four independent trials to detect binding pockets for each probe. We concluded that a binding site was found when the probe got stuck in a location where it was impossible to escape without increasing the user force range. Attractive pockets at the enzyme surface where the ion could easily be retrieved were not considered as binding sites.

2.2. *MyPal: an interactive simulation approach combining an electrostatic potential field and pairwise non-bonded interactions*

MyPal stands for **M**olecular scrutin**Y** of **P**otenti**AL**s. The application has a corresponding French name with the same meaning: *MonPote*, "**MO**lécules **N**aviguant sur un **POTE**ntiel". In this section we describe the underlying methodology developed in order to study the behaviour of ions or small molecular probes interacting with a static macromolecular target molecule. In this approach, based on a classical Newtonian simulation, ionic probes are immersed into and guided by an electrostatic potential field induced by the target molecule. In addition, the 3D shape of the target molecule is taken into account by representing its static 3D structure interacting with the probe molecules. The simulation deals with both electrostatic properties and steric constraints induced by the 3D shape of the target molecule. Moreover, in order to study the behaviour of several interacting ions or molecules,

non-bonded Coulomb and van der Waals interactions between all the probes are calculated. These inter-probe interactions are merged with the electrostatic potential from the map and the van der Waals interactions with the target molecule. The user can remain passive and observe the movements of the ionic probes from their initial positions during a simulation in progress. Alternatively, the user can act on the probes by selecting and applying external forces taken into account by the simulation, in order to explore the potential and identify binding sites.

A *MyPal* simulation requires two datasets as schematically shown in Figure 1. The first dataset corresponds to the structure and potential of the target macromolecule, including all static particles in the simulation (left). The second dataset describes a separate dynamic molecular structure including the ionic probes (right).

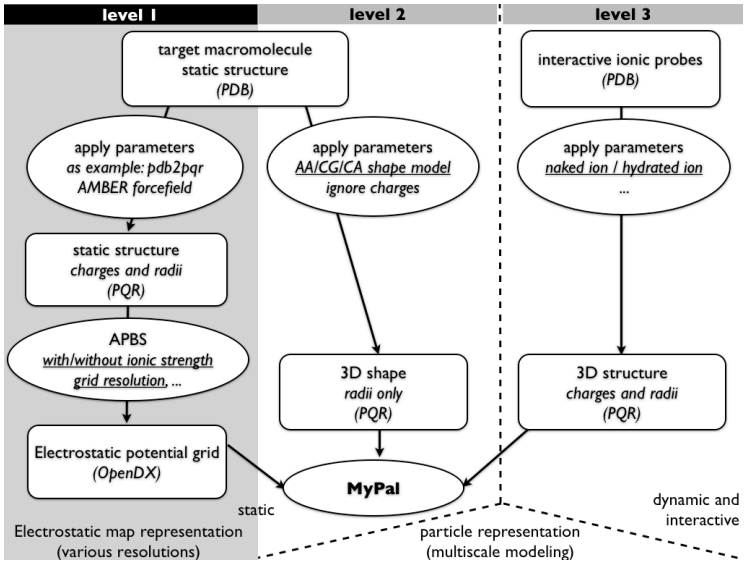


Figure 1: Schematic flow diagram for setting up a *MyPal* simulation. **Level 1** deals with the calculation of the electrostatic potential grid. Here we use the *pdb2pqr* tool in order to set up the radii and charges of the target macromolecule using the atomistic *AMBER* parm99 force field. Then we compute its electrostatic potential map using the *APBS* software [14]. The potential is saved as an *OpenDX* file and used as input for *MyPal*. At **level 2**, the 3D shape of the target molecule is modelled as a set of static spherical particles at the desired resolution (all-atom: AA, multi-bead coarse grained: CG or single-bead carbon alpha: CA). The ionic probes are modelled at **level 3**, using a set of radii and charges. The latter parameters may represent different resolutions such as naked *vs.* hydrated ions. Levels 1 and 2 are static, whereas the particles in level 3 are dynamic and can be guided by the user.

During the simulation, forces are computed and applied on the dynamic particle set ($P_{dynamic}$). We explicitly consider potential, van der Waals, Coulomb and external forces.

The potential forces F_{elect} act on the ionic probes and originate from the electrostatic potential map. They are defined by computing the gradient of the electrostatic potential. In Equation (1), we consider particle p belonging to the spatial cell $C_{i,j,k}$ of the electrostatic potential grid, and $E_{i,j,k}$ the value of the potential in this cell. We define the gradient as the mean of the difference between the $E_{i,j,k}$ potential and the potentials of the six adjacent cells, two for each axis. This method of computing the gradient reduces the bias related to the discretization of the grid.

$$\vec{F}_{elec}(p \in C_{i,j,k}) = \begin{bmatrix} \dfrac{\left(E_{i,j,k} - E_{i-1,j,k}\right) + \left(E_{i+1,j,k} - E_{i,j,k}\right)}{2.delta_x} q_p \\ \dfrac{\left(E_{i,j,k} - E_{i,j-1,k}\right) + \left(E_{i,j+1,k} - E_{i,j,k}\right)}{2.delta_y} q_p \\ \dfrac{\left(E_{i,j,k} - E_{i,j,k-1}\right) + \left(E_{i,j,k+1} - E_{i,j,k}\right)}{2.delta_z} q_p \end{bmatrix} \qquad (1)$$

The van der Waals forces F_{vdw} applied to the ionic probes ($P_{dynamic}$) are computed taking into account all interactions between the ionic probes and the static particles defining the 3D shape of the target molecule, and between ionic probes themselves (P_{all}). As shown in Equation (2), these forces are computed using the following approximation for the van der Waals interactions.

$$\vec{F}_{vdw}(p \in P_{dynamic}) = \sum_{p' \in P_{all}} \vec{u}_{pp'} 4\varepsilon_{pp'} \left[\left(\frac{\sigma_{pp'}}{9d_{pp'}} \right)^9 - \left(\frac{\sigma_{pp'}}{7d_{pp'}} \right)^7 \right] \qquad (2)$$

The Coulomb forces $F_{coulomb}$ applied to the ionic probes ($P_{dynamic}$) take into account all electrostatic interactions between the ionic probes according to Equation (3).

$$\vec{F}_{coulomb}(p \in P_{dynamic}) = \sum_{p' \in P_{dynamic}} -\frac{q_p q_{p'} \vec{u}_{pp'}}{4\Pi\varepsilon_0 d_{pp'}^2} \qquad (3)$$

Finally, these forces are summed with an external force provided by the user through the graphical interface during the simulation.

$$\vec{F}(p \in P_{dynamic}) = \vec{F}_{elec}(p) + \vec{F}_{vdw}(p) + \vec{F}_{coulomb}(p) + \vec{F}_{user}(p) \qquad (4)$$

This multi-resolution approach consists of three independent levels of modelling. The potential field is the first level of resolution used in this model (level 1 in Figure 1). In principle, any type of potential field could be used, not only electrostatic ones. Most of the scrutinized potentials in the present work are Poisson-Boltzmann type electrostatic potentials computed using the *APBS* software [14]. Such potentials are subject to several parameters (resolution, ionic strength, solvent and solute dielectric constants) and may thus implicitly take into account an averaging over selected solvent or solute degrees of freedom. More complex scenarios, for instance the presence of a biological membrane environment for a membrane protein target, could also be taken into account with tools such as AquaSol [16]. Deriving the potential field from an average over a molecular dynamics simulation could help to mimic the flexibility of the target molecule. These are only a few examples of how the resolution and underlying physics of the potential field can be varied. At a second level (level 2 in Figure 1), we model the accessibility of a binding site by representing the shape of the target macromolecule *via* non bonded van der Waals interactions between its static structure and the probe molecule(s). For this representation, a whole range of parameterizations commonly used in molecular mechanics is available. These comprise the detailed atomistic scale (AA), medium-resolution multi-bead coarse graining (CG) and single-bead carbon alpha models (CA) [17]. Level 3 in Figure 1 is the third, independent level of modelling and concerns the molecular probe itself. At this level, van der Waals and coulombic interactions are treated. Radii and charges can be chosen in order to fine-tune the desired properties of the ionic probes by considering effects such as solvation (increased radii) or charge screening (decreased charges).

It should be stressed that the *MyPal* approach is computationally cheap. The electrostatic map is computed offline prior to the interactive simulation and there are only a small number of dynamic ionic probes on which pairwise interactions are computed. Given the reduced number of degrees of freedom, namely the positions of the ionic probes, thorough sampling of phase space is easily achieved.

2.3. *Visualizing a simulation in progress*

During the design of the *MyPal* application, initial tests have been carried out with the VMD software [18]. This approach couples visualization and simulation with the *MDDriver* library [19], using the IMD protocol [20]. All screenshots in the present manuscript were obtained using VMD. VMD is however not optimally adapted to some of the specific tasks occurring during an interactive simulation. As an example, there is no visual feedback during the particle selection task, making it difficult to select the particle of interest with confidence. Multiple selections are not available using direct interactions with the mouse or a haptic device, but only using the Graphical User Interface (GUI). In order to surmount these shortcomings, we are now developing our own simple visualization tools specifically designed for interactive visualization. We use the visualization toolkit VTK [21] for visual rendering. VTK provides high-level features, such as isosurface rendering of electrostatic potential maps. In an enhanced version, our tool allows us to visualize up to several hundred thousand particles in interactive time using a GPU shader implementation of spherical representations. VTK was encapsulated into a Cocoa application, allowing us to quickly develop a GUI using the XCode and Interface Builder tools. As soon as this application will be finalized, it will be made freely available in order to simplify interactive potential exploration as described in this work [22].

2.4. *Interacting with the ionic probes during a simulation*

In order to interact with a *MyPal* simulation in progress, it is possible to use a mouse for adding force constraints on the probe particles. This approach provides two degrees of freedom, *e.g.* the x- and y-axes, for the interaction. However, using a haptic device is even better adapted to this task, in particular for selecting and moving particles in 3D space. Such a device with three instead of two degrees of freedom is more intuitive and efficient for interacting with a complex three-dimensional object. Furthermore, the immediate haptic feedback when a

particle is actually picked significantly improves the user experience and greatly helps to immerse the user in the molecular scene. When using visual feedback only, the user often asks for additional explanations before getting started. With force feedback, this barrier is lifted, as the interactive simulation becomes more intuitive and is comparable to dextrous manipulations such as those carried out in daily life. Hardware requirements are modest (details in section 2.6). In our experience, this approach is viable using a small and affordable haptic device, providing 3D positions and handling 3D directional force feedback. Such an entry-level solution designed for a desktop use is targeted at a large user community and is very easy to set up. The availability of convenient drivers that are compatible with software libraries such as *VRPN* [23] was a major criterion for our choice.

The interaction with the simulation was implemented as in VMD, allowing the user to impose forces on probe particles and experience a tactile feedback. The haptic device is used in order to control the direction of the forces applied to selected particles and to adjust the amplitude of these forces. This interaction method contains two stages. The first stage comprises the selection of a single probe particle or a set of particles that we will name $P_{selection}$, using a 3D tool attached to a haptic device and its buttons. In a second stage, the model described in Equation (5) is used in order to compute the forces $F_{simulation}$ applied to the selected particles and sent to the *MyPal* simulation as external force (see section 2.2). $F_{simulation}$ is proportional to the distance between the geometrical centre of the particle set and the tracker position P_T.

$$\vec{F}_{simulation}(p \in P_{selection}) = k_{simulation}(\vec{P}_T - \frac{1}{|P_{selection}|}\sum_{p' \in P_{selection}} \vec{P}_{p'}) \qquad (5)$$

The main idea of this approach is to link the selected atoms and the 3D haptic tool with a spring. Instead of providing direct haptic rendering of forces computed in the simulation, the force feedback $F_{feedback}$ only depends on the spring length according to Equation (6), which in turn is influenced by the way the simulation reacts to the applied force.

$$\vec{F}_{feedback} = -k_{feedback}(\vec{P}_T - \frac{1}{|P_{selection}|}\sum_{p \in P_{selection}} \vec{P}_p) \qquad (6)$$

We emphasize that the haptic loop computation frequency must be between at least 300 to 1000 Hz in order to provide a haptic rendering of good quality. A strong point of the approach described above is that a low physical simulation framerate does not cause instabilities and does not affect the quality of the haptic feedback. With this decoupled spring model, force feedback can be computed at a very high frequency required by the haptic device.

2.5. *Coupling simulation, visualization and interaction*

In this section, we briefly explain how simulation, interaction and visualisation modules exchange data. Figure 2 illustrates a schematic view of the data flow in the application presented in this work. We use the *MDDriver* library as central element in order to couple a visualization, interaction and *MyPal* simulation module. Previously, this library was successfully used to couple the *GROMACS* molecular dynamics engine [24] with *VMD* in order to carry out interactive studies of several biomolecular systems comprising enzymes and membrane proteins [2]. Here we use a similar approach in order to couple the *MyPal* program with our own visualization code based on *VTK* or with the visualization module *VMD*.

The *MDDriver* library provides a simplified and high-level API [19] for exchanging particle positions computed in a simulation module and custom user-generated forces from the visualization and interaction modules. The adapter layers shown in Figure 2 are specific for a given simulation or visualization software. They are mainly in charge of transforming the physical values into adequate units, but can also operate more complex transformations, such as changing data from a given discretization model to another one. The adapters are in charge of redistributing and collecting the data in a parallel application. For implementing *MDDriver* in a given software package, the major part of the work consists of locating the adequate routines in the simulation algorithm where the data exchange needs to take place. The exchange with the visualization tool includes sending atom positions, receiving forces and managing control events for the simulation.

Further coupling is needed by our interactive approach. It consists of coupling the peripherals management library, *e.g.* for working with a haptic device, and the visualization module. We use the *VRPN* library [23], which offers a device independent implementation, in order to read out the position and orientation of a tool used to select particles in the visualization module. *VRPN* is further used in order to generate force feedback along the lines described in section 2.4.

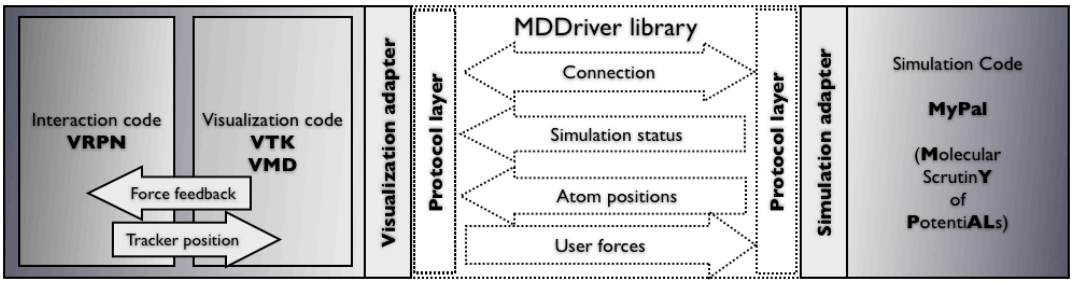


Figure 2: The *MDDriver* library couples a simulation code with a visualization code. *MDDriver* provides an API for exchanging particle positions computed in a simulation module and custom forces from the visualization and interaction modules [19]. Haptic device management is provided by the *VRPN* library [23] and force feedback is computed as described in section 2.4

2.6. *Hardware setup and typical configuration*

All interactive *MyPal* simulations were performed on a 3GHz dual quad-core MacPro Apple computer. Although the *MyPal* application itself is not parallelized, a multi-core workstation makes it possible to run all parts of the application on the same machine without performance issues. This includes visualization, haptic device server and calculation tasks. The molecular scene display was rendered using an NVIDIA Quadro FX 5600 graphics card. Stereo rendering was achieved via a Crystal Eyes stereovision device. Two devices were tested for user interaction and tactile feedback: a Phantom Omni haptic device and a higher precision Phantom Premium 1.5A, both by Sensable Technologies. Scene navigation was achieved with a Spaceball device providing six degrees-of-freedom.

The *MyPal* application can easily be run in a desktop context, with minimal spatial requirements, and does not require any sophisticated hardware setup. In our experiments, the typical configuration was to mount the haptic device, Spaceball, mouse and keyboard in front of a big screen. Figure 3 illustrates such a typical setup.

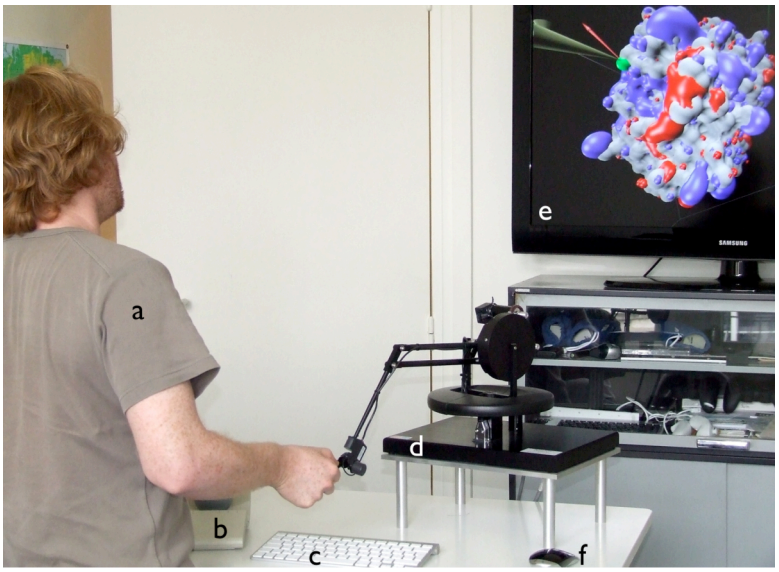


Figure 3: Typical hardware setup for an interactive exploration experiment based on the *MyPal* simulation tool. The user (a) can navigate the scene with a Spaceball device (b). Further control can be achieved *via* a keyboard (c) and a mouse (f). Interaction with the calculation and force feedback from *MyPal* occur *via* a haptic device (d). The scene is shown on a large screen (e) with an avatar representing the user-driven tool (cone in top-left corner of the screen), and a visual rendering indicating the user force (red arrow) acting on the ionic probe (green sphere).

3. Biological application – results and discussion

3.1. *DNase I cation binding sites revealed by atomistic molecular dynamics simulations*

We will use our previous results on cation binding sites in the DNase I enzyme as reference to compare our interactive simulations to. It has been shown elsewhere that molecular dynamics (MD) simulations are powerful – yet computationally expensive – tools for locating such sites [25]. Specific and non-specific protein-DNA interactions imply the formation of intermolecular interfaces requiring electrostatic and structural complementarity between the related partners. A precise functional role for metal ions in this process is certainly

an interesting hypothesis worthwhile exploring. The DNase I/DNA complex is a representative and rather simple model to study these non-specific interactions. We have recently carried out four 25 ns molecular dynamics simulations of the *apo* enzyme where the cation composition was varied between Na^{+}, Ca^{2+} and Mg^{2+} [1]. Detailed analysis of cation coordination in these simulations revealed four distinct sites. Two sites show a preference for calcium and two are selective for magnesium (Figure 4, left panel). The calcium binding sites were previously known [9], whereas the two magnesium sites were unexpected. We have shown that cations are essential for the biological function of DNase I, allowing DNA to bind to the enzyme. Indeed, the concurrent occupation of all four sites with the proper cation is required for an electrostatic fit between DNA, negatively charged, and the enzyme interface, also negatively charged in the absence of cations. This may imply that the information about the binding sites is encoded in the electrostatic potential map of the enzyme (Figure 4, right panel). In the following paragraphs, we reproduce key results of the MD study using the interactive *MyPal* application.

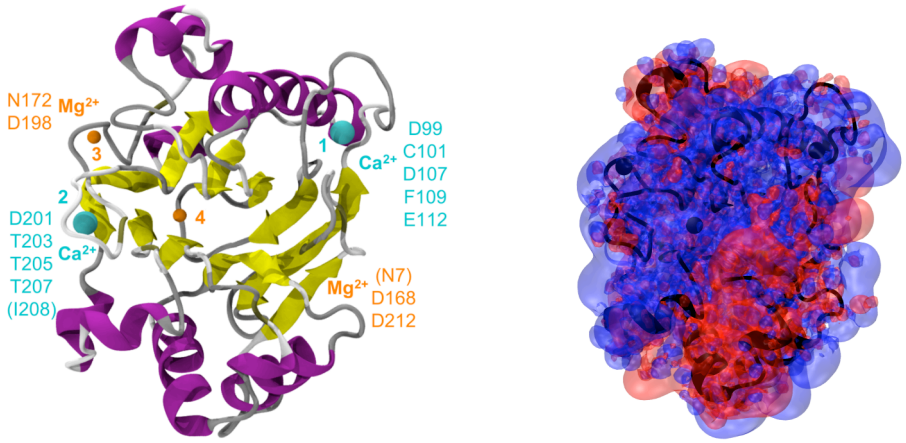


Figure 4: On the left, DNase I cation binding sites numbered from 1 to 4. Calcium binding sites are shown in cyan, magnesium binding sites in orange. The main amino acid residues constituting each site are indicated. On the right, electrostatic potential isosurfaces for potential values of -15.0(blue)/+15.0(red) kT/e are superposed onto the DNase I backbone structure and ions (in black).

3.2. *Locating the four previously identified ion binding sites with MyPal using interactive exploration at atomic resolution*

Figure 4 (right panel) illustrates the difficulty of visually identifying ion binding pockets due to the complexity of typical macromolecular electrostatic potentials. Even knowing their location (Figure 4, left panel), the binding sites cannot be distinguished. By using tactile feedback with our interactive *MyPal* application, it was however possible to locate all four binding sites with confidence and high precision (e.g. 0.9 Å positional deviation for the first site, mainly limited by the spacing of the potential grid). The interactive approach proved quick and intuitive.

Starting from a relaxed reference structure extracted from the MD trajectories, we interactively scanned the entire electrostatic potential surface of DNase I. An initial potential was calculated considering standard physiological ionic strength (0.15M NaCl). Na^{+}, Ca^{2+} and Mg^{2+} are potential ionic probes. For the identification of the binding sites we primarily used a hard sphere representation of the small divalent Mg^{2+} cation as it offers several advantages as a probe. Its double charge facilitates long-range electrostatic steering towards the binding pockets and its small size (0.79 Å) increases the accuracy for sensing the rough and detailed molecular surface at atomic resolution. The structural model of DNase I was extracted from MD simulations with binding sites occupied by ions that are mainly coordinated by amino acids and not water molecules. Consequently, it is important to consider a “naked” ion and not a hydrated one as probe. Otherwise the ion would not fit into the preformed coordination sphere at each site. More generally, considering hydrated ions using an implicit hydration shell remains feasible by increasing the probe radius and possibly varying the charge in order to take into account solvent shielding.

The parameters used for calculating the electrostatic potential affect the interactive exploration, in particular ionic strength. Taking into account ionic strength leads to locally more accurate potential maps. The downside of this for interactive experiments is that the detection of potential wells becomes harder as the range of electrostatic steering shortens. Without ionic strength, we achieved comparable precision, but did more easily detect several of the binding sites. The results for both series of experiments, with and without ionic strength, are presented in Figure 5. Despite the simplicity of the representations of protein surface and ions, all four ion binding sites identified by MD are retrieved by our approach. No other strong binding sites (false positive

locations) were detected, which can be a problem with other approaches [26]. We did observe several attractive interactions located at pockets on the enzyme surface. These locations correspond to shallow minima requiring little force to extract the ions and were not considered as binding sites.

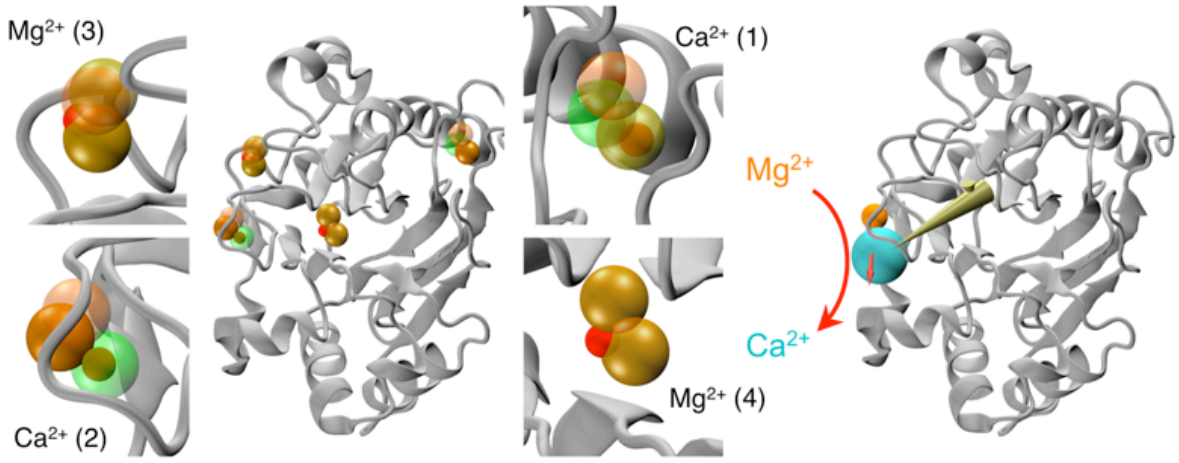


Figure 5: Visual summary of the interactive experiments. On the left, the results of a series of experiments for detecting *a priori* unknown ion binding sites are shown. The reference position of each binding pocket as determined by MD simulation is shown as a red sphere. Transparent spheres display *MyPal* predictions obtained for a potential map with (orange) or without (green) ionic strength. On the right, an ion substitution experiment ("molecular-billiard") at site 2 is depicted. Such an experiment probes the selectivity of a given ionic pocket for different ions.

3.3. *"Molecular-billiard" for probing site selectivity by ion substitution*

In the previous section, we successfully described how to locate the four DNase I cation binding sites. We did however not assess their selectivity. For this purpose, we have carried out a series of additional experiments, each starting with a different ionic probe at a given site. We then tried to interactively substitute the probe by another ion. We considered magnesium, calcium, sodium and chloride ions as probes. Figure 5 and Table 1 illustrate and summarize the results for these ion substitution "molecular-billiard" simulations.

As might be expected, chloride as an anion cannot be stabilized within any of the four cation binding pockets, nor can it displace a bound cation. Sites 1 and 2 are calcium selective, which is generally verified. It is however surprising that magnesium is able to substitute for calcium at site 1. This may be related to the simplicity of our model in which selectivity depends on the shape of the pocket itself and the pathway for accessing it. Generally speaking, buried and narrow sites are unreachable for large ions, whereas sites localized at the enzyme surface are readily subject to ion exchange. In the latter case, haptic feedback helps the user to distinguish between favourable and unfavourable substitutions. The favourable substitution of Na^+ with Ca^{2+} requires little user forces, whereas Na^+ can only displace Ca^{2+} using excessive force. Generally, Mg^{2+} can displace Ca^{2+} but the opposite remains impossible, except for the exposed calcium selective site 2. Sites 3 and 4 are selective towards the small divalent Mg^{2+} cation (r_{Mg2+}=0.79 Å). Site 4 is deeply buried and does not permit binding of the two bigger positive ions (r_{Ca2+}=1.71 Å ; r_{Na+}=1.87 Å). Site 3 can accommodate sodium and calcium in agreement with the 1ATN crystal structure [10]. Those ions are however easily displaced by magnesium. On the other hand, magnesium cannot be displaced by any of the other two cations, corroborating the magnesium selectivity of this site.

Table 1. Ion substitution simulation results. The table indicates whether exchange from X to Y is possible (→) or impossible (↛). For instance, Ca ↛ Cl means that Ca^{2+} cannot be displaced by Cl^-. A minus sign indicates that initial positioning of the chosen probe ion at the given binding pocket was not possible *via* our approach.

Probe (Site)	Ca^{2+} (1)	Ca^{2+} (2)	Mg^{2+} or Ca^{2+}(3)	Mg^{2+} (4)
Mg^{2+}/Ca^{2+}	Ca → Mg Mg ↛ Ca	Ca → Mg Mg → Ca	Mg ↛ Ca Ca → Mg	Mg ↛ Ca -
Na^+	Ca ↛ Na Na → Ca	Ca → Na Na → Ca	Mg ↛ Na Na → Mg	Mg ↛ Na -
Cl^-	Ca ↛ Cl -	Ca ↛ Cl -	Mg ↛ Cl -	Mg ↛ Cl -

The selectivity assessment remains a qualitative one, in particular due to the simplicity of the metal model. Recent work seems to indicate that induced polarization effects in the coordination sphere might be required to fully capture metal ion binding [27]. Such an improved potential function would presumably require further refinements, such as a completely polarizable forcefield, flexible sidechains and explicit water molecules [28, 29]. In our approach, we intentionally use approximations in order to speed up the search. More refined representations should be used *a posteriori* in order to refine the results of this initial haptic search.

The current implementation of *MyPal* was not designed in order to provide precise quantitative binding affinity estimates, but to be capable of distinguishing in real time between non-existing, weak and strong ion binding sites and assess the relative selectivity of significantly different ionic probes. The approximations made in the choice of the model representation limit the precision that could be obtained with the current implementation. Despite these limitations, it remains in principle possible to quantify the strength of binding. This aspect will be addressed in future work. One possible approach would be to calculate the work required by the user to extract an ion from its binding site.

These future extensions would also be beneficial for applications in docking and drug discovery. *MyPal* is already capable of handling rigid multi-atomic drug-like molecules. It is possible to displace a probe and assess whether it interacts favourably with cavities and pockets of a macromolecule. Currently the application does however lack tools to assess the ranking of different binding poses and the comparison between different probes. The implementation of such tools using the non-bonded interactions between target and probe molecules would be straightforward.

3.4. *Predicting the location of ionic binding pockets: multi-resolution exploration using a coarse grained protein shape*

Up to this point we have described experiments on a model structure with preformed ion binding sites. In this sub-section we validate our interactive approach using DNase I extracted from the DNase I/DNA 1DNK structure [11], crystallized without any divalent cation (ion-free DNase I). The expected locations of the binding sites were defined by superimposing the MD model onto the crystallographic structure.

We start with the same representation as previously, using a high-resolution atomistic model in order to describe the protein shape. All four binding sites can be located, albeit with reduced accuracy. The static conformation captured by the ion-free crystal structure biases the exploration by imposing an ill-suited local shape for the putative binding sites. For instance, in the case of the Mg^{2+} binding sites, the most solvent accessible and thus the least affected site deviates 2.5 Å from the reference location. Site 4, the most buried – *e.g.* potentially most biased – pocket can only be detected with significantly less precision and a deviation up to 3.9 Å from the expected site.

By lowering the resolution of the molecular shape representation, it should be possible to reduce these artefacts observed at atomic resolution. We used a coarse-grain representation with several beads per residue. In these simulations, the offset of the most buried magnesium site largely decreased from 3.9 to 2.2 Å. This decrease represents an important gain in precision. Localization of the more solvent-accessible pockets benefits to a lesser extent from coarse graining, with improvements ranging from 0.2 Å (site 3) to 0.4 Å (site 2).

Introducing flexibility in the model is an alternative to lowering the resolution of the representation. The protein or the complex could then be deformed in order to fit a binding interaction. The haptic device would help to render deformations fast and easy. Flexibility requires the introduction of bonded interactions in the model. Such a development is currently under way in our laboratory.

A positive conclusion can be drawn on the predictivity of our method. The multi-resolution approach combining an electrostatic potential grid computed from an all-atom structure and a coarse-grained molecular surface representation for steric repulsion is efficient for predicting ion binding sites in a quick and interactive manner. As an explanation, we may venture that the electrostatic contribution is a cooperative effect of several coordinating charged groups. A single misplaced sidechain is not critical and atomic resolution is appropriate even for distorted binding sites. A single atom sterically occluding the binding site is however a severe problem, calling for a low resolution representation of the molecular shape or alternatively for the introduction of flexibility in the model.

4. Conclusion and outlook

MyPal is a method for interactively locating ion binding sites by steering ionic probes into electrostatic potential maps using a haptic device. This multi-resolution approach, coupled with interactive devices is an important improvement of existing methods. This new strategy already emerges as potentially useful for other applications such as docking small ligands on proteins. Here we show how it facilitates the discovery of new relevant ion

binding sites by successfully retrieving the location of cation binding sites in DNase I and assessing their selectivity, combining atomic and coarse-grained resolutions.

The interactive experiments described in this article are particularly well adapted in order to acquire an intuitive understanding of the possible role of counterions in a biomolecular system. They can be carried out without any specific equipment, just using a mouse and on-screen visual feedback. However, we show that tactile feedback is essential for an efficient exploration of complex electrostatic potentials. Generally speaking, immediate feedback from the simulation adds a new dimension of user perception. Haptic feedback in particular is very intuitive and further increases the understanding of a given system. The force resistance allows the user to experience physical effects such as electrostatic steering or steric repulsion while extracting or inserting an ion in a binding pocket. A movie illustrating some interactive *MyPal* experiments on DNase I is available [22].

MyPal is complementary to existing methods such as deterministic or random computer generated searches for ionic binding sites. The human element based on the user's expertise is an essential feature of our approach. At the same time this human element makes it difficult to exhaustively test *MyPal* and compare it to other methods. Different users will use *MyPal* differently and a user's attention will wear off with time. Locating four ion binding sites such as in the case of DNase I remains feasible in a single work session. Such an exploration guided by the expertise of the user should be used prior to more computationally expensive approaches. For example, the 4th ion binding site in DNase I was quickly identified with *MyPal*, whereas it only appeared after extensive equilibration in an MD simulation. The MD simulation does however enable a very detailed and quantitative characterization of ion binding with extensive statistics.

The *MDDriver* library developed in order to couple molecular simulation engines with a visualization tool was used here in the context of a multi-resolution and multi-physics application. *MDDriver* is readily available on http://mddriver.sourceforge.net. The *MyPal* software implementing the simulation methodology presented in this manuscript, as well as the graphical tool designed for interactive simulations and simple visualization tasks based on Cocoa and VTK, will both be made available very soon. Important future improvements may concern visualization, *e.g.* the implementation of volume rendering, and ways to quantify, record and analyze interactive experiments more systematically.

Many future applications for *MyPal* can be imagined. Ions play a particularly central role inside ion channels and their selectivity, gating mechanisms and preferred ionic transport pathways could be investigated *via* interactive simulations. Closer to the example of the DNase I enzyme, the nucleosome still remains uncharted territory when it comes to locating its ion binding sites and assessing their functional role. Ion channels in membrane proteins and ions bound to the nucleosome are good examples of very large systems for which *MyPal* could provide a simple and straightforward method to capture molecular properties. Substantial time savings regarding simulation time and analysis are expected compared to classical approaches using MD. Furthermore, even at the present stage, *MyPal* is not limited to the docking of ions or small ligands, and should therefore be generally applicable in many biological domains. For instance, it opens the prospect to investigate larger molecular assemblies and their interactions. As an example, one could imagine moving the DNase I enzyme described in this manuscript along its DNA substrate to probe sequence specificity. This would be particularly challenging for a nucleosomal DNA substrate.

Acknowledgments

This interdisciplinary work was supported by the DEISA Consortium (co-funded by the EU, FP6 projects 508830/031513) and IDRIS (CNRS's National Supercomputer Center in Orsay). The project was funded by the French Agency for Research (grants ANR-06-PCVI-0025 and ANR-07-CIS7-003-01). We thank Christopher Amourda, Anthony Bocahut, Chantal Prévost and Sophie Sacquin-Mora for providing useful feedback and testing the method presented in this manuscript on a selection of biological systems. Their critical comments enabled us to continually improve the *MyPal* application.

References

1. M. Guéroult, J. Abi Ghanem, B. Heddi, C. Prévost, P. Poulain, M. Baaden and B. Hartmann, *in Albany 2009: Conversation 16* (2009).
2. O. Delalande, N. Férey, G. Grasseau and M. Baaden, *J. Comput. Chem*, in press, published online (2009).
3. H. Nagata, H. Mizushima and H. Tanaka, *Bioinformatics* **18**, 140 (2002).
4. N. Bocquet, H. Nury, M. Baaden, C. Le Poupon, J.P. Changeux, M. Delarue and P. J. Corringer, *Nature* **457**, 111 (2009).
5. R. C. Wade, R. R. Gabdoulline, S. K. Lüdemann and V. Lounnas, *Proc. Natl. Acad. Sci. USA* **95**, 5942 (1998).

6. D. Tworowski, A. V. Feldman and M. G. Safro, *J. Mol. Biol.* **350**, 866 (2005).
7. R.E. Kozack, M.J. d'Mello and S. Subramaniam, *Biophys. J,* **68**, 807 (1995).
8. L. Hemsath, R. Dvorsky, D. Fiegen, M.-F. F. Carlier and M.R. Ahmadian, *Mol Cell* **20**, 313 (2005).
9. C. Oefner and D. Suck, *J. Mol. Biol.* **192**, 605 (1986).
10. W. Kabsch, H. G. Mannherz, D. Suck, E. F. Pai and K. C. Holmes, *Nature* **347**, 37 (1990).
11. S. A. Weston, A. Lahm and D. Suck, *J. Mol. Biol.* **226**, 1237 (1992).
12. D. A. Case *et al.* (2002) AMBER 7 (University of California, San Francisco).
13. T. J. Dolinsky, J. E. Nielsen, J. A. McCammon and N. A. Baker, *Nucl. Acids Res.* **32**, W665 (2004).
14. N. A. Baker, D. Sept, S. Joseph, M. J. Holst and J. A. McCammon, *Proc. Natl. Acad. Sci. USA* **98**, 10037 (2001).
15. M. Zacharias, *Protein Sci.* **12**, 1271 (2003).
16. C. Azuara, H. Orland, M. Bon, P. Koehl and M. Delarue, *Biophys. J.* **95**, 5587 (2008).
17. M. Baaden and R. Lavery, *in Recent Adv. in Protein Engineering*, 173 (2007).
18. W. Humphrey, A. Dalke and K. Schulten, *J. Molec. Graphics* **14**, 33 (1996).
19. N. Férey, O. Delalande, G. Grasseau and M. Baaden, *in Proceedings of the 15th ACM Symposium on Virtual Reality Software and Technology*, 91 (2008).
20. J. E. Stone, J. Gullingsrud, K. Schulten and P. Grayson, *in Proceedings of the 2001 ACM Symposium on Interactive 3D Graphics*, 191 (2001).
21. W. Schroeder, K. Martin and B. Lorensen, The *Visualization Toolkit An Object-Oriented Approach To 3D Graphics, 4th Edition.*
22. http://mddriver.sourceforge.net/
23. R. M. Taylor II, T.C. Hudson, A. Seeger, H. Weber, J. Juliano and A.T. Helser, *in Proceedings of the ACM Symposium on Virtual Reality Software & Technology,* 55 (2001).
24. http://www.gromacs.org
25. D. S. Glazer, R. J. Radmer and R. B. Altman, *Structure* **17**, 919 (2009).
26. C. Claperon, R. Rozenfeld, X. Iturrioz, N. Inguimbert, M. Okada, B. Roques, B. Maigret and C. Llorens-Cortes, *Biochem. J.* **416**, 37 (2008).
27. G. Kupparaj, M. Dudev and C. Lim, *J. Phys. Chem. B* **113**, 2952 (2009).
28. D. Jiao, C. King, A. Grossfield, T. A. Darden and P. Ren, *J. Phys. Chem. B* **110**, 18553 (2006).
29. F. Jalilehvand, D. Spångberg, P. Lindqvist-Reis, K. Hermansson, I. Persson and M. Sandström, *J. Am. Chem. Soc.* **123**, 431 (2001).

PREDICTING RNA STRUCTURE BY MULTIPLE TEMPLATE HOMOLOGY MODELING

SAMUEL C. FLORES[†*], YAQI WAN[°*], RICK RUSSELL[°], RUSS B. ALTMAN[†]

[†]*Bioengineering Department, Stanford University, Clark Center S231, 318 Campus Drive, Stanford, California 94305-5444, USA*

[°]*Department of Chemistry and Biochemistry, Institute for Cellular and Molecular Biology University of Texas at Austin, 1 University Station A4800, 2500 Speedway Austin, Texas 78712, USA*

Despite the importance of 3D structure to understand the myriad functions of RNAs in cells, most RNA molecules remain out of reach of crystallographic and NMR methods. However, certain structural information such as base pairing and some tertiary contacts can be determined readily for many RNAs by bioinformatics or relatively low cost experiments. Further, because RNA structure is highly modular, it is possible to deduce local 3D structure from the solved structures of evolutionarily related RNAs or even unrelated RNAs that share the same module. RNABuilder is a software package that generates model RNA structures by treating the kinematics and forces at separate, multiple levels of resolution. Kinematically, bonds in bases, certain stretches of residues, and some entire molecules are rigid while other bonds remain flexible. Forces act on the rigid bases and selected individual atoms. Here we use RNABuilder to predict the structure of the 200-nucleotide *Azoarcus* group I intron by homology modeling against fragments of the distantly-related *Twort* and *Tetrahymena* group I introns and by incorporating base pairing forces where necessary. In the absence of any information from the solved *Azoarcus* intron crystal structure, the model accurately depicts the global topology, secondary and tertiary connections, and gives an overall RMSD value of 4.6 Å relative to the crystal structure. The accuracy of the model is even higher in the intron core (RMSD = 3.5 Å), whereas deviations are modestly larger for peripheral regions that differ more substantially between the different introns. These results lay the groundwork for using this approach for larger and more diverse group I introns, as well for still larger RNAs and RNA-protein complexes such as group II introns and the ribosomal subunits.

1. Introduction

RNA plays pervasive roles in gene regulation and expression. Messenger RNA provides the template for protein synthesis but also forms structures to regulate that synthesis (1). MicroRNAs inhibit protein production by promoting degradation of their targeted mRNA transcripts or stalling of their translation (2, 3). Even the remarkable machinery that synthesizes proteins, the ribosome, is composed primarily of RNA. Further, recent genomics approaches have indicated that much of the human genome is transcribed and most of it is not translated into protein, suggesting that many more functions of RNA are yet to be discovered (4, 5). The diversity of functional roles for RNA has profound implications for the early development of life, and the elucidation of these functions holds the promise of novel treatments for human diseases (6).

However, our understanding of RNA structure and function is continually hampered by a persistent lack of structural coordinates. Part of the challenge arises from the experimental difficulty of crystallizing a highly charged, very flexible molecule with a dearth of the distinctive surface features needed for specific crystal packing (7). Compounding the problem, many structured RNAs can adopt alternative conformations at equilibrium or as long-lived kinetic traps during folding (8, 9). Theoretical approaches are also challenged by these features, both by the delicate energetic balance between alternative conformations and by the long times required to equilibrate during folding.

The structures of relatively small RNAs can often be predicted by one of several methods. Fragment Assembly of RNA (FARNA) assembles structures by sampling trinucleotide fragments from a database and screens these structures using a coarse grained potential that favors base pairing and stacking geometries (10). Similarly, MC-Sym samples four-nucleotide cycles from a database that are consistent with known base pairing contacts and

[*] These authors contributed equally to the work.

progressively builds up structure (11). Both of these methods, however, have computer time requirements that scale poorly with size, and rely on fragment databases that include only limited diversity. They therefore have not been shown to predict the structure even of molecules as large as tRNA (at ~75 nucleotides), except when using a fragment library that contains tRNA (12). Discrete Molecular Mechanics (DMD) has a simplified potential for RNAs, which are represented by three pseudoatoms per nucleotide; it can solve the structure of tRNA but larger molecules remain out of reach. The Nucleic Acid Simulation Tool (NAST) uses one pseudoatom per nucleotide to represent RNA structure and can fold the (~150 nt) P4/P6 domain of the *Tetrahymena* group I intron (13). However, its force field is too coarse to discriminate the native state from decoys at larger size scales.

Homology modeling can be used to predict the structure of larger molecules when structural homologs are available. The first step is to obtain a correspondence between residues in the molecule to be modeled and a template; if the sequence identity is low this process must be done manually. The next step is to geometrically align residues from the model onto corresponding residues in the template. Lastly, the structure of any inserted regions must be solved and deletions must be closed. Kevin Sanbonmatsu and collaborators have threaded the *E. coli* 16S ribosomal RNA onto a template from *Thermus Thermophilus*. The model and template have 75% sequence identity, considerably higher than that between *Azoarcus* and *Twort* ribozymes (<50%). The only insertions are in nine hypervariable regions that are not in contact with each other. These were dealt with by adding fragments from additional structures. Since the model does not have the flexibility needed to align and form long range contacts, it would be difficult to build structures that are interconnected in regions with no single structural homolog, as occurs with the *Azoarcus* ribozyme. Further, there is no code distributed for this method, leaving the technique to be applied by computational experts (14).

There is therefore a need for a homology modeling program that can *1)* structurally align corresponding residues while allowing close user control over the threading, *2)* incorporate templates from a third molecule or molecules, *3)* solve the structure of connecting regions with no template by enforcing sterics, chemistry, and base pairing while allowing flexibility and *4)* be accessible to the experimentalist. In the current work we describe a multi-resolution modeling approach (see Background) that does all of this.

Requirement *1* is met by applying forces which align the threaded and template bases. These are specified by the user just like the base pairing forces described below. Requirement *2* is met by adding more templates and connecting them to the threaded molecule. The molecules are completely rigid, a feature of our multi-resolution modeling (MRM) approach.

Requirement *3* is the ability to predict structures of connecting regions by applying steric exclusion and base-pairing forces while maintaining bond lengths and angles. Our method requires secondary and tertiary base pairing contacts, which are provided by the user in a simple format. Each of these contacts becomes a term in our force field, with force and torque components, which act to bring the paired bases into the indicated geometry. Simulated annealing is then used to escape kinetic traps or local energy minima. The kinematics treats each backbone atom as an independent body and the bases as rigid units. In parallel, the base pairing forces act on a single atom per base, and the sterics are treated with contact spheres on a few atoms per nucleotide. The parallel treatment of forces and mobilities at different levels of resolution is another MRM aspect of our method. RNABuilder can solve the structures of small connecting regions without a template and does not compute interactions between all near neighbors, thus addressing the scaling issues inherent in both fragment assembly and Molecular Dynamics(MD)-like methods.

Requirement *4* is met by providing an easy-to-use interface that was used in this work to specify all parameters of our model, including the sequences, base-pairing contacts, and correspondence between the model and templates. A single input file is prepared by the user to provide the RNA sequences, base-pairing contacts, and simulation

parameters. A tutorial is provided (*Distribution* section), which includes example input files to use as a starting point for the user's own runs. Although no significant computational skill is required, modeling decisions require molecule-specific knowledge which an experimentalist would be expected to have about his or her system.

Here, we use RNABuilder to model the structure of a large, multi-domain RNA, the 200-nucleotide *Azoarcus* group I intron, by homology modeling with multiple templates. Group I introns have been powerful model systems for understanding RNA structure, folding, and function since they were discovered as the first catalytic RNAs more than 25 years ago (9, 15-18). By comparing our model to the solved crystal structure of the intron, we show that the model correctly captures the domain structures, connections and topology and gives an overall RMSD of 4.6 Å.

2. Background

2.1. *Classification of multi-resolution modeling (MRM) techniques*

Multi-resolution modeling (MRM) has emerged as a useful paradigm. MRM refers to the modeling of molecules at coarse grained resolution, with direct or indirect connection to the corresponding atomistic model (in which each atom is an independent body). According to the Ayton, Noid, and Voth classification (19), such schemes can be either serial or parallel. In a serial scheme, the parameters and form of the coarse grained model can be developed from the atomistic one (S-A), from various sources of knowledge (S-B), or from thermodynamic data (S-C). In a parallel scheme, the coarse and atomistic models can interact (P-1) or run concurrently under a resolution exchange methodology (P-2). RNABuilder is an MRM method of type P-1, since the kinematics (which in certain places are atomistic, and in other places very coarse) and the forces (mostly coarse grained) form part of a single system.

2.2. *Multibody dynamics*

Internal coordinate multibody dynamics is a method for simulating molecules. Instead of storing the Cartesian coordinates of each atom as would be done in a conventional MD simulation, one represents the state of the molecule via its natural "internal coordinates", such as bond lengths and torsional angles (20). It then becomes trivial to constrain any internal coordinate: one simply keeps it fixed, only integrating the unconstrained degrees of freedom. Internal coordinate multibody dynamics is much more difficult to implement than traditional MD, since it is necessary to transform atom locations to Cartesian coordinates before calculating forces, to transform the forces back to internal coordinates again before integrating, and to calculate the effective inertia of each coordinate based on the current state of the molecule at each time step; however there exist efficient linear time algorithms for performing these operations. When a system is highly constrained, internal coordinate multibody dynamics is far more efficient than Cartesian MD. In fact, the more highly constrained the system is, the more efficient it becomes, in direct contrast to Cartesian constraint algorithms which become less efficient as constraints are added (21).

2.3. *Simbody and Molmodel*

Our RNABuilder package is written using the Simbody (22) internal coordinate mechanics library and its molecular mechanics extension, Molmodel, both available from SimTK.org. Simbody includes variable step size integrators (23), which continually attempt to maximize the integration step size without exceeding error tolerances, leading to significant time savings as well as stable behavior when dealing with large forces. Molmodel permits us to rigidify all template molecules and part of the modeled structure saving the computational expense often associated with modeling large structures. RNA bases can be modeled as rigid bodies and so base pairing forces and torques can be applied to the base rather than to individual atoms, reducing the number of calculations to be performed. Lastly, Simbody provides collision-detecting Contact spheres (24), which we use on selected atoms as an approximate treatment of sterics.

2.4. *RNABuilder force field, mobilizers, and constraints*

The coarse grained force field used in this work consists of forces and torques which act to bring the interacting bases into the base pairing geometry specified by the user using the *baseInteraction* commands. No forces act between bases unless specified by the user, except stacking forces which are automatically added to helices. The first base has an attachment frame which is part of the glycosidic nitrogen body but which is often located several angstroms from the nucleus, and which has a specified relative angular orientation. The second base has only a body frame located at the glycosidic nitrogen nuclear center and rotated such that the x-axis lies along the glycosidic bond axis and the z-axis lies normal to the base plane and points in the 3' direction, assuming helical geometry. The forces act to pull the attachment and body frames together in cartesian space, and the torques act to align them rotationally. Since the bases are rigid by default this is sufficient to bring all atoms into a desired base pairing geometry. Parameterization of the force field thus consists primarily of determining the translation and rotation of the attachment frame that will result in the base pairing geometry; a program is provided to compute these parameters but most users will simply use the distributed parameters. These parameters include those corresponding to all base pairs catalogued by Leontis et al (including the WatsonCrick) (25), plus stacking and a Superimpose interaction used in threading to align the threaded with the template base. See Methods for enforcement of these interactions.

The geometric relationship between attachment and body frames is enforced by means of a potential that has a strength parameter cutoffPotential (user adjustable in the RNABuilder parameter file) based loosely on base pairing enthalpies (26, 27). It also has a range parameter cutoffRadius (set by the user in the RNABuilder input file) based on ranges reported for base pairing and stacking forces (10, 27). RNABuilder detects runs of three or more consecutive WatsonCrick base pairs and automatically enforces helical geometry. The potential is harmonic at close range and inverse at long range. The requirement that the potential and its derivative match at cutoffRadius leads to the following form:

$$U_{translational}(r) = \begin{cases} \frac{cutoffPotential}{2} \cdot (3 - \frac{r^2}{cutoffRadius^2}), 0 \le r < cutoffRadius \\ \frac{cutoffPotential \cdot cutoffRadius}{r}, r \ge cutoffRadius \end{cases}$$

The force is obtained by differentiation. These quantities are plotted in Figure 1 below.

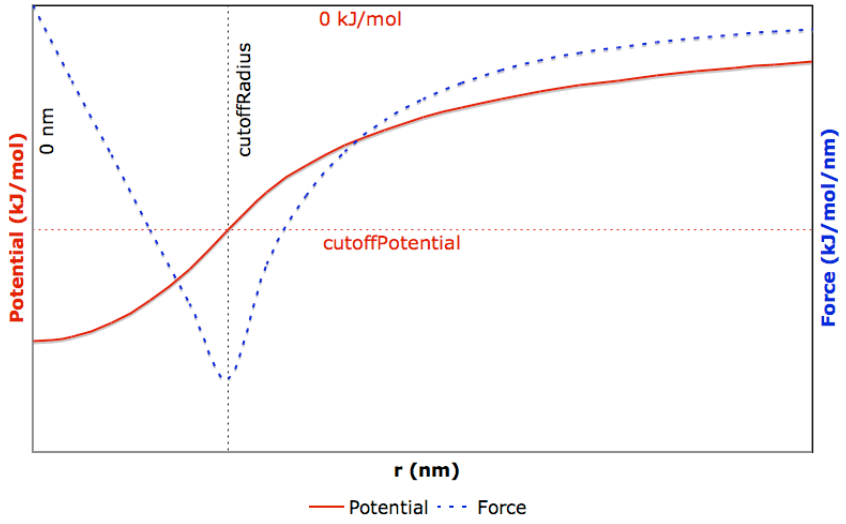


Figure 1. Potential and force as a function of distance from *attachment frame* of residue 1 to *body frame* of residue 2.

We provide a *contact* force, which can attach spheres to selected atoms to prevent steric clashes. These spheres interact repulsively when they overlap. The HardSphere keyword specifies that spheres be added to 2-3 atoms per residue, for any specified range of residues. We used this on selected residues to push apart overlapping strands.

RNABuilder also provides a series of *mobilizer* commands which set the flexibility of the molecule. The default mobilizer rigidifies all base bonds; most backbone bond lengths and angles are fixed but dihedral angles are allowed to vary. This guarantees that all bonds will have reasonable local geometry. In this work we use the 'Rigid' mobilizer to rigidify the template molecules, and parts of the threaded molecule which at certain stages were considered to be converged. The Rigid mobilizer reduces computational expense since it simply removes certain degrees of freedom rather than imposing constraint equations which must then be solved.

Lastly, RNABuilder includes a *constraint* called 'Weld' which fixes the C3' carbon of any residue to that of any other. We used this where two separate chains had been rigidified and needed to be held together. A constraint comes at a cost since it adds an equation that must be solved in Simbody.

3. Methods

The *Azoarcus* group I intron sequence was aligned and threaded to the *Twort* intron (Fig. 2) (28). The RNAs include secondary structure features that are numbered by convention and designated P (helix), J (junction), or L (loop). Direct correspondences are present over most of the length of the RNA, allowing straightforward threading of secondary elements P3, P4, P6, and P7, as well as several internal lops and single-stranded joining segments (J3/4, J6/7, J8/7, J4/5, and J6/6a). However, in regions that are sufficiently different between the two RNAs that direct correspondences are not present, alternative methods were used. Two 'tetraloop-receptor' tertiary interactions were threaded against a fragment of the Tetrahymena intron that includes the same type of interaction. For three short connecting segments, no existing group I intron structures were identified with corresponding sequences in the same positions and with the same numbers of nucleotides, and these segments were therefore left unthreaded (shown in black in Fig. 2). Similarly, two hairpin loops were not threaded because structures with the same sequences were not available.

3.1. *Modeling the P1-P2 stacked helix*

The P1 helix, which includes the strand that is cleaved during splicing, is three base pairs in the *Azoarcus* intron but four base pairs in the *Twort* intron. However, this length difference is compensated by a difference in P2, which is one base pair longer in the *Azoarcus* intron. Therefore, the first base pair of the *Azoarcus* P2 was threaded to the fourth base pair of P1 in the *Twort* structure. Because the final base pair of the *Azoarcus* P2 element was modeled onto P1, but the joining sequence J2/3 emanates from P2, J2/3 was not threaded.

3.2. *Modeling the P9 region*

From its sequence, the *Azoarcus* intron has the potential to form a P9.0 helix consisting of two G-C base pairs, but on the 5′-side there are three consecutive Gs, and it is not clear from the sequence which two form these base pairs (see Fig. 2). We modeled base pairs by the two 5′-most Gs, which allowed threading of the single nucleotide linking P7 and P9.0. From P9.0 to P9 on the 5′-side, there are two unpaired nucleotides in the *Azoarcus* intron sequence. We threaded these nucleotides into P9, where they could form one canonical and one non-canonical base pair, generating a P9 helix that would be the same length as in the *Twort* intron. The *Azoarcus* intron has three unpaired nucleotides connecting P9 to P9.0 (J9/9.0), whereas the *Twort* intron has seven unpaired nucleotides connecting in the corresponding junction. These three nucleotides were threaded onto the three nucleotides immediately upstream of P9.0 in the *Twort* intron.

3.3. *Modeling tetraloop-receptor interactions using additional templates*

It was not possible to thread two tetraloop-receptor interactions of the *Azoarcus* intron. One of these sites was disordered in the *Twort* intron crystal structure and the other site has a different type of receptor (29, 30). We threaded both of these interactions using the structure of the P4-P6 domain of the *Tetrahymena* ribozyme (31), which has the same type of tetraloop and receptor as the *Azoarcus* intron. After completing an initial round of all of the modeling steps (see below), we observed that one of the loops, L2, had moved so that it approached its receptor too closely to be physically reasonable. We therefore reinforced the correct conformations in this contact by again threading these regions to the *Tetrahymena* intron template (see Results).

3.4. *Hairpin loops*

Some hairpin loops could not be threaded to the *Twort* intron because of natural or engineered differences. Two such hairpins, P5a and P8a, were threaded to sequences in the *Tetrahymena* intron as described above. A third, P6a, was omitted and modeled as a blunt-ended helix.

3.5. *Using RNABuilder to perform threading*

We began by threading the flexible *Azoarcus* group I ribozyme onto the *Twort* ribozyme using the RNABuilder commands described in the Background section. There is a gap in the *Twort* structure where the distal end of P5 is deleted; accordingly we modeled this template using two strands which were then rigidified and welded to each other. We used the 'Superimpose' twoTransformForce to pull together bases in *Azoarcus* and *Twort* that were aligned as described above. Of the residues that have no equivalent in *Twort*, some were left to be folded by threading to additional templates as described above. Others were in known helical regions and modeled by applying 'WatsonCrick' twoTransformForces. The 'HardSphere' contact was applied only to residues where a steric clash occurred.

The model we obtained in this step was a global 3D structure with P1/2, P4-P6, P3-P7 and P9 formed exactly as *Twort* ribozyme but with several other portions not yet folded. The latter included the tetraloop-receptor in L2-P8, especially the receptor part in P8, since there is no template in *Twort* for this type of receptor. *Azoarcus* L9-P5 is similar to *Twort* but as mentioned the top of P5 stem including half of the receptor was distorted and deleted from the crystal structure. Lastly, the distal end of P6a in the *Azoarcus* intron construct used for crystallization has an engineered U1A loop, which of course is not present in the *Twort* intron or in the natural *Azoarcus* intron. We introduced a chain break in the model between nucleotides 109 and 110 to circumvent the engineered loop.

To model L2-P8 and L9-P5, in the second stage, we introduced as additional templates two copies of the L5b and P6 fragments from the *Tetrahymena* ribozyme. In each copy, L5b and P6 were rigidified using the 'Rigid' mobilizer and welded together using the 'Weld' constraint. Then 'Superimpose' baseInteractions were used to attach one copy each to the ends of the *Twort* P8 and P5 helices. With these prostheses the template now had the desired motifs in L2-P8 and L9-P5. The appropriate *Azoarcus* intron nucleotides were then threaded onto the corresponding nucleotides of the *Tetrahymena* intron fragment.

4. Results

To use homology modeling to predict the structure of the *Azoarcus* group I intron, we first aligned the secondary structure of the *Azoarcus* intron, established by comparative sequence analysis, to a version of the *Twort* secondary structure that was validated by its crystal structure (28) (Fig. 2). Although the crystal structure of the *Azoarcus* intron has been solved (32, 33), we did not use any information from this structure in the alignment or in the subsequent modeling. Despite representing different subgroups – the *Azoarcus* is designated as a IC3 intron and the *Twort* intron is a IA2 intron – the core elements of secondary structure and their connections are conserved (34). Further, many of these secondary structure elements and connections are identical in length between the two introns

and were aligned in a straightforward manner. Although the *Twort* intron includes some helical elements that are not present in the *Azoarcus* intron, the *Azoarcus* intron is not as large and complex as many group I introns, and all of its helical elements are present in the *Twort* intron.

Nevertheless, some structural features differ significantly between the two introns, such that it was not possible to directly thread certain local regions to the *Twort* intron structure (Fig. 3; see Methods). Most notably, two tertiary contacts in the *Azoarcus* intron are formed by canonical tetraloop-receptor interactions between a GAAA tetraloop and a receptor that includes an internal loop and has been termed the 11-nucleotide receptor (29). One of these contacts is between L2 and P8 and the other is between L9 and P5. Although the *Twort* intron has tetraloop-receptor interactions at equivalent positions, the interacting partners are of different structural classes. Instead of being the 11-nucleotide receptor, one of the receptors consists of two Watson-Crick base pairs, and the tetraloop for this receptor is GUAA instead of GAAA. The other receptor, in P5, includes an internal loop and interacts with a GAAA tetraloop. However, it is a non-canonical version of the 11-nucleotide receptor, and it could not be used as a template regardless because it was disordered in the crystal structure (28).

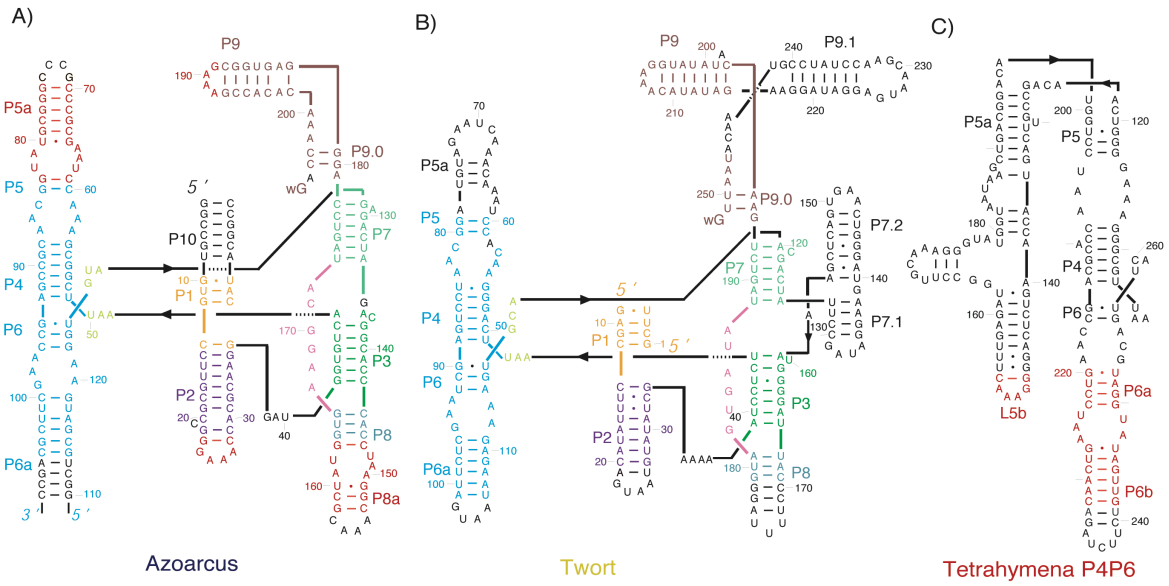


Figure 2. The secondary structures of A) the *Azoarcus* intron, B) the *Twort* intron, and C) the P4-P6 domain of the *Tetrahymena* intron. The domains are color-coded, with like colors indicating a correspondence between the *Azoarcus* model and the *Twort* or *Tetrahymena* intron templates. This correspondence was used as the basis for threading. Note that the tetraloop-receptor structure from the interaction of L5b with J6a/6b of the *Tetrahymena* intron (orange) was used as a template for both tetraloop-receptor contacts within the *Azoarcus* intron. Regions that were not threaded are shown in black.

To model these tetraloop-receptor interactions, we took advantage of the presence of the canonical 11-nucleotide receptor in the P4-P6 domain from the *Tetrahymena* intron, whose structure has been solved by x-ray crystallography (31). We aligned the *Tetrahymena* intron fragment to the *Twort* intron structure by using the base pairs adjacent to the receptor motif and then threaded the tetraloop and receptor sequences from the *Azoarcus* intron onto the corresponding nucleotides within the *Tetrahymena* intron fragment (Fig. 3; see also Fig. 2).

The final model of the *Azoarcus* intron was evaluated by superimposing it against the structure of the intron determined by crystallography (Fig. 4). The model has an RMSD with respect to the crystallographic coordinates of 4.59 Å (Table 1). The topology is correct throughout the intron, and the placement of helices is also largely correct. The closest overall agreement is found in the core region, with an RMSD value of 3.54 Å. The active site has a slightly higher RMSD value of 3.70 Å.

The peripheral regions also give a good overall agreement with the structure, as illustrated in Fig. 4. Nevertheless, larger differences were observed in the periphery than in the core. The receptor within P8, which was modeled using the *Tetrahymena* intron fragment, was positioned correctly and maintained the correct local structure, as expected from the known structural conservation of this tetraloop-receptor motif (31-33). However, a portion of P2 is shifted downward along its axis relative to its position in the crystal structure. Because the interaction of the tetraloop with the receptor in P8 was enforced by threading directly to the *Tetrahymena* intron fragment, the shift of P2 results in a minor, local distortion of P2 close to its distal end (Fig. 5A). Intriguingly, the shift of P2 can also be seen for the *Twort* intron structure relative to the *Azoarcus* intron structure (data not shown), suggesting that the subtle structural mismatch in this region, relative to the *Azoarcus* intron, is inherent to the *Twort* intron template.

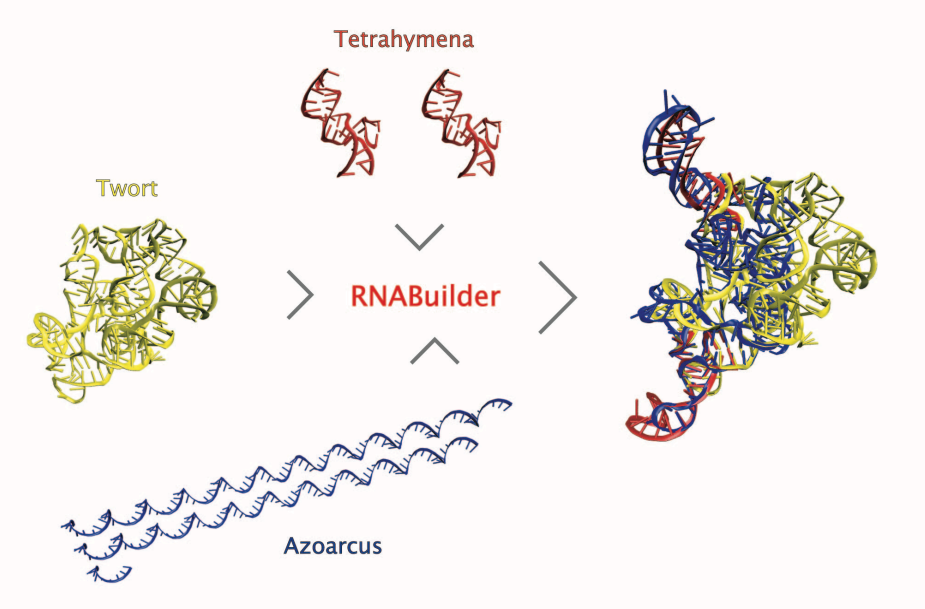


Figure 3. Modeling the *Azoarcus* intron (blue) by threading to fragments of the *Twort* and *Tetrahymena* intron structures. Three rigid fragments were used as templates: the nearly-complete intron from *Twort* (left, yellow) and the tetraloop receptor from the P4-P6 domain of the *Tetrahymena* intron (top, orange). The fragments corresponding to the *Azoarcus* intron were initially in extended conformations (bottom). The final model of the threaded *Azoarcus* intron, superimposed on the *Twort* and *Tetrahymena* template fragments, is shown at right.

Table 1

Region	Nucleotides	RMSD (Å)
Entire intron	4-206	4.59
Core: P1, P3, P4, P6, P7, P8, P8a	10-12, 41-59, 86-99, 120-153, 158-178, 206	3.54
Active site	8-12, 85-89, 126-132, 137, 168-181, 180-181, and 203-205, and substrate residues -3 to +2	3.70
Periphery: P2, P5, P5a, P6a, P9, P9.0	13-40, 60-84, 100-119, 154-157, 179-205	5.40

There are also small but significant differences in the P9 region. Notably, nucleotides G182 and A183, which were threaded into P9 of the *Twort* intron, do not form P9 pairs in the *Azoarcus* intron crystal structure (Fig. 5B). Instead,

they stack with the preceding nucleotides, forming an extension of P9.0 with non-canonical base pairs to A201 and A202. In large part because of this base-pairing difference, the RMSD value for P9.0 and P9 ranges above 5.0 Å, well above the average (Table 2). Nevertheless, the global and even local architectural features of this region are intact, with a sharp bend from P9.0 to P9 and the formation of a tetraloop-receptor contact with P5.

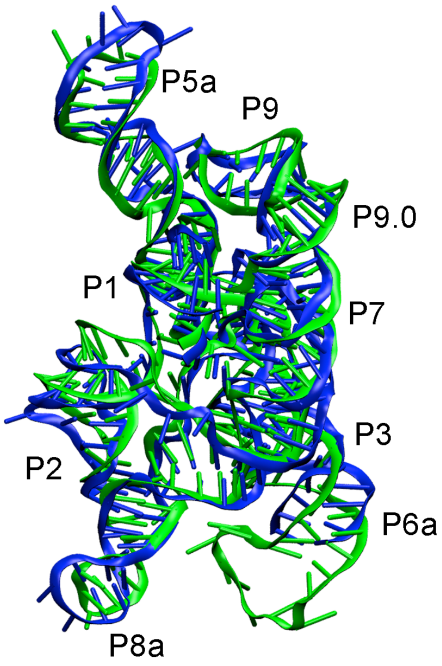


Figure 4. Model of the Azoarcus intron (blue) superimposed on the structure determined by x-ray crystallography (green). Visible helices are labeled (see Fig. 2).

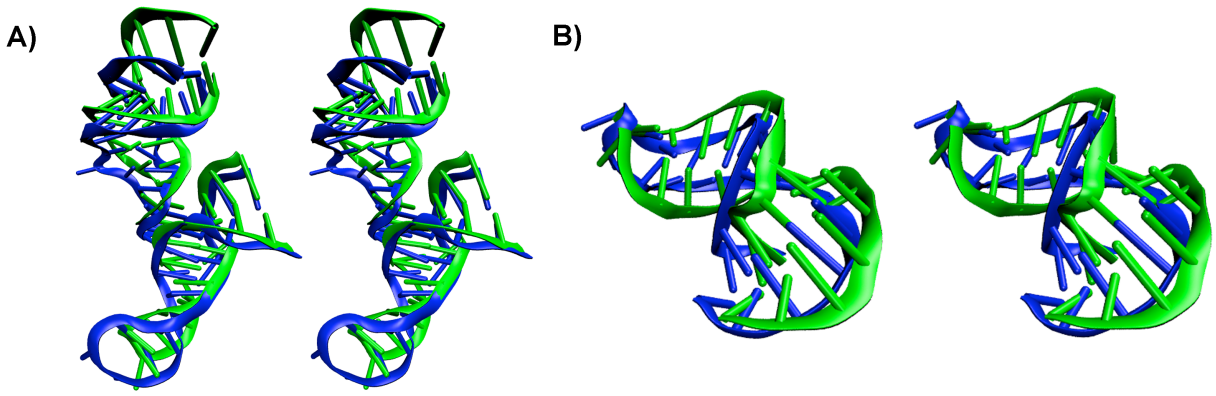


Figure 5. Regions of the Azoarcus intron model (blue) superimposed on the corresponding regions of the crystal structure (green). A) Tetraloop-receptor interaction of L2 and P8. B) P9 and P9.0. Each region is shown in wall-eyed stereoview in the same orientation as Figure 3.

Table 2

Region	Nucleotide range	RMSD (Å)
P1	10-12, 1-3 of the substrate oligonucleotide	2.55
P2	13-37	4.34
J2/3	38-40	7.20
P3	41-47, 136-143	3.45
J3/4	48-50	5.55
P4	51-56, 88-93	3.11
J4/5	57-59, 85-87	2.75
P5/5a	60-84	5.62
P6	94-96, 122-124	3.37
J6/6a	97-99, 120-121	3.22
P6a	100-119, 108-111 omitted	4.90
J6/7	125-127	3.94
P7	128-134, 173-178	4.01
P8	144-166	4.34
J8/7	167-172	3.66
P9.0	179-181, 203-205	4.90
J9/9.0	182-183, 198-202	6.00
P9	184-197	5.96
wG	206	3.05

5. Discussion

We used the RNABuilder software package to predict the structure of the *Azoarcus* group I intron by homology modeling to multiple templates. RNABuilder is an MRM method in that the templates, bases, and converged part of the model are rigidified, while forces are simultaneously treated at multiple levels of granularity. The model generated using this method displayed good agreement to the solved crystal structure of the *Azoarcus* intron, suggesting that this approach will be useful for a variety of applications in the structural biology of RNA.

RNABuilder takes advantage of several features of the SimTK core toolkit (https://simtk.org), including multibody mechanics and the Contact subsystem, which make the software well-suited for applications such as that described here. Because they permit accurate physical simulation of constrained models, these features are valuable for generating structural predictions by modeling against known RNA structures or fragments. Multibody mechanics is faster for simulations with many constraints, whereas in Cartesian mechanics constraints add computational cost. In homology modeling it is useful to constrain portions of the structure while others are built, making multibody mechanics the logical choice. For the regions that are allowed to move, the contact spheres used by RNABuilder are an economical way to approximately detect and prevent steric clashes. With these features, RNABuilder is an exceptionally efficient software package for modeling structures of large, highly constrained systems like group I introns and other structured RNAs.

In determining how to model the *Azoarcus* intron, we took steps to ensure that we were not inadvertently introducing information from the *Azoarcus* intron crystal structure. For the alignments, we only used a secondary structure model of the *Azoarcus* intron that was generated before the structure was solved (28). For regions that required decisions about how to thread them to the *Twort* intron structure, we adopted a systematic approach in which we maximized the correspondence to the *Twort* intron. In the region of P9, where the correspondences were the least clear (see Methods), this strategy resulted in successful prediction of the P9.0 base pairs but caused us to incorrectly predict two base pairs within P9. The tetraloop-receptor partners, which we modeled by threading to the

Tetrahymena intron fragment, are identical to the *Tetrahymena* intron sequences and had been predicted to share the same structure (28).

Although the agreement between the model and the *Azoarcus* crystal structure was strong throughout the molecule, it is instructive to consider the regions that gave the highest similarity and the regions that were lower. The most striking agreement was found within the core of the intron (Table 1), as might be expected because this is the most highly conserved region. Therefore, threading to even the distantly-related group I intron from *Twort* gave a good structural model for the *Azoarcus* ribozyme. The RMSD in the active site is somewhat higher, mostly because this region contains P9.0 and J9/9.0 which had higher disagreement (see below). On the other hand, the agreement was less strong in the periphery, where the conservation between the *Azoarcus* and *Twort* introns is much weaker.

Lower accuracy was also observed in regions where the peripheral architectures differ between the introns. The *Twort* intron possesses additional peripheral elements, relative to the *Azoarcus* intron, which influence the connections between structural domains. Specifically, the junction between P9 and P9.0 is part of a three-helix junction in the *Twort* intron, whereas it is a two-helix junction in the *Azoarcus* intron (see Fig. 2). Analogously, the connection between P7 and P3 is part of a complex multi-helix junction in the *Twort* intron, whereas it is a simple stacked-helix connection in the *Azoarcus* intron. In light of these substantial structural differences, it is striking that the threading approach can give as good agreement as it does in these regions. Presumably, an important factor favoring accurate modeling here is that the *Azoarcus* intron possesses a minimal set of peripheral structure elements, and it is therefore essential for the function of the intron that each peripheral element conform to the orientations and contacts found in homologous structural elements in other introns.

The success of the modeling here suggests that this approach is likely to be useful for modeling of even larger and more complex RNAs. The use of internal coordinates (in which dynamics are computed in linear time) and a force field consisting only of user-imposed interactions (avoiding long-range physical forces whose calculation scales poorly) means that significantly larger molecules can be treated with only a proportionate increase in cost. Further, some portions converge early and these can be rigidified while the rest of the threading continues, reducing computational cost. In this work the helices of azoarcus that were not threaded were formed by manually specifying base pairing contacts; by extension larger regions or even entire molecules could potentially be formed by specifying a sufficient number of such contacts. Alternatively these regions could be formed using MC-Sym, DMD, or NAST, which can build molecules in the 50-150 residue size range.

In one potential extension of this work, recent crystal structure determination and model of group II introns may prove useful for modeling of distantly related group II introns (35, 36). It may also be possible to model the ribosomal subunits as well as the much larger complete ribosomes, viral genomes (37), and spliceosomal complexes using this method. Key to the extension of our method is the ability to model larger regions of the molecule without templates and to handle such molecules without excessive computer time and memory requirements. The existing code makes substantial progress in these directions. In future work, continued improvements in speed, memory usage, and accuracy of the force fields will be useful for modeling of larger molecules and complexes.

6. Distribution

A binary distribution of RNABuilder is available for download from the RNAToolbox project on the Stanford Simbios Center's website, SimTK.org. A tutorial is available in the "Downloads" section. Simbios also provides software support and workshops.

Acknowledgements

We thank Chris Bruns, Michael Sherman, Jack Middleton, and Mark Friedrichs for substantial explanations and help with Simbody, as well as adding useful features to that code for our use. We also thank Charles Janac for help with the parameterization of the RNABuilder force field. S. Flores is supported by Simbios, the NIH Roadmap for Medical Research; Grant number: U54 GM072970 to R.B.A. This work was also supported by grants to R.R. from the National Institutes of Health (GM 070456), the Welch Foundation (F-1563), and the Norman Hackerman Advanced Research Program (003658-0242-2007).

References

1. A. Roth and R.R. Breaker, *Annu. Rev. Biochem.,* 2009, **78**, 305-334.
2. D.P. Bartel, *Cell,* 2004, **116**, 281-297.
3. R.W. Carthew and E.J. Sontheimer, *Cell,* 2009, **136**, 642-655.
4. S. Katayama, et al., *Science,* 2005, **309**, 1564-1566.
5. P. Carninci, et al., *Science,* 2005, **309**, 1559-1563.
6. T.A. Cooper, L. Wan, and G. Dreyfuss, *Cell,* 2009, **136**, 777-793.
7. A.R. Ferre-D'Amare, K. Zhou, and J.A. Doudna, *J. Mol. Biol.,* 1998, **279**, 621-631.
8. D.K. Treiber and J.R. Williamson, *Curr. Opin. Struct. Biol.,* 1999, **9**, 339-345.
9. R. Russell, *Front. Biosci.,* 2008, **13**, 1-20.
10. R. Das and D. Baker, *Proc. Natl. Acad. Sci. U. S. A.,* 2007, **104**, 14664-14669.
11. M. Parisien and F. Major, *Nature,* 2008, **452**, 51-55.
12. F. Major, D. Gautheret, and R. Cedergren, *Proc. Natl. Acad. Sci. U. S. A.,* 1993, **90**, 9408-9412.
13. M.A. Jonikas, et al., *RNA,* 2009, **15**, 189-199.
14. C.S. Tung, S. Joseph, and K.Y. Sanbonmatsu, *Nat. Struct. Biol.,* 2002, **9**, 750-755.
15. K. Kruger, et al., *Cell,* 1982, **31**, 147-157.
16. T.R. Cech, *Annu. Rev. Biochem.,* 1990, **59**, 543-568.
17. G.J. Narlikar and D. Herschlag, *Annu. Rev. Biochem.,* 1997, **66**, 19-59.
18. S.A. Strobel and J.A. Doudna, *Trends Biochem. Sci.,* 1997, **22**, 262-266.
19. G.S. Ayton, W.G. Noid, and G.A. Voth, *Curr. Opin. Struct. Biol.,* 2007, **17**, 192-198.
20. N. Vaidehi, A. Jain, and W.A. Goddard, 3rd, *J. Phys. Chem.,* 1996, **100**, 10508-10517.
21. W.F. van Gunsteren and H.J.C. Berendsen, *Molecular Biophysics,* 1977, **34**, 1311-1327.
22. J.P. Schmidt, et al., *Proceedings of the IEEE,* 2008, **96**, 1266-1280.
23. R.E. Crosbie and W. Heyes, *Appl. Math. Modeling,* 1976, **1**, 137-140.
24. M.C. Lin, *Ph.D. thesis, University of California, Berkeley, CA,* 1993,
25. N.B. Leontis and E. Westhof, *Curr. Opin. Struct. Biol.,* 2003, **13**, 300-308.
26. S.M. Freier, et al., *Proc. Natl. Acad. Sci. U.S.A.,* 1986, **83**, 9373-9377.
27. E. Stofer, C. Chipot, and R. Lavery, *J. Am. Chem. Soc.,* 1999, **121**, 9503-9508.
28. B.L. Golden, H. Kim, and E. Chase, *Nat. Struct. Mol. Biol.,* 2005, **12**, 82-89.
29. M. Costa and F. Michel, *EMBO J.,* 1995, **14**, 1276-1285.
30. M. Costa and F. Michel, *EMBO J.,* 1997, **16**, 3289-3302.
31. J.H. Cate, et al., *Science,* 1996, **273**, 1678-1685.
32. P.L. Adams, M.R. Stahley, A.B. Kosek, J. Wang, and S.A. Strobel, *Nature,* 2004, **430**, 45-50.
33. P.L. Adams, et al., *RNA,* 2004, **10**, 1867-1887.
34. J.J. Cannone, et al., *BMC Bioinformatics,* 2002, **3**, 2-32.
35. N. Toor, K.S. Keating, S.D. Taylor, and A.M. Pyle, *Science,* 2008, **320**, 77-82.
36. L. Dai, et al., *Mol. Cell,* 2008, **30**, 472-485.
37. M.J. Roossinck, D. Sleat, and P. Palukaitis, *Microbiol. Rev.,* 1992, **56**, 265-279.

CONSTRUCTING MULTI-RESOLUTION MARKOV STATE MODELS (MSMS) TO ELUCIDATE RNA HAIRPIN FOLDING MECHANISMS

XUHUI HUANG[†]
Department of Chemistry, The Hong Kong University of Science & Technology
Kowloon, Hong Kong
Department of Bioengineering, Stanford University
Stanford, CA, 94305, U.S.A.

YUAN YAO
School of of Mathematical Sciences, Peking University
Beijing, China, 100871
Department of Mathematics, Stanford University
Stanford, CA, 94305, U.S.A.

GREGORY R. BOWMAN
Biophysics Program, Stanford University
Stanford, CA, 94305, U.S.A.

JIAN SUN
Department of Computer Science, Stanford University
Stanford, CA, 94305, U.S.A.

LEONIDAS J. GUIBAS
Department of Computer Science, Stanford University
Stanford, CA, 94305, U.S.A.

GUNNAR CARLSSON
Department of Mathematics, Stanford University
Stanford, CA, 94305, U.S.A.

VIJAY S. PANDE
Department of Chemistry, Stanford University
Stanford, CA, 94305, U.S.A.

Simulating biologically relevant timescales at atomic resolution is a challenging task since typical atomistic simulations are at least two orders of magnitude shorter. Markov State Models (MSMs) provide one means of overcoming this gap without sacrificing atomic resolution by extracting long time dynamics from short simulations. MSMs coarse grain space by dividing conformational space into long-lived, or metastable, states. This is equivalent to coarse graining time by integrating out fast motions within metastable states. By varying the degree of coarse graining one can vary the resolution of an MSM; therefore, MSMs are inherently multi-resolution. Here we introduce a new algorithm Super-level-set Hierarchical Clustering (SHC), to our knowledge, the first algorithm focused on constructing MSMs at multiple resolutions. The key insight of this algorithm is to generate a set of super levels covering different density regions of phase space, then cluster each super level separately, and finally recombine this information into a single MSM. SHC is able to produce MSMs at different resolutions using different super density level sets. To demonstrate the power of this algorithm we apply it to a small RNA hairpin, generating MSMs at four different resolutions. We validate these MSMs by showing that they are able to reproduce the original simulation data. Furthermore, long time folding dynamics are extracted from these models. The results show that there are no metastable on-pathway intermediate states. Instead, the folded state serves as a hub directly connected to multiple unfolded/misfolded states which are separated from each other by large free energy barriers.

[†]To whom correspondence should be addressed. E-mail: xuhuihuang@gmail.com

1. Introduction

Conformational changes are crucial for a wide range of biological processes including protein folding[1], RNA folding[2] and the operation of key cellular machinery[3-5]. Extensive genetic, biochemical, biophysical and structural experiments can be performed to understand these conformational changes[3-5]. However, probing the mechanisms of conformational changes at atomic resolution is very difficult experimentally and without these details it is impossible to understand the fundamental chemistry they perform. Computer simulations may complement such experiments by providing dynamic information at an atomic level. However, there is a gap between the timescales where interesting biologically relevant conformational changes occur (typically microseconds and up) and those we can simulate at atomic resolution (typically only tens of nanoseconds). The length of atomistic simulations is limited by the need to take small timesteps (1 or 2 fs), which is determined by high frequency motions such as chemical bond stretching. One natural way to bridge this timescale gap is to use coarse grained models where the smallest unit of the system represents a group of atoms[6, 7]. In these models, much longer timesteps are allowed since the high frequency motions are not explicitly simulated. Coarse grained simulations work well for a variety of problems[8-12]; however, these models sacrifice accuracy for speed, making them less than ideal for investigating the detailed mechanisms of conformational changes.

An alternative approach to overcome the timescale gap is to build discrete-time Markov State Models (MSMs) [13-17]. These models may be built from many short (nanosecond timescale) simulations and then propagated to give long timescale dynamics, such as processes occurring on microsecond timescales or even longer. MSMs partition phase space into a number of distinct states, called metastable states, such that intra-state transitions are fast but inter-state transitions are slow. Such separation of timescales ensures that the model is Markovian, in that the probability of being in a given state at time $t+\Delta t$ depends only on the state at time t. In an MSM, the time evolution of a vector representing the population of each state may be calculated by repeatedly left-multiplying by the transition probability matrix.

$$P(n\Delta t) = [T(\Delta t)]^n P(0) \qquad (1)$$

where $P(n\Delta t)$ is a vector of state populations at time $n\Delta t$ and T is the column-stochastic transition probability matrix. Any model is Markovian for a sufficiently long lag time ($\tau = \Delta t$), because the system is able to relax to an equilibrium distribution from any arbitrary starting distribution after one lag time. The key point is to build a model with a lag time that is shorter than the timescale of the process of interest with a reasonable number of states. This requires a very good state decomposition, which is difficult. A few different approaches have been developed to address this issue[13-18]. There also exist other methods to bridge the timescale gap such as milestoning [19]. However, most of these methods require the reaction coordinate is known a priori, while this information is often difficult to obtain.

MSMs are also multi-resolution in nature[13, 14]. In order to achieve a Markovian model at a certain lag time, the states must be defined such that large internal free energy barriers are avoided and conformations within the same metastable state can interconvert within one lag time. Thus, the number of states needed in an MSM depends on the desired lag time. The smaller the lag time is, the more states the MSM needs to ensure that dynamics within each state are memory-less after one lag time. A short lag time would result in a high resolution MSM having many metastable states, capturing numerous free energy minima separated by small barriers. A longer lag time results in a low resolution MSM with only a few states, each of which contains multiple local free energy minima. We introduce a new algorithm, Super-density-level Hierarchical Clustering (SHC), to construct MSMs at different resolutions for conformational dynamics. To our knowledge, SHC is the first algorithm focusing on generating MSMs at multiple resolutions.

The key insight of the SHC algorithm is to cluster conformations hierarchically using super density level sets in a bottom-up fashion starting with the densest regions of phase space, which correspond to the bottoms of free energy minima. This algorithm can generate multi-resolution models by tuning the super density level sets, and each level of resolution constitutes a discrete-state MSM with a particular partitioning of phase space. At low

resolution, it generates a coarse state decomposition with a small number of metastable states while at high resolution it generates a finer state decomposition with more metastable states. This leaves one the flexibility to select an MSM at the best resolution to study their biological problem.

The procedure to build MSMs using SHC is as follows. (1) Partition the conformations into a large number of states, called microstates, according to their structural similarity. An approximate K-centers clustering algorithm[20] is used here as it gives states with approximately uniform size, resulting in a correlation between the population of each state and its density. (2) Split the microstates into *n* density levels ordered from high to low density ($\boldsymbol{L}$= $\{L_1, \ldots L_n\}$) such that each level contains approximately the same number of conformations. Then construct super density level sets $\boldsymbol{S_i}$, where $S_i = L_1 \bigcup L_2 \ldots \bigcup L_{i-1} \bigcup L_i$. Thus each super density level contains all previous levels $S_1 \subseteq S_2 \ldots \subseteq S_i$. (3) Within each super density level (S_i), perform spectral clustering to group kinetically related microstates. Metastable regions are better separated at high density super levels, since most of the fuzzy microstates in the transition region are excluded at these levels. Now, build a graph representing the connectivity of the states across super density levels. Then generate gradient flows along the edges of the graph from low to high density levels. Each attraction node (or attractive basin) where the gradient flow ends is assigned to a new metastable state. (4) Assign every microstate not belonging to an attraction node to the metastable state it has the largest transition probability to. Thus we have a complete state decomposition for an MSM. Furthermore, this procedure may be repeated with different super density level sets to construct MSMs at different resolutions. The larger the number of super density levels, the finer the resolution and the larger the number of metastable states in the final MSM.

In order to test SHC, we apply it to a small RNA hairpin with microsecond time scale dynamics: an eight nucleotide RNA GCAA tetraloop with the sequence 5'-GCGGCAGC-3'. It has 4 bases in the loop and two stem base pairs as shown in Figure 1. RNA hairpins are a ubiquitous secondary structure motif often involved in tertiary contacts[21]. Much experimental work has been done on these systems as a step towards understanding larger RNA molecules but knowledge of their folding is still incomplete[22-28]. Despite their small size, even eight nucleotide hairpins fold on a microsecond timescale[23], about two orders of magnitude longer than typical atomic simulations. However, using SHC, we are able to construct multi-resolution MSMs from many short 45 ns atomistic simulations. These models are able to predict microsecond timescale dynamics. We compare MSMs at different resolutions and also validate them by confirming their ability to reproduce the original simulation trajectories. Furthermore, we extract the kinetics between the most populated metastable states from our MSMs. The results suggest that the folded state is a hub connected to many non-native metastable states that are mostly uncoupled from one another. No metastable intermediate states are identified, while there are a few misfolded states such as states with shifted base pairing or an unfolded loop. This indicates that folding of an eight nucleotide RNA hairpin with only two stem base pairs might be different from RNA hairpins with longer stems where stable thermodynamic intermediate states were seen in previous simulations[22].

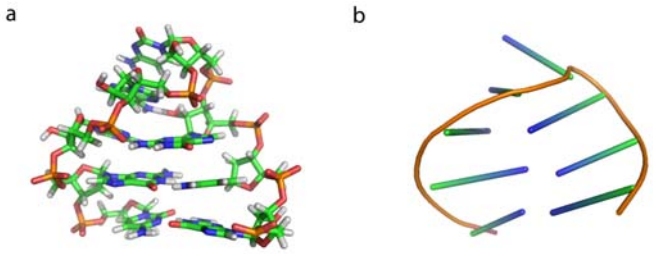


Figure 1. (A) Structure of the 8 nucleotide RNA GCAA tetraloop, generated by truncating the two terminal base pairs from the NMR structure of a 12 nucleotide tetraloop (PDB ID 1zih). (B) The cartoon representation of the same structure using sticks to represent the orientation of the bases. The same cartoon representation will be used in Figure 6 to illustrate representative structures from different metastable states.

2. Methods

Here we explain the SHC algorithm in detail using an RNA GCAA tetraloop as an example. The dataset we examine here contains 9,963 45ns explicit solvent molecular dynamics simulations with an aggregate simulation time of 448 microseconds. Conformations are saved every 0.2 ns, and the total number of conformations is about 2.3 million. These simulations are initiated from different metastable regions of phase space identified by short Simulated Tempering[29, 30] simulations following the Adaptive Seeding Method (ASM)[31]. More simulation details are available in Appendix A.

2.1. *Partitioning conformations into microstates*

Modern computer simulations can easily generate massive data sets with millions of conformations, making analysis of these data sets computationally challenging. To reduce the dimensionality of the data, we first group conformations into a large number (a few thousand or tens of thousands) of small clusters called microstates based on their structural similarity, in this case measured using the Root Mean Square Deviation (RMSD) between all heavy atoms. Each microstate must be small enough to ensure conformations in the same state can interconvert rapidly. An approximate K-centers clustering algorithm[20] was used here to generate microstates by minimizing the maximum cluster radius, where the cluster radius is defined as the maximum heavy atom RMSD distance between the cluster center and any other conformation within the cluster. The detailed implementation of the algorithm is discussed elsewhere[18, 20], and the code for the approximate K-centers clustering is available through the MSMBuilder package[18]. This algorithm has a computational complexity of $O(kN)$, where k is the number of clusters and N is the number of conformations to be clustered. Moreover, it gives states with approximately equal radii. As a result, there is a correlation between the population of each microstate and its density, allowing us to define density levels in the subsequent steps.

We have clustered ~2.3 million conformations into 10,000 microstates, and the same microstate decomposition is used to build all MSMs in this work. The cluster radius distribution has a sharp peak around 4 Å, confirming that the clusters have approximately equal radius (data not shown). Thus, the population of each microstate is a reasonable indicator of its conformational density. However, we note that even small differences in the radius of microstates may imply relatively large variations in their volumes due to the high dimensionality of conformation space. We empirically find that assuming all clusters have approximately equal volumes is useful. In the future, we can improve the density estimation step by working on low dimensional sub-manifolds where density estimation is consistent and accurate. These low dimensional sub-manifolds can be constructed with nonlinear dimensionality reduction techniques[32].

2.2. *Super density level set formation*

In this step, we first split the microstates into n density levels $\boldsymbol{L} = \{L_1, \ldots L_n\}$. As discussed above, the density of microstates $d_1 \ldots d_k$ can be estimated from their populations by dividing number of conformations within each microstate by the total number of conformaitons. We order microstates according to the value of d_i and classify the microstates into n consecutive levels. Each level contains about the same number of conformations. Density levels are ordered from high to low density, and labeled *1* to n. For example, from our RNA dataset, we have generated a density level set with three levels $\boldsymbol{L} = \{L_1, L_2, L_3\}$. L_1, L_2, and L_3 contain 146, 615, and 1810 microstates respectively, and approximately an equal number of conformations (each level contains about 25% of the total conformations, the remaining conformations are ignored until the final step of the algorithm). Thus, level L_1 has the least number of microstates and contains only the highest density regions. From the density level set, we can easily construct the super density level set $\boldsymbol{S} = \{S_1, \ldots, Sn\}$ by defining $S_i = L_1 \bigcup L_2 \ldots \bigcup L_{i-1} \bigcup L_i$. Each super density level contains all previous levels $S_1 \subseteq S_2 \ldots \subseteq S_i$. In our example, three super density levels S_1, S_2, and S_3 are created, containing 25%, 50% and 75% of the total conformations respectively. Recently, a topological data analysis approach[33, 34] based on similar ideas regarding clustering in density level sets has been successfully

applied to perform geometric clustering on biomolecular data. However, we found in this study that super level sets yield better results than density level sets in identifying kinetically metastable states (data not shown).

2.3. *Spectral Clustering within super density levels*

Spectral clustering [35-38] is performed on a transition probability matrix within each super density level (S_i). Since these transition probablity matrixs are generated by normalizing number of transitions between pairs of micorstates by counting directly from the original simulation trajectories, applying spectral clustering on them is able to lump kinetically related microstates into larger metastable states. Metastable regions are better separated in high density super levels, since most of the fuzzy microstates in transition regions are excluded at these levels. For example, in the RNA dataset, multiple disconnected blocks are found in the transition probability matrix for level S_1, indicating good separation of metastable regions. When we move up to levels containing more low density microstates, less and less disconnected blocks are found in the transition probability matrix, and eventually the matrix becomes completely connected. In the example with three density levels, the first level S_1 contains 35 metastable states, S_2 contains 25, and S_3 contains only 6 states. In order to identify nearly disconnected blocks in a transition matrix, we choose eigenvalues very close to 1 for spectral clustering. In particular, a constant spectral gap of $\Delta\lambda = 0.0001$ is used for this example.

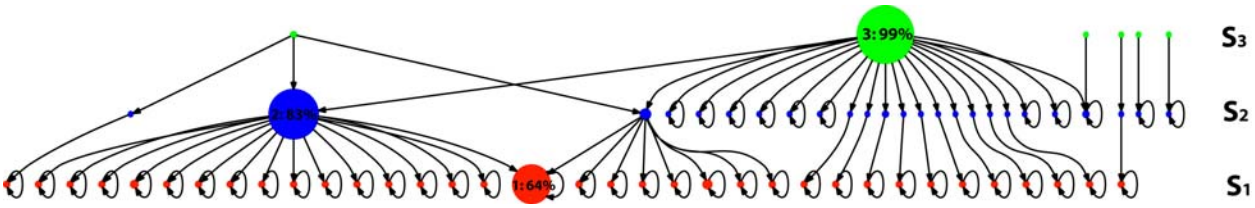


Figure 2. A graph describing the connectivity of the metastable states generated by SHC. Each node in the graph denotes a single metastable state. Each row corresponds to one super density level: states belonging to S_1 (in red), S_2 (in blue), and S_3 (in green) contain 25%, 50%, and 75% of all the conformations respectively. Two nodes are connected if they share microstates, and the arrows represent the gradient flows from low density to high density regions, i.e. from S_3 to S_1. Arrows representing self transitions are plotted at attraction nodes where the flow ends. The radius of each node is scaled linearly by its population within each super level.

Next we build a graph representing the connectivity of the metastable states across super density levels. Figure 2 is an example of such a graph with three levels. Each node in the graph represents one metastable state. As discussed above, the number of nodes in each level decreases from S_1 to S_3. In S_1, there is a large node (node 1) containing 64% of all the conformations in that level. Similar nodes can also be found in other levels such as node 2 (83%) in S_2 and node 3 (99%) in S_3. These results suggest that there is a large metastable state corresponding to the folded state, to be discussed in more detail in the *Results and Discussion* section. In the next step, gradient flows are generated along the edge of the graph from low to high density levels. Nodes that do not have any flow into denser states correspond to basins of attraction, or metastable states. For example, node 1 is an attraction node, while nodes 2 and 3 are not. As shown in Figure 2, there are 46 attraction nodes in this model (35 in S_1 and 11 in S_2). Thus the model contains 46 metastable states.

2.4. *Assigning microstates not in attraction nodes*

In the previous step, all the attraction nodes were selected as metastable states. Here, we will assign the remaining nodes to metastable states, as well as microstates that were not included in any of the density levels. This is achieved by computing the transition probabilities from each of these microstates to all possible metastable states, and assigning each microstate to the metastable state it has the largest transition probability to. If a particular microstate cannot transition to any of the metastable states in a single step we consider a progressively larger number of steps until we see transitions between this microstate and some metastable state.

Following the above steps yields a complete state decomposition for an MSM. In the example shown in Figure 2, a 46-state MSM is generated. In order to construct MSMs at different resolutions we repeat the same procedure using different numbers of super density levels.

3. Results and Discussion

3.1. *Constructing MSMs at different resolutions*

Using SHC, we have constructed four different MSMs by varying the number of super density levels (N_L) all with a lag time of 0.2 ns. The super density level set is defined as $\boldsymbol{S= \{d_0/N_L, 2d_0/N_L,\ldots, d_0\}}$, where $d_0 = 0.75$. Specifically, we used 3, 6, 9, and 15 super density levels, yielding MSMs referred to as L3 MSM, L5 MSM, L9 MSM and L15 MSM respectively. In addition, we also built a model (L1 MSM) with L = 1 as a control. Some properties of these models are listed in Table 1.

Table 1. Number of states (N), metastability (Q), and average self transition Probability ($<T_{ii}>=Q/N$) for five MSMs generated by SHC using super density level sets containing L levels.

L	1	3	6	9	15
N	6	46	57	63	68
Q	5.95	44.3	54.2	59.3	63.4
$<T_{ii}>$	99.1%	96.3%	95.1%	94.1%	93.2%

The first property in the table is the number of macrostates in each MSM. This number increases with L, and L15 MSM contains more than ten times more states than L1 MSM. With many more states, L15 MSM is a higher resolution model than L1 MSM. Thus SHC is able to generate multi-resolution MSMs by changing the number of super density levels N_L. Metastability is another important property for an MSM. A good MSM should contain a state decomposition which maximizes the separation of timescales. The self-transition probability, indicating the stability of each macrostate, is a simple and straightforward way to check if there is a good separation of timescales. The metastability (Q) listed in Table 1 is defined as the sum of the self-transition probabilities (T_{ii}) of each macrostate. Table 1 also shows the average self transition probability: $<T_{ii}> = Q/N$, where N is the number of metastable states. $<T_{ii}>$ decreases with L, indicating higher resolution models have smaller average self transition probabilities. This is consistent with the fact that higher resolution models will capture smaller free energy minima, which are separated by smaller free energy barriers and therefore less metastable.

Another interesting property, which is not listed in the table, is the population of each macrostate. For the control model L1 MSM, the populations of the six states ordered from high to low are: 98.0%, 1.6%, 0.2%, 0.05%, 0.05%, and 0.05%. Only two states have populations greater than 1%, and the rest have negligible populations. A closer look at the data shows that these four states each contain only a single microstate, and they are almost disconnected from the rest of phase space. Thus these four states might not be significant metastable regions, but just noise due to insufficient sampling. This is one issue with spectral clustering algorithms such as PCCA[37] and PCCA+[38], which tend to first separate the most disconnected blocks from the transition probability matrix. This makes it difficult to choose a proper number of metastable states in order to identify all the significant metastable regions. SHC is able to overcome this issue by clustering from the highest density super level, which guarantees that the most populated metastable regions are identified first. L3 MSM, L5 MSM, L9 MSM, and L15 MSM contain 8, 15, 12, and 10 states with populations larger than 1% respectively.

3.2. *Validating MSMs*

In this section, we will validate the MSMs discussed above in two ways: implied timescales and Chapman-Kolmogorov equation.

Implied timescales. Examining the behaviors of the implied timescales is one way to check if the model is Markovian as first suggested by Swope. *et. al.*[16]. Implied timescales (τ_k) can be computed from the eigenvalues of the transition matrix T as shown below:

$$\tau_k = -\frac{\tau}{\ln \mu_k(\tau)} \qquad (2)$$

where μ_k is an eigenvalue of the transition matrix with the lag time τ. Each implied timescale describes an aggregate transition between subsets of macrostates. If the model is Markovian and Equation (1) holds, the exponentiation of T should be identical to an MSM constructed with a longer lag time, and the implied timescales will be independent of the lag time. This requires that lag times are sufficiently long. The shortest lag time for this condition to hold is defined as the Markovian time, which is correlated with the longest internal equilibrium time of any state. Figure 3 displays implied timescales plots as a function of the lag time for L3 MSM. As shown in Figure 3 (a), the implied timescales level off around a lag time of 20ns. This implies that the model is Markovian with long enough lag times. However, big fluctuations are observed for the three slowest timescales. A further investigation shows that these slow timescales are due to low-population states which are nearly disconnected from the other states. If we exclude three states (with populations 0.1%, 0.09%, and 0.04%) containing very few non-self transition counts from our analysis, these slowest timescales disappear (see Figure 3 (b)). The implied timescale plots for other resolution MSMs also level off as shown in Figure 4. These results suggest that MSMs generated from SHC are Markovian with sufficiently long lag times. Higher resolution MSMs with a finer discretization of phase space should have shorter Markovian times, since the intra-state equilibrium times are shorter. Looking at Figure 4, the implied timescales of L15 MSM seem to level off slightly faster than those of L6 MSM. However, it is hard to tell by eye whether there is any large difference in the Markovian times for these models. Thus, the implied timescales check has some drawbacks. It is difficult to determine by eye if and where the implied timescales level off. In addition, small uncertainties in the eigenvalues can induce large uncertainties in the implied time scales[14].

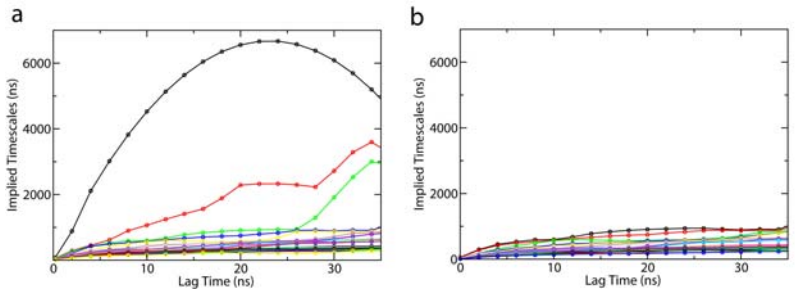


Figure 3. Top twenty implied timescales as a function of the lag time for the L3 MSM (L3 denotes the super density level set containing 3 levels) The plots are generated by using (a). the transition probability matrix with all 46 states. (b) the transition probability matrix with only 43 states with three nearly uncoupled states excluded (These three states have very few transition counts to other states).

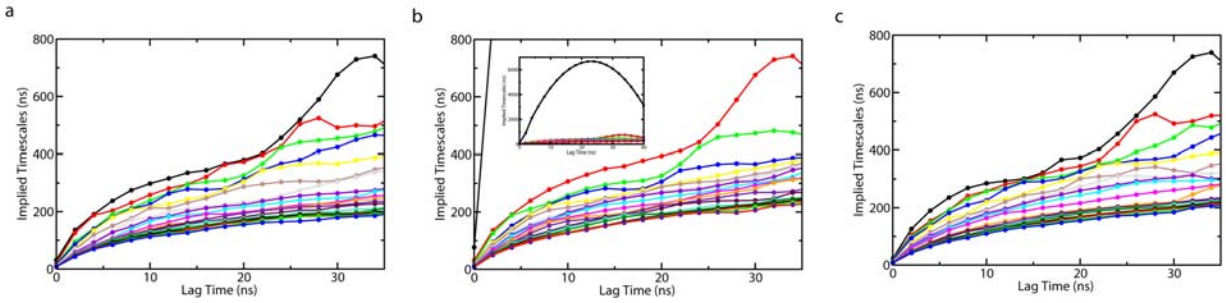


Figure 4. Top twenty implied timescales as a function of the lag time for (a) L6 MSM, (b) L9 MSM, and (c) L15 MSM. L6, L9, and L15 indicate that 6, 9 and 15 super density levels are used to generate these MSMs respectively. The insert in (b) is the same as the main figure except that the y axis goes up to 7 microseconds in order to show one very long implied timescale.

Chapman-Kolmogorov Check. An alternative way to validate MSMs is to directly check if Equation (1), a form of the Chapman-Kolmogorov equation, holds[14]. Figure 5 shows the time evolution of the populations of the top eight most populated states in L3 MSM. Populations extracted from the raw data are compared with those generated by the MSM starting from the same initial populations (see Equation (1)). As shown in Figure 5, these populations agree well within statistical error. Similar agreement was found for the other MSMs as well (data not shown). These results suggest that MSMs generated by SHC are consistent with the original dataset from which they were constructed. The final obervation is that population distriutions are almost flat, which may suggest that the starting conformations of the simulations generated from the Adaptive Seeding Method[31] are already close to the equilibrium distribution (See Appendix 1 for details).

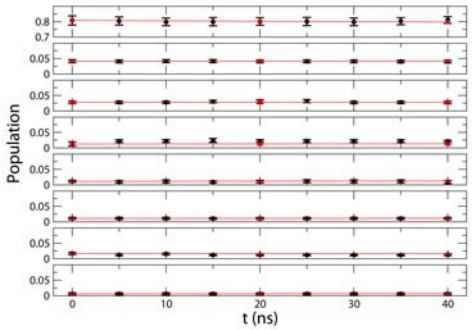


Figure 5. Comparison between the time evolution of the populations of the eight most populated states (with populations larger than 1%) in the L3 MSM (red) and the raw data (black). The error bars in the black curves are the standard deviations computed from one hundred boot strapping runs each of which randomly selected 8,000 of 9,963 trajectories with replacement. A 20ns lag time is used to build the transition probability matrix based on the L3 MSM state decomposition.

3.3. *RNA hairpin folding mechanism*

Despite the small size of RNA hairpins, there is some debate over whether they fold in a two-state or multi-state manner. Thermodynamic measurements such as temperature melting[25] support the two-state model, while kinetic experiments such as temperature jump suggest a multi-state model[39]. Using the laser temperature jump technique, the Gruebele group[23] observed two unfolding relaxation phases of the eight nucleotide gcUUCGgc hairpin at low temperatures: a fast phase of 1-2 microseconds, and a slow phase of 5-10 microseconds[23, 40]. They also developed a lattice model with four metastable states that accurately reproduced the experimental data[23]. However, it is difficult to extract information at atomic resolution from this simple model.

MSMs are a useful tool for extracting kinetics from atomistic simulations. From L3 MSM, we have computed the Mean First Passage Time (MFPT) between the eight most populated metastable states. The MFPT is defined as the average time taken to get from the initial state to the final state[41]. It can easily be computed from a transition probability matrix (see the Appendix B for details). The results of this calculation are displayed in Figure 6, along with representative structures from each state. State 1 is the folded state and has the largest population (77.1%), indicating the free energy surface is biased to the native state at 300K. Multiple non-native states, each directly connected to the folded state, are also identified: e.g. states 3 and 4 with coil structures, state 2 with a shifted base pairing, and state 5 with an unfolded loop. MFPTs for folding (i.e. transitions from non-native states to the folded state) are all around a few hundred nanoseconds, while MFPTs for unfolding are at least an order of magnitude longer (from a few to tens of microseconds). This confirms that the folded state is the most stable state at 300 K. All MFPTs between non-native states are at least eight microseconds, much longer than those for folding. This suggests that these states are uncoupled from each other. Therefore, no metastable on-pathway intermediate states are indentified in this system. The transition from state 1 (folded) to 8 (shifted base pairing) has the longest MFPT (45.7 microseconds) among all the unfolding transitions, indicating a large energy barrier for breaking non-native

base pairing/stacking followed by forming native ones. State 5 (unfolded loop) has the shortest MFPT (0.16 microseconds) among all the folding transitions, which suggests the kinetics of loop rearrangements are relatively rapid.

We have successfully extracted kinetic information between the most populated metastable states from our MSMs. The overall unfolding timescales fall in a range of a few to tens of microseconds, in qualitative agreement with experimental observations. However, direct comparisons between our simulations and laser T-jump experiments are not possible at present because our simulations are at a single temperature and are therefore unable to capture effects due to the temperature jump. No stable thermodynamic intermediate states were found for folding of this 8 nucleotide RNA hairpin, in contrast to a previous study of a 12 nucleotide hairpin[22]. These results suggest that increasing the number of stem base pairs complicates the folding mechanisms of RNA hairpins.

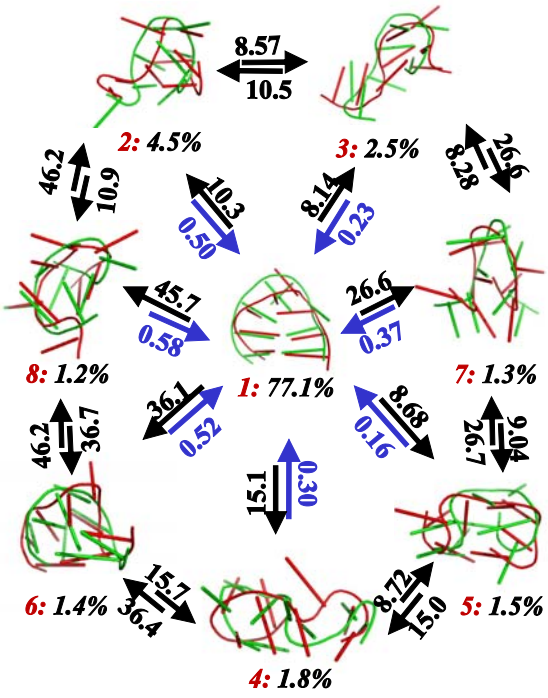


Figure 6: Mean First Passage Times (MFPTs) between the eight most populated states in the L3 MSM with a lag time of 20ns (L3 MSM is generated from a super level set with three levels, see Table 1 for details). All the MFPTs are in units of microseconds. States are labeled in red from 1 to 8 according to their populations in descending order. The populations of each state are shown in black. Two representative conformations are shown from each state using Pymol[42] with a cartoon representation. These conformations were extracted by selecting the centers of the top populated microstates in each macrostate.

4. Conclusions and Future Plans

Markov State Models (MSMs) are a useful tool for bridging the gap between experimental and computational timescales. MSMs are inherently multi-resolution, however, algorithms focused on constructing MSMs at different resolutions are lacking. Here we have introduced a new algorithm, called Super-level-set Hierarchical Clustering (SHC), which is capable of constructing MSMs of conformational dynamics at multiple resolutions. The key insight of this algorithm is to perform spectral clustering hierarchically using super level sets starting from the highest density level, which guarantees that highly populated metastable regions are identified before less populated

ones. This is an improvement over direct application of spectral clustering to the full data set, which tends to identify sparse states that are very weakly coupled to the rest of phase space due to insufficient sampling before identifying real metastable states in denser regions of phase space. We applied SHC to an 8 nucleotide GCAA RNA tetraloop, and built four MSMs at different resolutions. Each of these models was validated by both the implied timescales and Chapman-Kolmogorov checks. The overall unfolding timescales predicted from our MSMs are between a few and tens of microseconds, which are qualitatively consistent with those observed by laser temperature jump experiments. Our results suggest that there are no metastable intermediate states. Instead, the folded state is directly connected to multiple unfolded and misfolded states, which all fold faster than they interconvert with one another.

In SHC, we use the populations of microstates from K-centers clustering to approximate their conformation density. However, estimating densities in high dimensional spaces is quite challenging. In particular, our approximate K-centers algorithm only generates clusters with approximately equal radii and small variances in the cluster radius may induce large volume differences. In the future, we plan to improve our density estimates by computing kernel density functions around microstate centers or the average of the kernel density for a few randomly selected conformations within the state. Alternatively, we may employ nonlinear dimensionality reduction techniques[32] to discover lower dimensional spaces where the density may be estimated more easily. We have demonstrated that SHC is able to generate a large number of MSMs at different resolutions. However, we haven't discussed how to determine which one is the best model. A Bayesian approach to compare different MSMs by Bacallado *et al.*[43] may be used for model selection in the future. Finally, while we have focused on identifying metastable states in this work, SHC may also be used to identify intermediate and transition states by studying non-attractive nodes in lower density super density levels. In addition to being biologically relevant themselves, identification of these states could allow us to perform adaptive sampling by starting more simulations from transition states in order to rapidly sample transition events between metastable states.

Acknowledgments

XH is supported by NIH Roadmap U54 GM072970 and a startup fund from Hong Kong University of Science & Technology. XH would also like to acknowledge the support of Prof. Michael Levitt. YY, JS, LG, and GC are supported by DARPA HR0011-05-1-0007. YY and GC are also supported by NSF DMS-0354543, and LG by NSF FRG-0354543 and NIH GM-072970. GRB is supported by the NSF Graduate Research Fellowship Program. This work is also funded by NIH R01-GM062868 and NIH P01 GM066275. Computer resources were provided by NSF award CNS-0619926 and Folding@Home volunteers.

Appendix A: Simulation Details

Our simulations were generated using the Adaptive Seeding Method (ASM)[31]. First, two sets of 1120 27ns Simulated Tempering (ST) simulations[29, 30] were run: one started from a folded state and the other from a random coil. An independent MSM with 10 states was then built using MSMBuilder[18] for each dataset in order to identify the dominant metastable states. Next, one hundred random conformations were selected from each metastable state and used as starting points for new constant temperature simulations (2,000 points in total). Five 45ns constant temperature 300K MD simulations were launched from each point. This resulted in a dataset with 9,963 trajectories (some simulations were not completed). All the simulations were performed using Stanford's *Bio-X*^{2} cluster and Folding@Home[44]. We used nucleic acid parameters from the AMBER99 force field[45, 46]. The RNA molecule was solvated in a water box with 2,543 TIP3P[47] waters and 7 Na^+ ions. The simulation system was minimized using a steepest descent algorithm, followed by a 100ps MD simulation applying a position restraint potential to the RNA heavy atoms. All NVT simulations were coupled to a Nose-Hoover thermostat with a coupling constant of $0.02ps^{-1}$[48]. A cutoff of 10 Å was used for both VdW and short range electrostatic interactions. Long-range electrostatic interactions were treated with the Particle-Mesh Ewald (PME) method[49]. Nonbonded pair-lists were updated every 10 steps with an integration step size of 2 fs in all simulations. All bonds were constrained using the LINCS algorithm[50].

Appendix B: Mean First Passage Time (MFPT)

The mean first passage time (MFPT) from initial state *i* to final state *f* in an MSM is the average time taken to get from state *i* to state *f*[41]. The MFPT (X_{if}) given that a transition from state i to *j* was made first is the time it took to get from state *i* to *j* plus the MFPT from state *j* to *f*. Thus the MFPT (X_{if}) can be defined as (cite),

$$X_{if} = \sum_j P_{ij}(t_{ij} + X_{jf}) \qquad \text{(A.1)}$$

where t_{ij} is the lag time of the transition matrix T. The boundary condition for this calculation is:

$$X_{ff} = 0 \qquad \text{(A.2)}$$

The set of linear equations in Equation (A.1) and (A.2) can be solved to obtain the MFPT X_{if}.

References

1.Dobson, C.M., *Protein folding and misfolding.* Nature, 2003. **426**(6968): p. 884-90.
2.Brion, P. and E. Westhof, *Hierarchy and dynamics of RNA folding.* Annual review of biophysics and biomolecular structure, 1997. **26**: p. 113-37.
3.Kornberg, R.D., *The molecular basis of eukaryotic transcription.* Proc Natl Acad Sci U S A, 2007. **104**(32): p. 12955-61.
4.Song, J.J. and L. Joshua-Tor, *Argonaute and RNA--getting into the groove.* Curr Opin Struct Biol, 2006. **16**(1): p. 5-11.
5.Marshall, R.A., et al., *Translation at the single-molecule level.* Annu Rev Biochem, 2008. **77**: p. 177-203.
6.Levitt, M. and A. Warshel, *Computer simulation of protein folding.* Nature, 1975. **253**(5494): p. 694-8.
7.Tozzini, V., *Coarse-grained models for proteins.* Curr Opin Struct Biol, 2005. **15**(2): p. 144-50.
8.Shelley, J.C., et al., *A Coarse Grain Model for Phospholipid Simulations.* The Journal of Physical Chemistry B, 2001. **105**(19): p. 4464-4470.
9.Marrink, S.J., A.H. de Vries, and A.E. Mark, *Coarse Grained Model for Semiquantitative Lipid Simulations.* The Journal of Physical Chemistry B, 2003. **108**(2): p. 750-760.
10.Friedel, M. and J.-E. Shea, *Self-assembly of peptides into a beta-barrel motif.* The Journal of Chemical Physics, 2004. **120**(12): p. 5809-5823.
11.Brown, S., N.J. Fawzi, and T. Head-Gordon, *Coarse-grained sequences for protein folding and design.* Proc Natl Acad Sci U S A, 2003. **100**(19): p. 10712-7.
12.Buchete, N.V., J.E. Straub, and D. Thirumalai, *Orientational potentials extracted from protein structures improve native fold recognition.* Protein Sci, 2004. **13**(4): p. 862-74.
13.Noe, F. and S. Fischer, *Transition networks for modeling the kinetics of conformational change in macromolecules.* Curr Opin Struct Biol, 2008. **18**(2): p. 154-62.
14.Chodera, J.D., et al., *Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics.* J Chem Phys, 2007. **126**(15): p. 155101.
15.Buchete, N.V. and G. Hummer, *Coarse master equations for peptide folding dynamics.* J Phys Chem B, 2008. **112**(19): p. 6057-69.
16.Swope, W.C., J.W. Pitera, and F. Suits, *Describing Protein Folding Kinetics by Molecular Dynamics Simulations. 1. Theory.* J. Phys. Chem. B 2004. **108**: p. 6571-6581.
17.Sriraman, S., I.G. Kevrekidis, and G. Hummer, *Coarse master equation from Bayesian analysis of replica molecular dynamics simulations.* J Phys Chem B, 2005. **109**(14): p. 6479-84.
18.Bowman, G.R., X. Huang, and V.S. Pande, *Using generalized ensemble simulations and Markov state models to identify conformational states.* Methods, 2009.
19.Faradjian, A.K. and R. Elber, *Computing time scales from reaction coordinates by milestoning.* J Chem Phys, 2004. **120**(23): p. 10880-9.
20.Sun, J., et al., *A Fast Geometric Clustering Method on Conformation Space of Biomolecules.* In prepration, 2009.
21.Uhlenbeck, O.C., *Tetraloops and RNA folding.* Nature, 1990. **346**(6285): p. 613-4.
22.Bowman, G.R., et al., *Structural insight into RNA hairpin folding intermediates.* J Am Chem Soc, 2008. **130**(30): p. 9676-8.
23.Ma, H., et al., *Exploring the energy landscape of a small RNA hairpin.* J Am Chem Soc, 2006. **128**(5): p. 1523-30.
24.Ma, H., et al., *DNA folding and melting observed in real time redefine the energy landscape.* Proc Natl Acad Sci USA, 2007. **104**(3): p. 712-6.
25.Ansari, A., S.V. Kuznetsov, and Y. Shen, *Configurational diffusion down a folding funnel describes the dynamics of DNA hairpins.* Proc Natl Acad Sci USA, 2001. **98**(14): p. 7771-6.

26.Sorin, E.J., Y.M. Rhee, and V.S. Pande, *Does water play a structural role in the folding of small nucleic acids?* Biophys J, 2005. **88**(4): p. 2516-24.
27.Stancik, A.L. and E.B. Brauns, *Rearrangement of Partially Ordered Stacked Conformations Contributes to the Rugged Energy Landscape of a Small RNA Hairpin.* Biochemistry, 2008. **47**(41): p. 10834-10840.
28.Garcia, A.E. and D. Paschek, *Simulation of the pressure and temperature folding/unfolding equilibrium of a small RNA hairpin*, in *J Am Chem Soc*. 2008. p. 815-+.
29.Marinari, E. and G. Parisi, *Simulated Tempering: a New Monte Carlo Scheme.* Europhysics Letters, 1992. **19**: p. 451-458.
30.Huang, X., G.R. Bowman, and V.S. Pande, *Convergence of folding free energy landscapes via application of enhanced sampling methods in a distributed computing environment.* J. Chem. Phys, 2008. **128**(20): p. 205106.
31.Huang, X., et al., *Rapid Equilibrium Sampling Initiated from Non-equilibrium Data.* Proc Natl Acad Sci U S A, 2009. **In Press**.
32.Das, P., et al., *Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction.* Proc Natl Acad Sci U S A, 2006. **103**(26): p. 9885-90.
33.Yao, Y., et al., *Topological methods for exploring low-density states in biomolecular folding pathways.* J Chem Phys, 2009. **130**(14): p. 144115.
34.Singh, G., F. Memoli, and G. Carlsson, *Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition*, in *Eurographics Symposium on Point-Based Graphics*. 2007.
35.Schütte, C. and W. Huisinga, *Biomolecular Conformations can be Identified as Metastable Sets of Molecular Dynamics.* Handbook of Numerical Analysis X, 2003: p. 699-744.
36.Schütte, C. and W. Huisinga, *Biomolecular Conformations as Metastable Sets of Markov Chains.* Proceedings of the 38th Annual Allerton Conference on Communication, Control, and Computing, 2000: p. 1106-1115.
37.Deuflhard, P., et al., *Identification of almost invariant aggregates in reversible nearly uncoupled Markov chains.* Lin. Alg. Appl., 2000. **315**: p. 39–59.
38.Deuflhard, P. and M. Weber, *Robust Perron cluster analysis in conformation dynamics.* . Linear Algebra and Its Applications, 2005. **398**: p. 161-184.
39.Jung, J. and A. Van Orden, *A three-state mechanism for DNA hairpin folding characterized by multiparameter fluorescence fluctuation spectroscopy.* J Am Chem Soc, 2006. **128**(4): p. 1240-9.
40.Sarkar, K., et al., *Folding of an RNA tetraloop on a rugged energy landscape using a stacking-sensitive probe.* Submitted, 2009.
41.Singhal, N., C.D. Snow, and V.S. Pande, *Using path sampling to build better Markovian state models: predicting the folding rate and mechanism of a tryptophan zipper beta hairpin.* J Chem Phys, 2004. **121**(1): p. 415-25.
42.DeLano, W.L., *The PyMOL Molecular Graphics System.* 2002. **DeLano Scientific, Palo Alto, CA, USA**.
43.Bacallado, S., J.D. Chodera, and V.S. Pande, *Bayesian comparison of Markov models of molecular dynamics with detailed balance constraint.* Submitted, 2009.
44.Shirts, M. and V.S. Pande, *COMPUTING: Screen Savers of the World Unite!* Science, 2000. **290**(5498): p. 1903-1904.
45.DUAN, Y., et al., *A Point-Charge Force Field for Molecular Mechanics Simulations of Proteins Based on Condensed-Phase Quantum Mechanical Calculations.* J. Comp. Chem. , 2003. **24**: p. 1999-2012.
46.Wang, J., P. Cieplak, and P.A. Kollman, *How well does a restrained electrostatic potential (RESP) model perform in calculating conformational energies of organic and biological molecules?* 2000. p. 1049-1074.
47.Jorgensen, W.L.C., J.; Madura, J. D.; Impey, R. W.; Klein, M. L., J. Chem. Phys, 1983. **79**(926-935).
48.Hoover, W., Phys. Rev. A, 1985. **31**: p. 1695-1697.
49.Darden, T., D. York, and L. Pedersen., *A smooth particle mesh Ewald potential.* J. Chem. Phys., 1995. **103**: p. 3014-3021.
50.Hess, B., H. Bekker, H. J. C. Berendsen, and J. G. E. M. Fraaije., *LINCS: a linear constraint solver for molecular simulations.* J. Comput. Chem., 1997. **18**: p. 1463-1472.

MULTISCALE DYNAMICS OF MACROMOLECULES USING NORMAL MODE LANGEVIN

J. A. IZAGUIRRE[1], C. R. SWEET[2], and V. S. PANDE[3]

[1]*Dept. of Computer Science and Engineering,*
[2]*Center for Research Computing,*
Univ. of Notre Dame, Notre Dame, IN 46556 USA
E-mail: izaguirr@nd.edu, csweet1@nd.edu

[3]*Dept. of Chemistry, Stanford University, Stanford CA 94305 USA*
E-mail: pande@stanford.edu

Proteins and other macromolecules have coupled dynamics over multiple time scales (from femtosecond to millisecond and beyond) that make resolving molecular dynamics challenging. We present an approach based on periodically decomposing the dynamics of a macromolecule into slow and fast modes based on a scalable coarse-grained normal mode analysis. A Langevin equation is used to propagate the slowest degrees of freedom while minimizing the nearly instantaneous degrees of freedom. We present numerical results showing that time steps of up to 1000 fs can be used, with real speedups of up to 200 times over plain molecular dynamics. We present results of successfully folding the Fip35 mutant of WW domain.

Keywords: Normal mode dynamics; Langevin dynamics; Multiscale integrators

1. Introduction

Proteins are unique among polymers since they adopt 3D structures that allow them to perform functions with great specificity. Many proteins are molecular machines that serve numerous functions in the cell. For instance, protein kinases serve as signal transductors in the cell by catalyzing the addition of phosphate to specific residues in the same or different proteins. Different signals cause kinases to change from an inactive to an active state. To understand these biophysical processes, it is necessary to understand how proteins move (the mechanism), the kinetics (rates, etc.), and the stability of these conformations (thermodynamics).[1]

Despite many years of research, simulating protein dynamics remains very challenging. The most straightforward approach, molecular dynamics simulations using standard atomistic models (e.g. force fields such as CHARMM[2] or AMBER[3]), quickly runs into a significant sampling challenge for all but the most elementary of systems. Detailed atomistic simulations are currently limited to the nanosecond to microsecond regime. The fundamental challenge to overcome is the presence of multiple time scales: typical bond vibrations are on the order of femtoseconds (10^{-15} sec) while proteins fold on a time-scale of microsecond to millisecond. The identification of the slowest variables in the system (e.g. associated with the slowest time scales and transition rates) is to a large extent an unresolved problem.

We introduce a novel scheme for propagating molecular dynamics (MD) in time, using all-atom force fields, which currently allows real speedups of 200-fold over plain MD. We have an automatic procedure for discovering the slow variables of MD even as a molecule changes conformations, based on recomputing coarse-grained normal modes (CNMA). CNMA is fast, with cost comparable to force computation rather than diagonalization. We propose a scheme to propagate dynamics along only these slowest degrees of freedom, while still handling the near instantaneous dynamics of fast degrees of freedom. We present successful results for folding a WW domain mutant, and simulating dynamics of calmodulin and a tyrosine kinase (details in `http://www.normalmodes.info`).

Our slow variables are approximate low-frequency modes. Normal modes are the eigenvectors of the Hessian matrix H of the potential energy U at an equilibrium or minimum point x_0 with proper mass normalization. More formally assume a system of N atoms with $3N$ Cartesian positions and diagonal mass matrix M. Then, $M^{-\frac{1}{2}}HM^{-\frac{1}{2}}\mathcal{Q} = \mathcal{Q}\Lambda$, where Λ is the diagonal matrix of ordered eigenvalues and $\mathcal{Q}$ the matrix of column eigenvectors $q_1, \cdots, q_{3N}$. The frequency of a mode is equal to $\sqrt{\lambda}$ where λ is the eigenvalue. What we accomplish with normal mode analysis (NMA) is a partitioning in frequency. Low frequency modes

correspond to slow motions of the protein while the fastest modes are associated with fast local bond vibrations. This allows for efficient propagation of the slow dynamics. The following algorithm is used for 'partitioned propagation' for a system of N atoms.

1.1. *Initial setup.*

Starting at an initial conformation $x_0 \in \mathcal{R}^{3N}$, we define X and Y as vectors $\in \mathcal{R}^{3N}$ of displacements from x_0 in the slow and fast spaces, respectively. Then any configuration, x, can be written as $x = X + Y + x_0$, for projection matrices $\boldsymbol{P}(x)$ and its complement $\boldsymbol{P}^{\perp}(x) = (I - \boldsymbol{P})(x)$

$$X = \boldsymbol{P}(x)x, \; Y \; = \; \boldsymbol{P}^{\perp}(x)x. \tag{1}$$

The initial conformation x_0 is chosen as a local minimum so that we can expand the potential energy $U(x)$ about x_0. The essence of the method is to select $Q_0 \in 3N \times m$, as the first m column eigenvectors $(q_1, \cdots, q_m)$, ordered according to their eigenvalues, as the basis for the projection matrix s.t.

$$\boldsymbol{P}_0 = M^{-\frac{1}{2}} Q_0 Q_0^{\mathrm{T}} M^{\frac{1}{2}}. \tag{2}$$

In the linear case the time step is bounded by the asymptotic stability of the method[4] at a frequency equal to $\sqrt{\lambda_i}$, rather than the highest frequency in the system. Our results show this is a good heuristic to choose the time step.

1.2. *General step.*

At step i we propagate the system using the mapping Φ, which is based on a complete all-atom force-field, not a harmonic approximation:

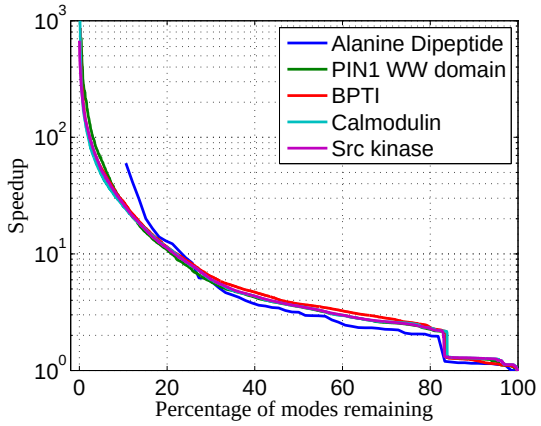
$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \Phi_{P_{i-1}} \begin{bmatrix} X_{i-1} \\ Y_{i-1} \end{bmatrix}, \tag{3}$$

such that the system is minimized w.r.t. the fast variables, i.e. $\nabla_Y U(X_i + Y_i + x_o) = 0$. *At the initial step,* $\nabla_X U(X + Y + x_o) = 0$ *as well, which is required for the initial frequency partitioning.* We then find the projection matrix, which is a function of x, from the eigenvectors of $M^{-\frac{1}{2}} H_i M^{-\frac{1}{2}}$, Q_i,

$$\boldsymbol{P}_i = M^{-\frac{1}{2}} Q_i Q_i^{\mathrm{T}} M^{\frac{1}{2}}. \tag{4}$$

This leads to a quadratic approximation for small Y and hence we can determine the frequencies for the fast variables by diagonalization to find the new projection matrix $\boldsymbol{P}_i = \boldsymbol{P}(x_o + X_{i-1})$. Note that without the initial frequency partitioning, it would not be possible to obtain a new projection matrix. *We assume that the distribution of frequencies in the model remains constant.* Figure 1 shows that this is a reasonable assumption for most folded proteins. In particular, this means that the eigenvalues associated with Y are nearly invariant with X.

Fig. 1. Maximum speedup possible for the dynamics in the slow subspace defined by keeping a certain percentage of modes for different molecules.



It is common to increase time steps in MD by constraining bonds to hydrogen, although indiscriminate constraining of bond angles significantly alters the dynamics. Similarly, some approaches to coarse-graining dynamics describe molecules as collections of rigid and flexible bodies. It is difficult to determine *a priori* the flexibility of different parts of a macromolecule, and there are numerical difficulties associated with the fastest timescales still present in the system which limit time steps to about a third of the fastest period.[5] *Equations of motion for low*

frequency modes allow substantially larger time step discretizations than fine-grained MD. This approach is more general than constraining parts of the molecule. The speedups possible are illustrated in Figure 1 for systems from 22 to 7200 atoms. These speedups are computed as the ratio of the largest time step possible by only keeping some percentage of modes over the time step needed when keeping all the modes. Notably, several different proteins show similar behavior, with potential speedups of three orders of magnitude when one resolves only a small numbers of modes. To make this discussion concrete, using 10 modes for calmodulin allows one a maximum time step of 1,000 fs, and for src kinase a time step of 2,000 fs.

1.3. *Partition Function.*

We can now define the partition function for the method as

$$Z = \int_{X+x_0} \int_Y \rho(\bar{Y}|\bar{X} + \bar{X} + x_0)\ \mathrm{d}\bar{Y}\ \mathrm{d}\bar{X}\ =\ \int_x \rho(\bar{x})\ \mathrm{d}\bar{x}, \tag{5}$$

as required.

1.4. *Partition Function Approximations.*

Exact solutions of Eqn. (5) require sampling the entire fast variable Y phase-space for a given slow variable value X (computing potential of mean force). This would require additional computation to the algorithm described above. For the initial implementation we choose the approximation, for slow variable X,

$$\int_Y \rho(\bar{Y}|X + X + x_0)\ \mathrm{d}\bar{Y} \approx \rho\left((Y_{\min}|X) + X + x_0\right), \tag{6}$$

which is readily available from the algorithm. For $\rho(a) = \exp(-\beta a)$ this is a reasonable approximation, but assumes we are at a global minimum w.r.t. X. In that case, it represents the most probable value of $\rho(Y|X)$. Even though this approximation may seem crude, our numerical results show that this is a good approximation for thermodynamics and kinetics, except perhaps when very few modes are used to form $\boldsymbol{P}$.

1.5. *Numerical discretization.*

We discretize Eqn. (3) using a numerical integrator that generates dynamics that sample Eqn. (6). Equations for the rate of change of the slow variables X with associated momenta Π need to be formulated. We wish to find a way to calculate $\mathrm{d}\Pi/\mathrm{d}t$ in terms of X and Π only. The following exact equation can be derived (which can also be found in[6]):

$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = \Pi, \qquad \frac{\mathrm{d}\Pi(t)}{\mathrm{d}t} = -\overbrace{\nabla_X A}^{\textbf{drift}} - \overbrace{\int_0^t C_r(s) \cdot \Pi(t-s)\,\mathrm{d}s}^{\textbf{friction}} + \overbrace{r(t)}^{\textbf{noise}}, \tag{7}$$

$$C_r(s) = \left\langle r(\tau + s)\, r(\tau)^T \right\rangle, \forall\tau \quad \text{(fluctuation dissipation theorem)} \tag{8}$$

These equations are in reduced units and we omitted the dependence of the memory kernel C_r on X and Π. The brackets $\langle\rangle$ define the thermodynamic average in the canonical ensemble. Eqn. (7) can be derived using the Mori-Zwanzig projection.[6] The potential $A(X)$ is the Potential of Mean Force (PMF, or Helmholtz free energy) for variable X. The integral in Eqn. (7) represents a friction. In this model the friction includes memory so this equation is often called the Generalized Langevin Equation (GLE). The last term $r(t)$ is a fluctuating force with zero mean: $\left\langle r(t)|X_0, \Pi_0 \right\rangle = 0$. This is a conditional average over Cartesian coordinates x and momenta p_x keeping $X = X_0$ and $\Pi = \Pi_0$ fixed. This equation can be rigorously derived from statistical mechanics and is therefore an attractive starting point to build coarse grained models.

We model the protein using implicit solvent models (ISM), which have been shown to be sufficiently accurate for a number of applications, including protein folding studies. They are attractive because they

greatly reduce the cost of simulating a protein. Using the approximation of Eqn. (6), Eqn. (7) can be simplified into a Langevin equation:

$$\mathrm{d}X = V\mathrm{d}t,\ M\mathrm{d}V = f\mathrm{d}t - \gamma MV\mathrm{d}t + (2k_{\mathrm{B}}T\gamma)^{1/2}\, M^{1/2}\mathrm{d}W(t), \tag{9}$$

where $f = -\boldsymbol{P}^{\mathrm{f}}\nabla U(Y_{\min}|X + X + x_0)$ is the instantaneous projection of the force unto the slow subspace, $\boldsymbol{P}^{\mathrm{f}} = M\boldsymbol{P}M^{-1}$, t is time, $W(t)$ is a collection of Wiener processes, k_{B} is the Boltzmann constant, T is the system temperature, V are the velocities and γ is a scalar friction coefficient, for instance $\gamma = 91\,\mathrm{ps}^{-1}$ for water-like viscosity. A Wiener process is a random function which is continuous but nowhere differentiable, and its derivative is white noise. A standard Wiener process has means and covariances $\langle W(t)\rangle = 0$, $\langle W(s)W(t)\rangle = \min\{s,t\}$. Systematic determination of the friction and noise can be done using efficient numerical algorithms proposed by Darve and collaborators.[7] The latter is important for kinetics but not for computing thermodynamic averages.

2. Methods

2.1. *Normal Mode Langevin (NML)*

We have previously published a numerical discretization of Eqn. (9) called Normal Mode Langevin (NML), which calculates the evolution of low frequency normal modes, while relaxing the remaining modes to their energy minimum or performing Brownian dynamics in the fast mode space (these two are equivalent).[8] NML was originally formulated using only an initial frequency partitioning to determine the slow and fast variables. However, the approximation to the low frequency dynamics afforded by NMA is only valid around an equilibrium configuration. After propagation of the slow modes, parasitic fast frequencies are introduced into the slow dynamics space, which invalidate the original frequency partitioning (both eigenvalue and eigenvector sets). Nonetheless, there is evidence that the physically relevant motions of proteins can be captured by a low frequency space,[9–15] which motivates our approach of keeping track of the low frequency space as the molecule changes conformation.

While the partitioning of slow and fast variables in the partition function above is dependent on the configuration x, in practical implementations, we only need to update this partitioning by rediagonalization when the frequency partitioning is no longer valid. Currently, we determine the rediagonalization frequency empirically, although we are exploring use of bounds on frequency content of the spaces to trigger diagonalization.

This paper presents several novel improvements of NML: The first is use of the algorithm for partitioned propagation of normal mode dynamics presented above, which does not depend on validity of the initial partitioning throughout a trajectory. The second is a scalable direct method, CNMA, for computing low frequency modes from the Hessian, which allows scaling to large molecules and long timescales. The third is a new numerical integrator, Langevin Leapfrog, that can more accurately take larger time steps when discretizing the equations of motion of NML than existing algorithms. Finally, we show that this formulation of NML can simulate protein folding.

A method similar in spirit to NML is LIN, developed by Schlick and collaborators.[16] LIN also performs a partitioning in frequency. However, LIN uses implicit integration for the low frequency modes and determines the evolution of the fast frequency modes using normal mode analysis. Langevin dynamics is used to dampen resonances. Clearly, the choices of numerical discretizations are very different: NML uses an explicit integrator for the low frequency modes, minimization or Brownian Dynamics to maintain the fast modes around their equilibrium values, and the scalable diagonalization to make rediagonalization affordable. A deeper difference is that in NML the Langevin equation is motivated by coarse-graining of the dynamics and the choice of implicit solvent model, rather than purely numerical reasons.

2.2. *Langevin Leapfrog*

In the original NML Eqn. (9) was discretized using the Langevin Impulse (LI) integrator.[17] LI is exact for constant force, and has shown numerical advantages over other commonly used Langevin integrators.[18] Schematically, a step of the NML propagator performs the following steps:

Update velocities: advance velocities using a long time step using the projection of forces unto slow subspace C.

Slow fluctuation: advance positions based on the projected velocities computed above.

Fast mode minimization: minimize positions on fast subspace $C^{\perp}$. The smaller the coupling between C and $C^{\perp}$, the fewer steps of minimization are needed. With very few modes in C coupling is very small.

In this paper we examine numerically the ability of Langevin integrators (including NML based on them) to correctly resolve dynamics for large time steps. We observe that even LI is not accurate and has a large discretization time step dependent over-damping. We derive a new integrator which we call Langevin Leapfrog, which can take large time steps with much greater accuracy. This is particularly true when used in the NML schemes.

Langevin Leapfrog is derived as a splitting method where velocities and positions are updated separately. Splitting methods arise when a vector field can be split into a sum of two or more parts that are each simpler to integrate than the original. Thus, we first integrate Eqn. (9) over time t for the velocities (from initial velocity $V(0)$ at time 0), assuming X to be constant during the velocity update:

$$V(\mathrm{t}) = e^{-\gamma \mathrm{t}} \left[\int_0^{\mathrm{t}} e^{\gamma \tau} \left(M^{-1} f(X) + \sqrt{2 k_{\mathrm{B}} T \gamma}\, M^{-\frac{1}{2}}\, \mathrm{d}W(\tau) \right) \mathrm{d}\tau + C_1 \right], \tag{10}$$

for constant C_1. Then,

$$V(\mathrm{t}) = \left(\frac{1 - e^{-\gamma \mathrm{t}}}{\gamma} \right) M^{-1} f(X) + \sqrt{2 k_{\mathrm{B}} T \gamma}\, M^{-\frac{1}{2}}\, e^{-\gamma \mathrm{t}} \int_0^{\mathrm{t}} e^{\gamma \tau} \mathrm{d}W(\tau)\, \mathrm{d}\tau + C_1 e^{-\gamma \mathrm{t}}. \tag{11}$$

The expression $e^{-\gamma \mathrm{t}} \int e^{\gamma \tau} \mathrm{d}W(\tau)\, \mathrm{d}\tau$ is equivalent to multiplying a random variable Z with mean zero and unit variance by the factor

$$e^{-\gamma \mathrm{t}} \sqrt{\int_0^{\mathrm{t}} \left(e^{\gamma \tau}\right)^2\, \mathrm{d}\tau} = \sqrt{\frac{1 - e^{-2\gamma \mathrm{t}}}{2\gamma}}. \tag{12}$$

If we assume $\mathrm{t} = 0$ at initial velocity V^n, initial positions X^n and $\mathrm{t} = \Delta t/2$ at $V^{n+\frac{1}{2}}$ then

$$C_1 \ = V^n, \tag{13}$$

and

$$V^{n+\frac{1}{2}} = e^{-\gamma \frac{\Delta t}{2}} V^n + \left(\frac{1 - e^{-\gamma \frac{\Delta t}{2}}}{\gamma} \right) M^{-1} f(X^n) + \sqrt{2 k_{\mathrm{B}} T \gamma}\, M^{-\frac{1}{2}} \sqrt{\frac{1 - e^{-\gamma \Delta t}}{2\gamma}}\, Z^n, \tag{14}$$

for random variable Z^n with zero mean and unit variance.

Positions can be found, assuming constant velocity during the position update, from:

$$X^{n+1} = X^n + \Delta t\ V^{n+\frac{1}{2}}, \tag{15}$$

and finally the remaining half step for the velocities

$$V^{n+1} = e^{-\gamma \frac{\Delta t}{2}} V^{n+\frac{1}{2}} + \left(\frac{1 - e^{-\gamma \frac{\Delta t}{2}}}{\gamma} \right) M^{-1} f(X^{n+1}) + \sqrt{2 k_{\mathrm{B}} T \gamma}\, M^{-\frac{1}{2}} \sqrt{\frac{1 - e^{-\gamma \Delta t}}{2\gamma}}\, Z^{n+1}, \tag{16}$$

for random variable Z^{n+1}, again with zero mean and unit variance.

2.3. *Coarse-grained Normal Mode Analysis*

The need to re-diagonalize a mass-weighted Hessian in NML, while greatly improving the accuracy of the model and making it possible to track conformational change, is very expensive, with $\mathcal{O}(N^3)$ computational time and $\mathcal{O}(N^2)$ memory. We have developed a *coarse-grained normal mode analysis that is scalable.* CNMA is a 2-level, direct method that uses a dimensionality reduction strategy that allows computation of low frequency modes in $\mathcal{O}(N^{9/5})$ time and $\mathcal{O}(N)$ memory.

2.3.1. *Dimensionality reduction strategy.*

The coarse-graining strategy to computing the frequency partitioning is based on 2 ideas. The first is to find a reduced set of normalized vectors E whose span contains the low frequency space of interest, C. The second is to find an orthogonal set of vectors V with the same span as E, which are ordered according to the diagonal elements of $V^{\mathrm{T}}HV$. The span of the first m columns of V, where m is the number of reduced collective motions, still spans C and constitute the approximate low frequency eigenvectors. To keep computational cost low, we form H, and matrix-vector products involving H, in linear cost, $\mathcal{O}(N)$. A brief description follows.

2.3.2. *Choice of reduced matrix E.*

In order to form E, we use a model of a protein that is easier to diagonalize than the full-atom forcefield model, but that nonetheless contains the same low frequency motion space. Our model is that of independent blocks of residues with arbitrary rotation, translation, and low frequency dihedrals. Clearly, such a model allows more flexibility than the full-atom protein model. We show that it can indeed contain the low frequency motion space of interest. We start from a block Hessian in which each block $\tilde{\boldsymbol{H}}_{ij}$ (composed of 1 or more residues) is zero if $i \neq j$. The remaining blocks on the diagonal are assumed to be independent of all other blocks. This block Hessian is then diagonalized, which is equivalent to performing independent diagonalization for each block. The block Hessian eigenvectors and eigenvalues, $\mathcal{Q}_i$ and D_i, are calculated as follows:

$$\tilde{\boldsymbol{H}}_{ii}\mathcal{Q}_i = \mathcal{Q}_i D_i.$$

Our hypothesis is that interactions among residues responsible for the low frequency space of interest will be included, either by projection or directly, in the first few eigenvectors of $\mathcal{Q}_i$, and need to be included in E. The source of these vectors is as follows:

1. External low frequency motions due to nonbonded interactions are projected onto the first 6 eigenvectors of $\mathcal{Q}_i$, corresponding to conserved degrees of freedom (d.o.f.) per block. In other words, external forces manifest themselves in rotations or translations of each residue-block.
2. External low frequency motions due to bonded interactions are projected onto the dihedral space, and will consist of 2 vectors of $\mathcal{Q}_i$, due to backbone dihedrals of up to 2 connecting blocks.
3. Internal low frequency motions, for instance due to side-chain dihedral motions, will also be in the dihedral space and thus will be in $\mathcal{Q}_i$.

Residue:	ARG	PRO	ASP	PHE	CYS	LEU	GLU	TYR	GLY
No. vectors:	15	9	11	13	10	12	13	13	8
Residue:	LYS	ALA	ILE	ASN	GLN	THR	VAL	SER	MET
No. vectors:	14	9	14	12	14	11	11	11	15

Table 1: Number of vectors, k, selected per residue for BPTI, showing that larger residues require greater numbers of vectors.

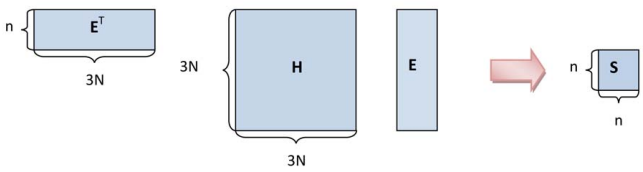
The number of vectors of $\mathcal{Q}_i$ included in E varies according to the residue composition. We refer to the average number of these vectors by block as the block d.o.f. (*bdof*). We expect that the eigenvectors

identified above will correspond to the first k ordered eigenvalues. The number k varies between blocks and is determined by selecting a cutoff frequency from the block eigenvalues. Table 1 gives values of k for BPTI, where the *bdof* $= 12$, and where each block has only 1 residue. As expected, larger residues such as ARG require a greater number of vectors to describe their low frequency motions than smaller ones like GLY.

2.3.3. *Finding orthogonal matrix V.*

Figure 2 illustrates the dimensionality reduction strategy. The dimensions of E are $3N \times n$, where $n \ll N$. The quadratic product $E^{\mathrm{T}}HE$ produces a matrix S of reduced dimensions $n \times n$. H is a Hessian that includes interactions among residue-blocks, i.e., the full Hessian or an approximation thereof. We use the Hessian coming from using cutoff in the nonbonded forces. From the diagonalization of S we can obtain Q. In particular, we (cheaply) diagonalize the symmetric matrix S to find orthonormal matrix $\tilde{Q}$ s.t.

Fig. 2. Dimensionality reduction strategy.



$$S\tilde{Q} = \tilde{Q}\Omega,$$

for diagonal matrix Ω. We can then write

$$Q^{\mathrm{T}}HQ = \Omega,$$

for $Q = E\tilde{Q}$. V is defined as the first m columns of Q, where m is typically in the range of 10 - 100. Our subspace of dynamical interest, C, is included in the span of V.

We can evaluate how well the span of E represents C using the following result: Let the i^{th} ordered diagonal of Ω be $\sigma_i = \Omega_{ii}$. It can be shown that the highest frequency mode in C, $f_{\max}$, satisfies

$$f_{\max} \leq \sqrt{|\sigma_m|}.$$

The Rayleigh quotient $\sigma_m = E_m^{\mathrm{T}}HE_m$ can be used to establish the maximum time step that can be taken in subspace C for stability. It follows that if σ_m is close to the m^{th} ordered eigenvalue of H, then V is a good representation of the low frequency space of interest. Since this result does not take account of conserved d.o.f., with zero eigenvalues, we need to be careful to include these in our target E.

2.3.4. *Efficient implementation.*

The quadratic product $E^{\mathrm{T}}HE$, which naively implemented would still be an $\mathcal{O}(N^3)$ operation, is made $\mathcal{O}(N)$ by exploiting the quasi-block structure of H and E and using cutoff for the electrostatics. This cost can still be maintained if full electrostatics are needed by using coarse-grained electrostatic representations. Figure 3 illustrates the block structure of H, which is similar to a protein contact map. Contiguous residues give a tri-diagonal block structure. Non-contiguous residues within a cutoff form off-diagonal blocks due to nonbonded forces. The block structure of E follows from its composition from eigenvectors of the block Hessians $\tilde{\boldsymbol{H}}_{\boldsymbol{ii}}$.

3. Results

3.1. *Normal Mode Langevin dynamics*

We performed folding simulations of the Fip35 mutant of WW domain using NML, discretized with Langevin Leapfrog, with periodic rediagonalization using CNMA. The force field is CHARMM 27 with the screened Coulomb potential implicit solvent model (SCPISM).[19] The temperature $T = 300\,\mathrm{K}$ and friction coefficient

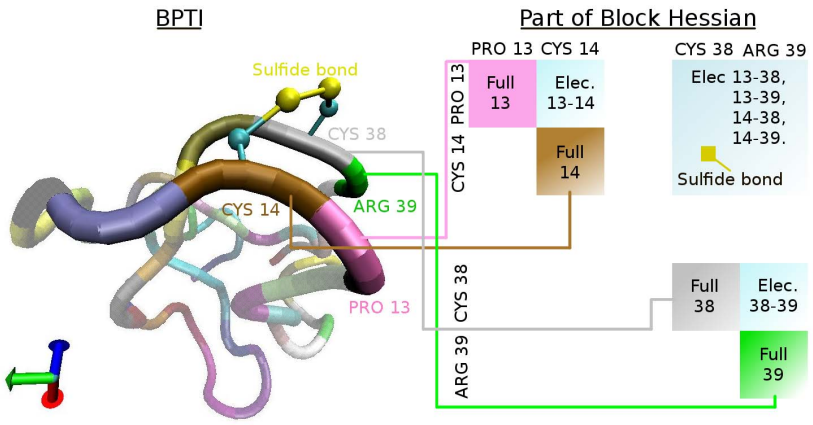
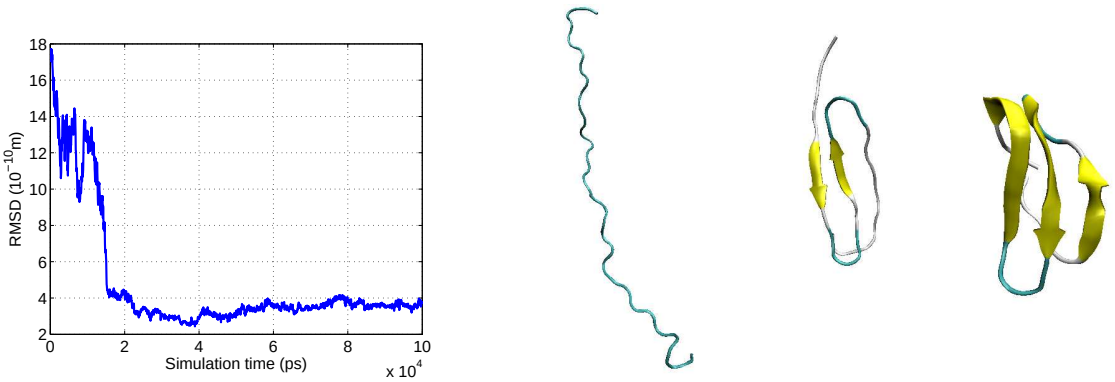


Fig. 3. Segment of a BPTI molecule and its associated Hessian entries. Here, for illustration, a block is defined by one residue. Each residue corresponds to a Hessian block containing all of the forces within the residue, denoted 'Full'. Adjacent residues have a corresponding electrostatic block denoted 'Elec.', e.g. Elec. 13-14. Physically local residues within the cutoff distance have a corresponding electrostatic block, e.g. Elec. 13-38. Bonds connecting non-adjacent residues, such as the disulfide bonds shown, correspond to small 3x3 blocks in the Hessian.



(a) RMSD (b) Snapshots

Fig. 4. (a) RMSD of C_α of β-sheets (in Å) for Fip35 WW folding NML simulation. (b) Snapshots from the same simulation, (i) at the start, (ii) after 125 ns, and (iii) after 330 ns.

$\gamma = 91\,\mathrm{ps}^{-1}$. We compared to a set of 1000 simulations using plain Langevin dynamics with same T and γ, with combined sampling of 393 μs from Ensign and Pande,[20] where 2 simulations fold. From a set of 200 NML simulations with combined sampling of 198 μs, we found 2 that folded. Folding is determined as in their work: the 3 β-sheets are fully formed and the RMSD of their residues to the native structure is less than 3 Å. These NML simulations used 10 modes, CNMA parameters of $bdof = 14$ and 2 residues per block, and time step of 100 fs. Figures 4(a) and 4(b) show the RMSD, and snapshots from an NML folding simulation, respectively. We estimated a folding rate of $\langle k \rangle = (66\,\mu s)^{-1}$ with $\sigma(k) = (114\,\mu s)^{-1}$. The rate reported by Ensign and Pande is $\langle k \rangle = (131\,\mu s)^{-1}$ with $\sigma(k) = (226\,\mu s)^{-1}$. An experimental rate has been reported[21] as $(13.3\,\mu s)^{-1}$. Our results suggest agreement of the free energy of activation within a factor of 2, a reasonable expectation for these force fields. The folding rate was computed using a Bayesian modification of a maximum likelihood estimate.[22] With n folding simulations out of N total simulations, with total necessary simulation time Θ, the estimators for the mean rate and its variance are:

$$\langle k \rangle = \frac{n+1}{\Theta}, \qquad \mathrm{var}(k) = \frac{n+1}{\Theta^2}.$$

Θ is the sum of the first-passage times for simulations that folded, plus total simulation time for simulations that did not fold. In our NML simulations, one simulation folded after 33 ns and another one after 303 ns, and all the simulations ran for 1 μs.

These results were not significantly different when running folding simulations with 15 and 20 modes. The number of residues per block used in CNMA is chosen to optimize runtime, and is a function of N as explained below. We determine the *bdof* to use for CNMA by comparing the $L2$ norm for the difference of the low frequency eigenvalues against those of a full diagonalization, or when this is too costly, against a diagonalization with large *bdof*. We use the minimum *bdof* after which the norm reaches a plateau.

It would be desirable to have more folding events for computation of the rate in order to reduce the standard deviation. The significance of our result is that NML with periodic rediagonalization can track large conformational change with very few modes.

Molecule	Atoms	Δt (fs)	Itrs	ns/day	Speedup
Fip35 WW	544	1	-	6	-
Fip35 WW	544	100	1.0	240	40
Fip35 WW	544	200	1.2	360	60
Fip35 WW	544	500	4.0	455	76
Calmodulin	2262	1	-	0.4	-
Calmodulin	2262	100	1.0	13.7	34
Calmodulin	2262	500	1.9	60.4	151
Calmodulin	2262	1000	8.2	90.9	227
Tyr kinase	7214	1	-	0.07	-
Tyr kinase	7214	100	1.2	1.6	23
Tyr kinase	7214	500	1.3	7.4	106
Tyr kinase	7214	1000	1.8	14.4	206

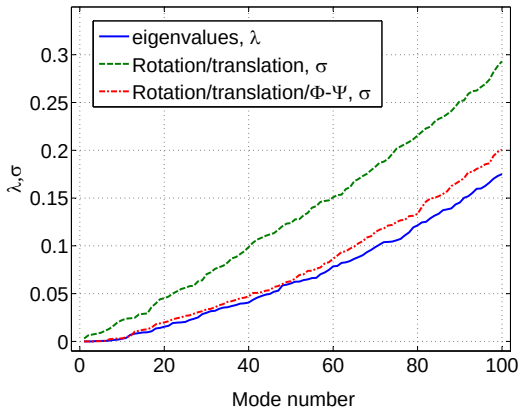
Table 2: Speedup of NMLL vs. MD. Rows with time step Δt of 1 fs correspond to MD. "Itrs" is the average number of minimization iterations. "Speedup" is the speedup of using NMLL vs. plain MD. All NMLL runs use 10 modes and rediagonalization every ps.

Timings of NMLL and MD on one core of Xeon E5420 2.5 GHz, computed by running 100 ps of equilibration and 1 ns of simulation are in Table 2. These results show that there is not a significant overhead in our NMLL approach, considering the performance gained from a significantly larger time step (in comparison with previous approaches, which could take large time steps as well, but with a large overhead that limited their applicability[16]).

3.2. *Coarse grained normal mode analysis (CNMA)*

Figure 5 shows how well these choices of coarse-graining represent the low frequency space of a protein. Results were obtained by using different values of k in constructing E vs. the true eigenvalues for BPTI. The closer the line is to the true eigenvalues, the better the coarse-graining strategy. Rotation/translation means that E is made from the first 6 eigenvectors of each residue-block matrix. Rotation/translation/$\Phi - \Psi$ means that in addition vectors are included that correspond to low frequency dihedral motions. Note that only including the first 6 eigenvectors of the block Hessians in E diverges from the true eigenvalues, whereas also including the low frequency dihedral

Fig. 5. Rayleigh quotients and true eigenvalues for different Hessian coarse graining schemes.



Molecule	Atoms	Lapack Time [s]	Lapack RAM [Gb]	CNMA Time [s]	CNMA RAM [Gb]
WW	551	14.4	0.04	0.37	0.01
BPTI	882	59.9	0.11	0.89	0.03
CaM	2262	980.6	0.74	3.89	0.12
Tyr Kinase	7214	31450.0	7.49	31.90	0.69
F1-ATPase	51181	11.2E6	377.0	1827.0	2.04

Table 3: Comparison of the 'brute force' Lapack diagonalization and the coarse grained method for different atomic models.

vectors gives very good results. *We can thus evaluate the fitness of any proposed Hessian coarse-graining procedure.*

3.2.1. *Scaling results.*

Five models were used for the comparison of the 'brute force' diagonalization and the CNMA method: Pin1 WW domain (PDB 1I6C), BPTI (PDB 4PTI), Calmodulin (PDB 1CLL), Tyrosine kinase (PDB 1QCF), and F1-ATPase (PDB 2HLD). The results can be seen in Table 3. The scaling with time for the 'brute force' Lapack diagonalization method is known to be $\mathcal{O}(N^3)$. For the coarse grained CNMA method using b blocks we have the cost of diagonalizing all b blocks as $\mathcal{O}((N/b)^3) \times b = \mathcal{O}(N^3/b^2)$ and for the small projected matrix as $\mathcal{O}(b^3)$, which has a minimum cost when $b \propto N^{3/5}$, giving an estimated cost of $\mathcal{O}(N^{9/5})$. This is borne out by the numerical evidence. For the coarse grained method the RAM resource usage is reduced from the 'brute force' scaling of $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$.

Other diagonalization techniques to obtain low frequency eigenvectors are reported in the literature, such as the ODMG energy functional, which was recently used by Dykeman and Sankey to analyze viral capsids.[23] Unlike these methods, which are iterative, and where convergence is highly system dependent, CNMA is a direct method, has been validated in a rigorous way, both through useful bounds on the eigenvalues, and through its application in NML. Our own numerical experiments with Tracemin and other iterative methods for solving our eigenvalue problem indicate that convergence of these iterative methods is very slow, since they require an approximation to the inverse of the Hessian from the start. Our dimensionality reduction strategy for CNMA is more effective.

3.3. *Langevin Leapfrog*

We applied NML to study the isomerization kinetics of blocked alanine dipeptide (ACE ALA NME) between the $C7$ equatorial and α_R conformations. Conformation A is C7 equatorial and C5 axial combined, and conformation B is α_R. Figure 6 shows the free energy as a Ramachadran plot for Alanine Dipeptide using a sigmoidal screened Coulomb potential.[24] We refer to NML with rediagonalization to update the low frequency modes as NML(m,*period*) where m is the number of slow modes propagated, and *period* the rediagonalization period in femtoseconds. LL greatly eliminates the over-damping due to the discretization time step of LI. Figure 7(a) shows the isomerization rates for alanine dipeptide using LI, NML using LI (NMLI) and NML using LL (NMLL). Note that NMLL can compute the rate with time steps of 16 fs using 12 modes (6 conserved plus 6 real modes) with rediagonalization every 100 steps, whereas LI and NMLI's rate significantly decreases with increasing time step.

Fig. 6. Free energy for alanine dipeptide

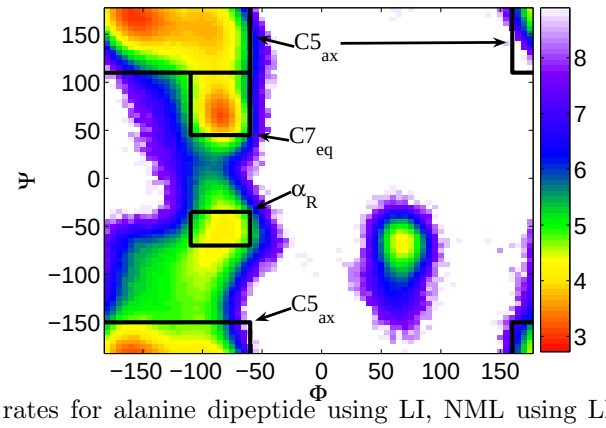
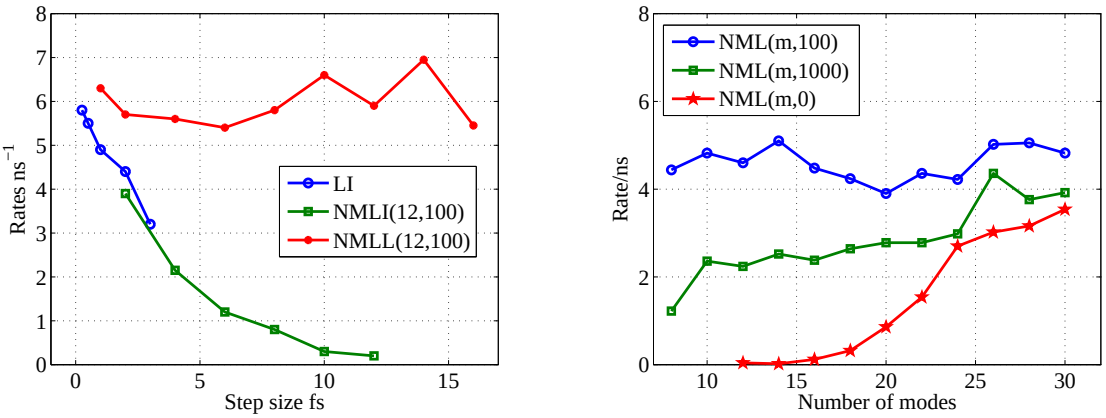


Figure 7(b) shows the isomerization rate for alanine dipeptide for varying rediagonalization periods: the rate is correctly computed for NMLL(m,100) for even 7 modes (6 conserved plus 1 real mode).



(a) Time step effect

(b) Rediagonalization effect.

Fig. 7. Isomerization rate as (a) function of time step for different Langevin integrators, and (b) different rediagionalization periods. Error bars are smaller than the symbol sizes.

4. Conclusions

We have presented a novel scheme to propagate multiscale dynamics of proteins and other macromolecules, based on computing periodically coarse-grained normal mode dynamics. We have presented results of folding a WW domain, as well as speedups on different proteins. A major advance detailed herein is a scheme to calculate NML dynamics in a very efficient way, i.e. with scaling of the form $\mathcal{O}(N^{9/5})$, which allows for the rapid calculation of long timescale dynamics. This was further demonstrated with specific examples, including the microsecond dynamics of the folding of a WW-domain. Thus, in its current implementation described herein, NML can greatly accelerate molecular dynamics calculation for a wide variety of applications, while retaining quantitative fidelity to more traditional methods.

The main approximation in NML is the assumption of Eqn. (6) regarding fast frequency motion. Numerical evidence suggests this is a reasonable assumption, but this issue needs to be more thoroughly evaluated, and if necessary, computationally efficient approximations of the PMF need to be derived. Since NML prescribes frequent rediagonalization, the PMF needs only be valid in a neighborhood of phase space around a given value of the slow variable X. Thus, this is a less formidable problem than in the general coarse-graining case.

Current work includes extending the Langevin equation to include memory, which may be relevant when using very few low frequency modes if one wants to compute kinetics; implementing a multilevel formulation of CNMA with $O(N \log N)$ complexity, and combining NML with Markov State Models to reach millisecond time scale dynamics. All these methods are included in the open source software Protomol.[25] There is an implementation reference and a tutorial on running NML, along with a discussion of how to choose the *bdof* parameter of CNMA and the number of modes in NML (`http://sourceforge.net/projects/protomol`). NML will be included in future releases of the library OpenMM.

Acknowledgments

JAI acknowledges funding from NSF (CCF-0622940, DBI-0450067) and VSP from NIH (NIH U54 GM072970, R01-GM062868) and NSF (CHE-0535616, EF-0623664). We have benefited from discussions with Prof. Eric

Darve at Stanford University, Prof. Robert D. Skeel at Purdue, and John Chodera at UC Berkeley. Students Santanu Chatterjee, Faruck Morcos, Jacob Wenger, and Antwane Mason at Notre Dame, and Dan Ensign at Stanford, performed some of the analyses presented here.

References

1. Russel, D., Lasker, K., Phillips, J., Schneidman-Duhovny, D., Veláquez-Muriel, J. A., and Sali, A. *Curr. Opinion Cell Biol.* **21**, 1–12 (2009).
2. MacKerell Jr., A. D., Wiorkiewicz-Kuczera, J., and Karplus, M. *J. Am. Chem. Soc.* **117**(48), 11946–11975 (1995).
3. Cornell, W. D., Cieplak, P., Bayly, C. I., Gould, I. R., Merz, K. M., Ferguson, D. M., Spellmeyer, D. C., Fox, T., Caldwell, J. W., and Kollman, P. A. *J. Am. Chem. Soc.* **117**(19), 5179–5197 (1995).
4. Batcho, P. F. and Schlick, T. *J. Chem. Phys.* **115**(9), 4019–4029 (2001).
5. Ma, Q., Izaguirre, J. A., and Skeel, R. D. *SIAM J. Sci. Comput.* **24**(6), 1951–1973 (2003).
6. Zwanzig, R. *Nonequilibrium Statistical Mechanics*. Oxford, (2001).
7. Darve, E., Solomon, J., and Kia, A. *Proc. Natl. Acad. Sci. USA* **27**, 10884 (2009).
8. Sweet, C. R., Petrone, P., Pande, V. S., and Izaguirre, J. A. *J. Chem. Phys.* **128**(11), 1–14 (2008).
9. Brooks, B. and Karplus, M. *Proc. Natl. Acad. Sci. USA* **82**, 4995–4999 (1985).
10. Levitt, M., Sander, C., and Stern, P. S. *J. Mol. Biol.* **181**, 423–447 (1985).
11. Bahar, I., Atilgan, A., and Erman, B. *Fold. Des.* **2**, 173–181 (1997).
12. Ma, J. *Structure* **13**, 373–380 (2005).
13. Tama, F. and Sanejouand, Y. H. *Protein Engng* **14**(1), 1–6 (2001).
14. Cui, Q., Li, G., Ma, J. P., and Karplus, M. *J. Mol. Biol.* **340**(2), 345–372 (2004).
15. Petrone, P. and Pande, V. *Biophys. J.* **90**, 1583–1593 (2006).
16. Zhang, G. and Schlick, T. *J. Comp. Chem.* **14**, 1212–1233 (1993).
17. Skeel, R. D. and Izaguirre, J. A. *Mol. Phys.* **100**(24), 3885–3891 (2002).
18. Wang, W. and Skeel, R. D. *Mol. Phys.* **101**, 2149–2156 (2003).
19. Hassan, S., Mehler, E., Zhang, D., and Weinstein, H. *PROTEINS: Struc., Func., and Genetics* **51**, 109–125 (2003).
20. Ensign, D. L. and Pande, V. S. *Biophys. J.* **96**, L53–L55 (2009).
21. Freddolino, P. L., Liu, F., Gruebele, M., and Schulten, K. *Biophys. J.* **94**(10), L75–77 (2008).
22. Zagrovic, B. and Pande, V. *J. Comp. Chem.* **24**, 1432–1436 (2003).
23. Dykeman, E. C. and Sankey, O. F. *Phys. Rev. Lett.* **100**, 028101 (2008).
24. Shen, M.-Y. and Freed, K. F. *J. Comp. Chem.* **26**(7), 691–698 (2005).
25. Matthey, T., Cickovski, T., Hampton, S. S., Ko, A., Ma, Q., Nyerges, M., Raeder, T., Slabach, T., and Izaguirre, J. A. *ACM Trans. Math. Softw.* **30**(3), 237–265 (2004).

INSIGHTS INTO THE INTRA-RING SUBUNIT ORDER OF TRIC/CCT: A STRUCTURAL AND EVOLUTIONARY ANALYSIS

NIR KALISMAN

Department of Structural biology, School of Medicine, Stanford University
Stanford, California 94305, USA

MICHAEL LEVITT

Department of Structural biology, School of Medicine, Stanford University
Stanford, California 94305, USA

TRiC is an important group II chaperonin that facilitates the folding of many eukaryotic proteins. The TRiC complex consists of two stacked rings, each comprised of eight paralogous subunits with a mutual sequence identity of 30-35%. Each subunit has unique functional roles that are manifested by corresponding sequence conservation. It is generally assumed that the subunit order within each ring is fixed, but this order is still uncertain. Here we address the problem of the intra-ring subunit order by combining two sources of information: evolutionary conservation and a structural hypothesis. Specifically, we identify residues in the TRiC subunits that are likely to be part of the intra-unit interface, based on homology modeling to the solved thermosome structure. Within this set of residues, we search for a subset that shows an evolutionary conservation pattern that is indicative of the subunit order key. This pattern shows considerable conservation across species, but large variation across the eight subunits. By this approach we were able to locate two parts of the interface where complementary interactions seem to favor certain pairing of subunits. This knowledge leads to restrictions on the 5,040 (=7!) possible subunits arrangements in the ring, and limits them to just 72. Although our findings give only partial understanding of the inter-subunit interactions that determine their order, we conclude that they are comprised of complementary charged, polar and hydrophobic interactions that occur in both the equatorial and middle domains of each subunit.

1. Introduction

Chaperonins are large protein complexes that assist in the folding of nascent and misfolded polypeptide chains. The group II chaperonins are found only in eukaryotes and archaea and share a similar overall structure [1]. They consist of two stacked rings with flexible tops that can open and close through an ATP binding and hydrolysis cycle. Upon closure, an isolated cavity is formed within each ring, where refolding of polypeptide chains can occur.

TRiC is an important member of the group II chaperonins, and is highly conserved in most eukaryote families. It has been implicated in the folding pathways of many proteins, most notably actins and tubulins [2,3,4]. Each ring in TRiC consists of eight paralogous subunits: A, B, G, D, E, H, Q and Z. The specific differentiation of the eight TRiC subunits is highly conserved. For example, the sequence identity between the α subunit of yeast and the α subunit of bovine is greater than 60%, yet the sequence identity between different subunits in bovine is only ~30%. Clearly, this remarkable conservation is likely to ensure a constant order of the subunits within the ring in addition to its other functional roles in TRiC [5,6].

Because of their allosteric nature, the lid closure and the ATP cycle are highly influenced by the order of subunits within the ring [7]. Yet, despite its functional role, the order of subunits in the ring is still debated. Liou and Willison [16] proposed an arrangement of subunits in the ring that is based on a very difficult experimental protocol that analyzes micro-complexes, i.e. very partial fragments of TRiC rings found in cell extracts. This arrangement is used to this day [7,8], but has not been validated by a different methodology. Unfortunately, structural data that could unambiguously resolve the order of the subunits is not available. TRiC is very refractory to crystallization, and cryo-EM reconstructions of TRiC are currently at insufficient resolution to differentiate the subunits [8].

In order to make progress in the elucidation of ring order, we tried a computational approach. The subunit order in TRiC is a challenging modeling problem because of the large number of possible arrangements for this complex. In addition, the sequence identity between the subunits is high as well (~30%) making the molecular key to the subunit order more subtle. To overcome these difficulties we combined two sources of information: evolutionary conservation and a structural hypothesis. The structural model is based on the crystal structures for a close orthologue of TRiC, the archaeal thermosome [9]. The thermosome is also a group II chaperonin that consists of

only two subunit types, which are not very differentiated (60% sequence identity). We analyzed the model by looking for residues that were part of the subunit interface and showed a conservation pattern that may indicate of the subunit order. Specifically, we looked for patterns that showed considerable conservation across species, but large variability across the eight subunits. We identified several areas of the interface where complementary interactions could putatively favor certain subunit pairings. From this analysis we derived a set of constraints that limit the number of possible subunits arrangements from 5,040 to 72.

2. Methods

The evolutionary profile in each position is based on sequence alignments from 13 species (number in parenthesis is sequence identity to the bovine gene): Bovine (100%), Human (97%), Zebrafish (86%), Ciona (74%), Drosophila (73%), C. Elegans (68%), Arabidopsis (66%), Yeast (63%), Neospora (61%), Candida (61%), Dictyostelium (61%), Plasmodium (58%) and Paramecium (58%). This order of decreasing sequence identity to the bovine genes is kept (in a left to right manner) throughout the presentations in the tables. The main guideline in choosing these specific organisms was to form a diverse set of family representatives that spans the eukaryotic evolutionary tree. A secondary guideline was to use only fully annotated genomes that had clear annotation of the eight different subunits. The eight sequences from each organism were aligned to each other and to the thermosome sequences using the COBALT multiple sequence alignment tool from NCBI [10].

The structural model (Fig. 1) was based on the crystallographic structure of the archaeal thermosome from T. acidiphilum (PDB 1A6D). The sequence identity between TRiC and the thermosome is ~40%, indicating a highly reliable model. Unlike TRiC, which has eight different subunits, each ring in the thermosome is composed of four repeats of two different subunits (α and β). This arrangement leads to only two different interfaces, and we focused on the α-β interface. We exclude the apical domain from the structural analysis because it is unclear whether TRiC assembles in a close or open conformation. We also exclude the loops spanning residues 53-57 and 163-166 since they are structurally variable between the two thermosome subunits.

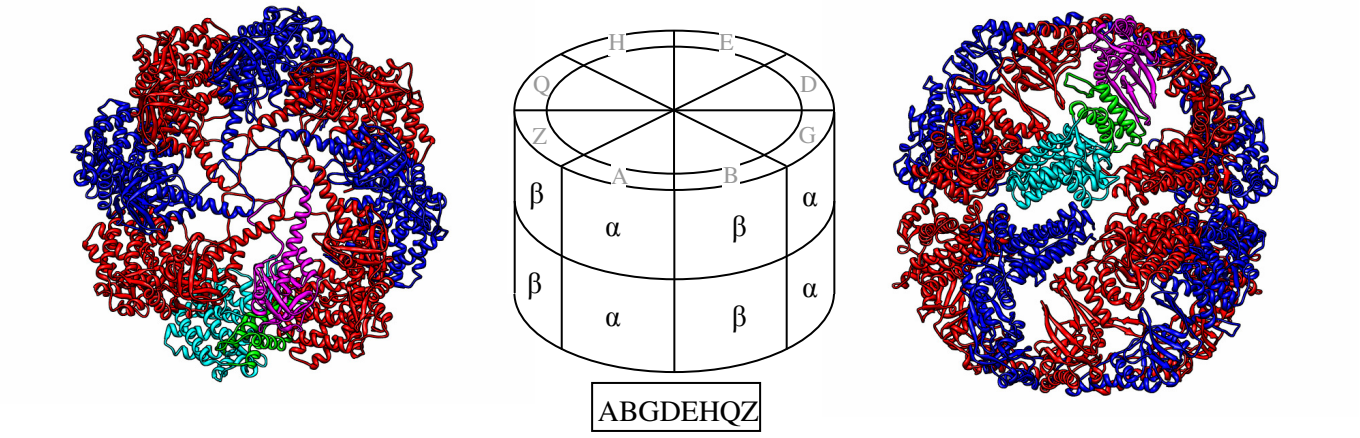


Figure 1. Our structural model is based on the known structure of the thermosome (PDB file 1A6D), which has α (blue) and β (red) subunits that alternate around each of the two rings. The structure is viewed from the top (left) and side (right). The three domains of one of the α subunits in the top ring are colored cyan, green & magenta for the equatorial, middle & apical domains, respectively. The central schematic shows how to map the 8 subunits - A, B, G, D, E, H, Q & Z - of TRiC in a proposed arrangement (boxed) onto the top ring of the thermosome. This arrangement is brought as an example only, and is not part of our suggested set of 72.

Throughout the text we present possible subunit arrangements in the ring as a string of eight letters. These refer to the subunit order in a counterclockwise direction in the top ring when viewed from the top. Wherever we refer to the left and right subunits of an interface, we mean the respective interacting subunits in the top ring when viewed from the side (Figure 1, right)[1].

[1] Molecular graphics images were produced using the UCSF Chimera package from the Resource for Biocomputing, Visualization, and Informatics at the University of California, San Francisco (supported by NIH P41 RR-01081) [11].

A position in a certain subunit was considered conserved across species if it showed consistent physio-chemical properties in 11 or more of the species (i.e. mostly hydrophobic, negatively-charged, long-polar etc.). A position was considered as a possible candidate for the molecular key of the subunit order if at least two subunits were conserved across species, but showed remarkably different physio-chemical properties (i.e subunit Z and H in position 1 in Table 1). The definition of the interaction centers was more subjective and was based on manual inspection of clusters of positions that were characteristic of the order key.

3. Results

This study presents a practical approach towards elucidation of the intra-ring subunit order of TRiC by combining two sources of information: evolutionary conservation and a structural hypothesis. Three assumptions are made:

1. The subunit order within the ring is fixed, and each of the eight subunits appears once in the ring. This assumption greatly limits the possible number of arrangements in the ring to 5,040 (=7!). In particular, it removes the burden of analyzing the interactions of any subunit with a same copy of itself.
2. Structural hypothesis. The structural interface between subunits in the ring resembles the interface observed in the solved crystallographic structure of the archaeal thermosome [9]. This assumption is justified by the high sequence identity (~34%) between TRiC and the thermosome, as well as by supporting data from EM structural studies [8]. The structural analysis is therefore based on a straightforward alignment of the bovine TRiC sequences onto the solved thermosome structure.
3. Evolutionary information. The subunit order, as well as the molecular key that determines it, is conserved across different eukaryote species ranging from unicellular organisms to mammals. Strong support for this assumption comes from the much higher sequence identity observed between subunits of the same type across species relative to the identity between different subunit types in the same organism. This assumption is a valuable source of evolutionary information, as it implies that the residues participating in the order key are under selective evolutionary pressure.

Guided by these assumptions, analysis progressed in three steps. In the first step, all the residues in the thermosome structure that participate in an inter-subunit contact (within van der Waals contact distance) were pooled into a putative interface set. In the second step, we checked the evolutionary pattern of each residue in the interface set, and looked for conservation pattern that may confer order information. Specifically, we looked for residue positions that showed considerable conservation across species, but large variation across subunits (see for example Table 1). If several such positions were spatially clustered in the thermosome structure, we pooled them together into a putative interaction center. Five putative interaction centers were identified in this manner, however only two of them are amenable to reliable structural interpretation at this stage. In the last step, we tried to infer a number of structural restrictions from the two final interaction centers. We used these restrictions to discard the majority of the 5,040 subunit arrangements. This step is especially sensitive to the subtle structural differences between TRiC and the thermosome. We, therefore, were very conservative in our arguments, preferring to retain more possible arrangements. By this analysis, we were able to limit the number of possible subunit arrangements by 70 fold to just 72 (Table 3).

3.1. *Interaction Center 1*

In the thermosome, residue D48 from the α subunit makes a partially buried salt bridge with residue R515 from the β subunit (Fig. 2). In this arrangement, the aspartic side-chain is quite buried within the interface, while the arginine side-chain is relatively free to move into the solvent. In the thermosome, this salt bridge is conserved in both the α-β and the β-α interfaces. Table 1 shows the residue types that are observed in these two positions across 13 species (rows) and across the different subunit types (columns). The 13 letters in each subunit and position are arranged in decreasing sequence similarity to the bovine genes, and range from bovine (left) to the unicellular paramecium (right). The table shows a consistent conservation pattern: each position conserves the residue property (charge, hydrophobicity etc.) across species, but large variation occurs between subunit types.

The data suggest that the observed salt bridge from the thermosome is conserved between some of the subunits in TRiC, but is absent in others. Because of the partial burial of the aspartic side-chain, there will be a considerable energetic advantage if a positively-charged side-chain from the next subunit can form a salt bridge with it. Following

this reasoning, the possible subunit arrangements are restricted to those where any of the {A,B,D,E,H} subunits is followed by any one of the {A,B,G,D,E,Q} subunits. The Q subunit is included in the latter group, even though it does not always have a positive charge in that position, so as to make our restrictions as conservative as possible. Applying this first restriction lowers the number of possible subunit arrangements from 5,040 to 480.

A conserved hydrophobic pattern is observed for subunits G and Z in Position 1 and for subunit Z alone in Position 2. Disregarding the Z-Z interface, the G-Z interface forms a very favorable hydrophobic interaction. Furthermore, a hydrophobic residue in Position 2 will cause unfavorable burial of any polar residue in Position 1. This reasoning restricts the subunit arrangements to those with a G-Z interface. Applying this restriction further lowers the number of possible subunit arrangements to 240. The G-Z restriction is validated independently by interaction center 2 (see below).

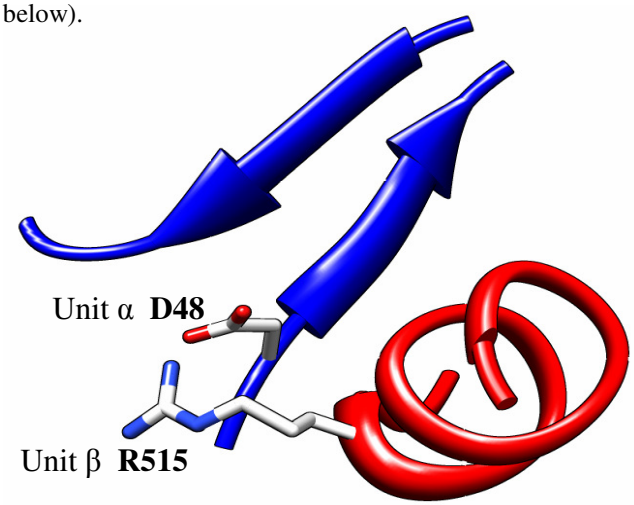


Figure 2. Two complementary residues at the interface of two thermosome subunits. Unit α (blue) and subunit β (red) are the left and right subunits, respectively, in the top ring when viewed from the side of the particle. In TRiC this salt bridge is not conserved between all subunits. The amino acids at these two positions in different organisms are shown in Table 1.

Table 1. The residue types observed in TRiC in the two homologous positions to the two marked thermosome residues in figure 2[a].

	Position 1: Unit α **D48**		Position 2: Unit β **R515**		
A	DDDDDDDDDDDDD	D	RRRRRRRRRRRRR	R	**A**
B	DDDDDDDDDDDDD	D	RRRRRRRRRRRRR	R	**B**
G	MMMLLLLLLLLLL	hp[b]	RRRRRRRRRRRRR	R	**G**
D	DDDDDDDDDDDDD	D	KKKKKKKRRRKKK	[KR]	**D**
E	DDDDDDDDDDDDD	D	KKKKKKKKKKKKK	K	**E**
H	DDDDDDDDDDDDD	D	SSSSSSSSGSSSS	[SG]	**H**
Q	NNNNNNNNNNNNK	[NK]	RRRSKKRSSSRKR	[RSK]	**Q**
Z	MMMMVLILILLYL	hp[b]	LLLLLLLLLLLLL	L	**Z**

[a] For each subunit and position the data from 13 different organisms are shown in the order: Bovine, Human, Zebrafish, Ciona, Drosophila, C. Elegans, Arabidopsis, Yeast, Neospora, Candida, Dictyostelium, Plasmodium and Paramecium. The amino acids that occur in each row are shown to the right of that row.
[b] "hp" is a conserved hydrophobic position.

3.2. *Interaction Center 2*

In the thermosome, the N-terminal part of a helix from the middle domain is interacting with the equatorial domain of the next subunit in the ring (Fig. 3). The interaction involves a glutamic acid side-chain (Position 3) that makes electrostatic interactions with both the positive histidine ring (Position 1) and the dipole-related charged N-terminal of the helix. In addition, the histidine side-chain makes a hydrophobic interaction with the methionine side-chain (Position 2). In TRiC, these positions are altered, but show conservation patterns that are consistent with the order

key (Table 2). As with the previous center, the table refers to adjacent subunits in the ring and the positions are as marked in Figure 3.

Position 3 is central in this network because it interacts with the helix terminal and possibly with all other positions. Most subunits in this position have a conserved a negatively-charged residue, which is compatible with the positive charge of the helix terminal as seen in the thermosome structure. In striking contrast, the Z subunit displays a conservation pattern of positively-charged amino acids. The expected electrostatic repulsion between the helix terminal and the Z subunit can be facilitated if any of the other positions provides a mediating negative charge. The only possibility is Position 1 in the G subunit that conserves negative charge. This match revalidates the G-Z condition we derived previously from interaction center 1. It also suggests that the helix terminal of the G subunit is protruding from the Z subunit, because the interaction is mediated by an additional side-chain.

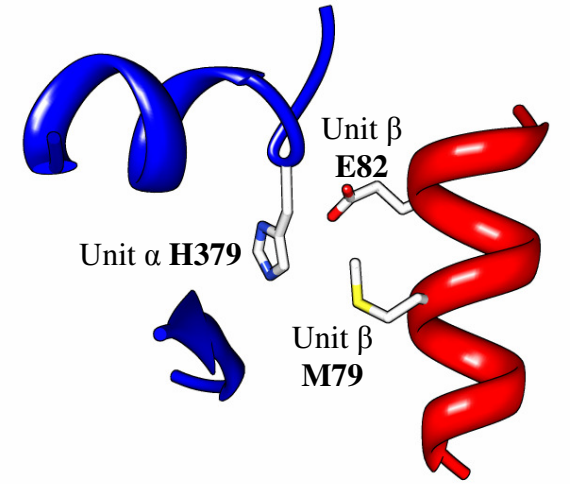


Figure 3. Three complementary residues at the interface between two thermosome subunits. Subunit α (blue) and subunit β (red) are the left and right subunits, respectively, in the top ring when viewed from the side of the particle. The amino acids at these three positions in different organisms are shown in Table 2.

Table 2. The residue types observed in TRiC for the three homologous positions to the three marked thermosome residues in figure 3 [a].

	Position 1: Unit α **H379**		Position 2: Unit β **M79**		Position 3: Unit β **E82**		
A	FFFIFVYYFYLFL	hp	VVVVVVVIIIIIV	hp	EEEEEEEEDEQNE	[EDQN]	**A**
B	QQQQQQHQQQHHH	[QH]	VVVVIVVVVVIII	hp	DDDEDDDNNNDDE	[DNE]	**B**
G	EEEDDDDDDDDDD	[DE]	SSSSSSSSSSSSS	S	EEEEEEEEEEEEE	E	**G**
D	LLLLLLLMLMLLL	hp	MMMMMMMMMMMMM	M	EEEEEEEEQDEEE	[EQD]	**D**
E	MMMMMMMMMMMMM	M	LLLLLLLLLLLLL	L	EEEEQEEQEQQEE	[EQ]	**E**
H	QQQQQQQQQQQQQ	Q	TTTTTTITITTIV	[TIV]	DDDDDDDDDDDDD	D	**H**
Q	NNNNNSSNNNNNN	[NS]	MMMMLILVLVMIM	hp	MMMIMMLMMMMKM	[MILK]	**Q**
Z	HHHHHHHYHYHYH	[HY]	LLLLMMMLMMLMM	hp	KKKKRKRRRRRRR	[RK]	**Z**

[a] See footnotes to Table 2.

Another unusual subunit in Position 3 is subunit Q, which shows a unique hydrophobic conservation. Together with Position 2, subunit Q displays a significant hydrophobic patch at this interaction center. A favorable hydrophobic interaction can form in the interface between this patch and one of the hydrophobic side-chains from Position 1. This reasoning restricts the subunit arrangements in the ring to those where the Q subunit is preceded by one of the subunits from the following group {A,D,E}. Applying this restriction further lowers the number of possible arrangements of subunits in a ring to 144. This interaction is also likely to lead to bulging at the helix terminal, since it does not form an interaction of the helix terminal with a negative side-chain in Position 3.

Finally, Position 2 shows a general hydrophobic conservation pattern. A strong exception is subunit G, which shows a conserved polar pattern (subunit H is similar, but less conserved). A hydrophobic side-chain from Position 1 is likely to bury this serine upon subunit binding. We can therefore forbid subunit arrangements in which the G subunit is preceded by one of the subunits from the group {A,D,E}. Applying this restriction further lowers the number of possible subunit arrangements to in the ring 72. Of all the restrictions discussed so far, this is the one we are least confident with, because of the weaker nature of polar burial as compared to charge burial. However, the two-fold reduction in possible subunits arrangements makes this restriction valuable.

Table 3. The 72 possible subunit arrangements in the ring sorted alphabetically.

ABDEQGZH	ABDEQHGZ	ABDQGZHE	ABDQHGZE	ABEDQGZH	ABEDQHGZ
ABEQGZHD	ABEQHGZD	ABGZDEQH	ABGZDQHE	ABGZEDQH	ABGZEQHD
ABGZHDEQ	ABGZHDQE	ABGZHEDQ	ABGZHEQD	ADBEQGZH	ADBEQHGZ
ADBGZEQH	ADBGZHEQ	ADEQBGZH	ADEQGZHB	ADEQHBGZ	ADEQHGZB
ADQBGZHE	ADQEBGZH	ADQGZHBE	ADQGZHEB	ADQHBGZE	ADQHEBGZ
ADQHGZBE	ADQHGZEB	AEBDQGZH	AEBDQHGZ	AEBGZDQH	AEBGZHDQ
AEDQBGZH	AEDQGZHB	AEDQHBGZ	AEDQHGZB	AEQBGZHD	AEQDBGZH
AEQGZHBD	AEQGZHDB	AEQHBGZD	AEQHDBGZ	AEQHGZBD	AEQHGZDB
AQBGZHDE	AQBGZHED	AQDBGZHE	AQDEBGZH	AQEBGZHD	AQEDBGZH
AQGZHBDE	AQGZHBED	AQGZHDBE	AQGZHDEB	AQGZHEBD	AQGZHEDB
AQHBGZDE	AQHBGZED	AQHDBGZE	AQHDEBGZ	AQHEBGZD	AQHEDBGZ
AQHGZBDE	AQHGZBED	AQHGZDBE	AQHGZDEB	AQHGZEBD	AQHGZEDB

Table 4. Number of occurrences of allowed subunits pairing in the 72 possible ring arrangements

Subunit	A	B	G	D	E	H	Q	Z	#Partners
A	-	16	-	16	16	-	24	-	4
B	12	-	36	12	12	-	-	-	4
G	-	-	-	-	-	-	-	72	1
D	16	16	-	-	16	-	24	-	4
E	16	16	-	16	-	-	24	-	4
H	14	12	18	14	14	-	-	-	5
Q	4	6	18	4	4	36	-	-	6
Z	10	6	-	10	10	36	-	-	5
#Partners	6	6	3	6	6	2	3	1	

The 72 possible arrangements of subunits we find here only sample 33 of the 56 different possible subunit pairings (see Table 4), and not in equal measure. This distribution could be used in experiment planning that can determine subunit pairing (such as the double immuno-labeling and cryo-EM reconstruction used by Martin-Benito *et al*. [12]). For example, an experiment of double labeling with antibodies against the Q and H subunits is guaranteed to lower the number of possible arrangements to 36. If the particle reconstruction reveals the extra density due to the bound antibodies on successive subunits, then the only possible pairing is QH as pairing HQ has already been eliminated. On the other hand, if the extra densities are further apart around the ring, then the only possible pairing for subunit H is ZH as there are only two allowed pairings with H to the right, QH and ZH.

4. Discussion

In this study we used evolutionary information and a structural model to narrow the number of possible subunit arrangements in the ring of TRiC from 5,040 to 72 (Table 3). This specific type of data analysis has already been used effectively in many other studies [13-15], but its application to TRiC is novel both in terms of the magnitude of the problem as well as in the level of detail demanded. Two sources of errors might jeopardize the validity of the results: wrong restrictions due to misinterpretation of the structural model or, more likely, because the model itself contains errors that reflect the difference between TRiC and the thermosome. These concerns are always inherent to this type of methodology, and we tried to alleviate them by adherence to the most conservative possible assumptions and views of the data. Somewhat paradoxically, the lack of great detail in our assumptions actually makes them more robust to errors in the model: minor changes in backbone or side-chain conformations are unlikely to break them.

We emphasize that the five positions described in the tables are not the entire molecular key for the subunit order. At least three other putative interaction centers show conservation patterns that are likely to be important for the interface specificity. These centers were not used here because we are currently unsure of their interpretations for two main reasons. First, some of these centers are more complex and involve three interactions or more. The analysis of so many interactions requires a more rigorous modeling of the side-chain positions. Second, in some areas a significant backbone change is likely as large variations are already evident between the two thermosome subunits (loop 53-57 for example). Analysis of these areas requires consideration for backbone flexibility.

Our results start to reveal the molecular basis for the subunit order within the ring. It appears that a hydrophobic core, which is conserved across all subunits, provides a general scaffold for the interface between two successive subunits in the ring. This scaffold is bordered by interaction centers that are subunit-pair specific. The subunit order results from complementary charged, polar, and (to lesser extent) hydrophobic interactions that make the correct pairing more favorable than the alternatives. The subunit similarity excludes the possibility of significant steric clashes as a source of order information. Only the equatorial domains are part of the general hydrophobic interface. Major ordering information is encoded in Interaction Center 2 from the middle domain. Since the middle domain mediates between the ATP site and the lid, it is likely that the subunit specificity in interaction center 2 has functional relevance.

In order to completely determine the subunit arrangement in the ring, we see two possible courses of action. Perhaps most intuitively, structural data from other studies could be used to further validate and limit the set of subunit arrangements presented here. We believe that EM structural studies will be extremely valuable in that context. Alternatively, more detailed modeling could lead to additional structural insights from the unexplored interaction centers, as well as to tighter restrictions from the two interaction centers already described. That level of detail requires the use of more computer intensive approach with explicit modeling of side-chain conformations and backbone flexibility. We are currently testing various computational tools for that purpose and hope to make further progress in the future.

4.1. *Inconsistencies with the Liou and Willison model*

Liou and Willison [16] suggested an arrangement of subunits in the ring that is still in use. They inferred this arrangement from 6 association patterns of TRiC micro-complexes, i.e. very partial fragments of TRiC rings found in cell extracts. Unfortunately, none of the 72 possibilities that we summarize in Table 3 is consistent with the Liou and Willison arrangement. The G-Z interaction is among the first restrictions to prohibit their arrangement, which does not include it. However, we see a very strong signal for its occurrence in this work. Furthermore, out of the 72 possibilities we suggest, only one is consistent with 4 out of the 6 association patterns from the Liou and Willison work, and the rest are consistent with 3 or less. The clear inconsistency between the two studies suggests that the micro-complexes they identified are not always genuine parts of the intact complex.

Acknowledgements
This study was supported by the BioX program as well as NIH Nanomedicine award 2PN2EY016525-02 (Wah Chiu PI).

References

1. C. Spiess, A. S. Meyer, S. Reissmann, J. Frydman, *Trends Cell Biol*. **14**, 598 (2004)
2. Y. Gao, J. O. Thomas, R. L. Chow, G. H. Lee and N. J. Cowan, *Cell*. **69**, 1043 (1992)
3. Y. Gao, I. E. Vainberg, R. L. Chow and N. J. Cowan, *Mol Cell Biol*. **13**, 2478 (1993)
4. J. Frydman, E. Nimmesgern, H. Erdjument-Bromage, J. S. Wall, P. Tempst and F. U. Hartl, *EMBO J*. **11**, 4767 (1992)
5. C. Spiess, E. J. Miller, A. J. McClellan and J. Frydman, *Mol Cell*. **24**, 25 (2006)
6. S. Tam, R. Geller, C. Spiess and J. Frydman, *Nat Cell Biol*. **8**, 1155 (2006)
7. D. Rivenzon-Segal, S. G. Wolf, L. Shimon, K. R. Willison and A. Horovitz, *Nat Struct Mol Biol*. **12,** 233 (2005)
8. C. R. Booth, A. S. Meyer, Y. Cong, M. Topf, A. Sali, S. J. Ludtke, W. Chiu and J. Frydman, *Nat Struct Mol Biol*. **15**, 746 (2008)
9. L. Ditzel, J. Löwe, D. Stock, K. O. Stetter, H. Huber, R. Huber and S. Steinbacher, *Cell*. **93**, 125 (1998)
10. J. S. Papadopoulos and R. Agarwala, *Bioinformatics*. **23,** 1073 (2007)
11. E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng and T. E. Ferrin, *J Comput Chem*. **25**, 1605 (2004)
12. J. Martín-Benito, J. Grantham, J. Boskovic, K. I. Brackley, J. L. Carrascosa, K. R. Willison and J. M. Valpuesta, *EMBO Rep*. **8**, 252 (2007)
13. A. Armon, D. Graur and N. Ben-Tal, *J Mol Biol.* **307,** 447 (2001)
14. S. Ahmad, O. Keskin, A. Sarai and R. Nussinov, *Nucleic Acids Res*. **36**, 5922 (2008)
15. S. Engelen, L. A. Trojan, S. Sacquin-Mora, R. Lavery and A. Carbone, *PLoS Comput Biol*. **5**, e1000267 (2009)
16. A. K. Liou and K. R. Willison, *EMBO J*. **16,** 4311 (1997).

"CROSS-GRAINING:" EFFICIENT MULTI-SCALE SIMULATION VIA MARKOV STATE MODELS

PETER M. KASSON
Departments of Chemistry and Structural Biology , Stanford University
Stanford, CA 94305 USA

VIJAY S. PANDE
Departments of Chemistry and Structural Biology , Stanford University
Stanford, CA 94305 USA

Accurate and efficient methods to simulate biomolecular systems at multiple levels of detail simultaneously are an ongoing challenge for the simulation community. Here we present a new method for multi-scale simulation where a complex system can be partitioned into two loosely-coupled sub-systems, one coarse-grained and one atomistic. If the coupling between the coarse-grained and atomistic systems can be encoded into discrete states that interconvert slowly, we can construct a Markov state model where we approximate any given transition $P[(s_i,t_j)->(s_k,t_l)]$ in the joint space of the coarse-grained and atomistic systems as the product of two orthogonal transitions $P(s_i->s_k \mid t_j)$ and $P(t_j->t_l \mid s_j)$. We provide a formalism for constructing such models and describe how they may be applied to multi-scale simulation of membrane proteins. This "cross-graining" methodology may provide a general means to efficiently simulate mixed-scale systems.

1. Introduction

Many biological systems have the following canonical challenge: outcomes depend on fine degrees of freedom, yet they involve large systems and large timescales. Indeed, an underlying principle behind biological signal transduction is that fine sensitivity to subtle molecular changes controls large-scale effects. This poses a challenge for mechanistic simulation; it is often computationally intractable to capture all the fine-scale details of a system while simultaneously reaching the time and length scales required to address emergent behavior.

Driven by this challenge, "coarse-grained" approaches have been developed to use a reduced degree-of-freedom representation for more efficient computation. However, many systems have important transitions governed by fine-grained degrees of freedom such that it is challenging to derive a coarse-grained representation that is computationally efficient yet sufficiently predictive. Full mixed-scale simulation, in which fine-grained and coarse-grained representations can be combined freely, is an ongoing challenge for the field. In this work, we derive an alternate multi-scaling technique where fine-grained and coarse-grained representations are combined at the level of a statistical reaction model, a Markov state model. We show that under certain conditions this method, which we term "cross-graining," performs at least as well as the ideal mixed-scale simulation that has yet to be fully realized.

Whether atomistic or coarse-grained representations are used, Markov state models have recently gained traction as a means to analyze and interpret molecular simulation data [1-4]. They draw on the notion that at some level molecular reactions are Markov chains—stochastic processes where, given a suitable encoding of state information at time t and a minimum "lag time" τ, the state of the system at time $t + \tau$ depends only on the state at t. Methods have recently been developed to determine such state encodings from a set of simulation trajectories, assess model quality, and use the resulting statistical reaction model to predict ensemble properties and long-timescale behavior of the system [5-7].

Here we address an important subset of "multi-scale" problems where one can partition a system into two processes of interest, each with associated degrees of freedom. For instance, one process may depend primarily on atomistic degrees of freedom $\{x_i\}$, while another process depends primarily on coarse-grained degrees of freedom $\{y_i\}$. An example is given in Figure 1. As we will discuss, these processes can be correlated. We show how Markov state models yield a means to obtain multi-scale dynamics in this case and further derive a scheme for "cross-graining" where each set of transitions can be computed using only the relevant degrees of freedom. We

provide an example of how this approach yields efficient sampling compared even to a full "ideal" multi-scale simulation.

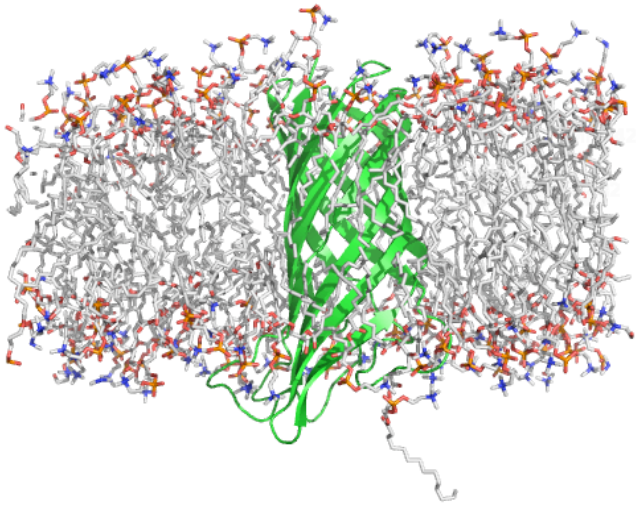


Figure 1. Sample membrane-protein system. Shown here is the *E. coli* outer membrane protein A (OmpA) inserted in a lipid bilayer [8]. OmpA is unfolded or aggregated in solution but assumes a beta-strand conformation in membranes; these strands then assemble to form a beta-barrel pore. To simulate this in a multi-scale manner, we consider the protein and its immediate environment atomistically as degrees of freedom $\{x_i\}$ and the larger membrane-water system as degrees of freedom $\{y_i\}$.

We motivate our cross-graining approach by considering the insertion and assembly of membrane pores. Contemporary coarse-gained force fields such as MARTINI [9] have had good success in simulating lipid membranes [10-16], and they have also been used to simulate protein assembly in membranes, given a constrained protein conformation [8, 17-19]. Robust simulation of protein refolding has proven more challenging. We seek an approach that can use atomistic simulation to determine the protein conformational dynamics within each coarse-grained "state" of the membrane system and also use the protein conformation in each state to inform a coarse-grained simulation of large-scale protein-membrane interactions—in this case by providing data for the coarse-grained protein torsions and conformational restraints. Grossly, one could imagine the fine-grained instantiations of each coarse-grained state as simulating the protein in solution, the protein in an encounter complex with the membrane, and a variety of membrane-inserted states. The coarse-grained state would determine the composition and macroscopic restraints on the protein surroundings for the fine-grained simulation; the fine-grained state would yield torsional restraints to constrain the protein for the coarse-grained simulations. Under the proper conditions, such an approach can provide both fine-scale conformational dynamics and efficient simulation of large systems. The cross-graining method we have developed addresses these challenges, although it also proves more general as well.

2. Theory

Let us consider a molecular system where the degrees of freedom can be partitioned into two sets, $\{x_i\}$ and $\{y_i\}$. Let us further consider two sets of processes, one of which primarily depends on $\{x_i\}$ (our fine-grained degrees of freedom) and one of which primarily depends on $\{y_i\}$ (our coarse-grained degrees of freedom). These two processes can be inter-dependent; we will both state this formally below and show a simple 2D example to make this intuitive. The fundamental approximation, however, is that at some level of granularity we can treat the joint free energy landscape $(\{x_i\},\{y_i\})$ as the direct product of the individual free energy landscapes $\{x_i\} \otimes \{y_i\}$.

Any classical molecular system can be represented as a Markov chain for some set of states and some time resolution [1]. A trivial proof of this is to consider the set of states corresponding to all points in phase space and

continuous time (or for a simulation, time resolution corresponding to the integrator time step). The time evolution of this system is a history-independent stochastic process. Therefore suppose we construct a Markov state model to represent the behavior of our system on degrees of freedom $\{x_i\}$ and a second Markov state model to represent the behavior on degrees of freedom $\{y_i\}$. For the moment, we will take the partitioning of degrees of freedom as given; determining the optimal partitioning of phase space into states and the optimal partitioning of degrees of freedom into $\{x_i\}$ and $\{y_i\}$ are separate problems, each non-trivial.

Since these degrees of freedom $\{x_i\}$ and $\{y_i\}$ may be coupled, any transition a -> b on $\{x_i\}$ will be Markovian on $\{x_i\}$ but may depend on $\{y_i\}$. Stated more formally, given a sequence of states $(u_1..u_n)$ on $\{x_i\}$ and a sequence of states $(v_1..v_n)$ on $\{y_i\}$, the mutual information $I(u_{n+1} ; u_n \mid u_1..u_{n-1}) = 0$ [20] but $I(u_{n+1} ; u_n \mid v_n) \geq 0$. We now partition state space $\{x_i\}$ into states $\{s_i\}$ and $\{y_i\}$ into "macrostates" $\{t_i\}$. We choose a set of macrostate definitions for S and T such that knowledge of (s,t) is sufficient information to make $P(a \rightarrow b)$ Markovian by the above criterion. In the limit of completely uncoupled degrees of freedom, there is no dependence on the state Y, while in the opposite limit of completely coupled degrees of freedom one could have a system where each point in phase space $\{y_i\}$ corresponds to a distinct state.

To make this intuitive, we consider the case where x and y each consist of one degree of freedom. A free energy diagram in this 2D space is schematized in Figure 2. Cross-graining applies well when each set of $(s_i,t_j) \rightarrow (s_k,t_l)$ transitions occurs on well-separated timescales. In one such case we may have $s_i \rightarrow s_k$ transitions followed sequentially by $t_j \rightarrow t_l$ transitions (Fig. 2a); in another the $s_i \rightarrow s_k$ transitions relax quickly enough compared to $t_j \rightarrow t_l$ (Fig. 2b) that we can make the following approximation:

$$P\left(\left(\{s\},t_j\right) \rightarrow \left(\{s\},t_l\right)\right) \approx \sum_{s_i \in \{s\}} P\left(s_i\right) \cdot P\left(\left(t_j \mid s_i\right) \rightarrow \left(t_l \mid s_i\right)\right) \qquad (1)$$

Cross-graining performs poorly where fast or highly cooperative transitions exist on both $\{x_i\}$ and $\{y_i\}$ (Fig. 2c); this can in principle be solved by using a shorter Markov timescale τ, but this comes at the cost of a greatly increased sampling requirement and thus can become prohibitively expensive.

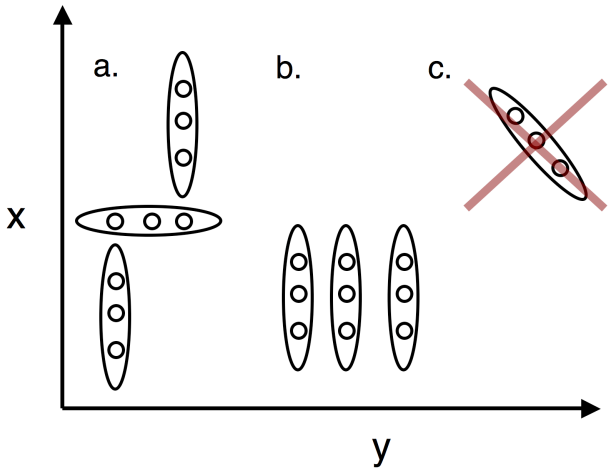


Figure 2. Two-dimensional free energy diagram. This schematic contour plot shows state configurations well suited (a and b) and poorly suited (c) for cross-graining.

Let us leave the problem of state selection aside for the moment and assume that we can select states, either using prior experimental knowledge or by applying a clustering algorithm to an initial sampling of state space [6, 7]. To specify the matrix of transition probabilities $P(s_i \rightarrow s_j \mid t \in T, \tau_1)$ we can sample trajectories in $\{x_i\}$ using a sampling algorithm of our choice while holding t constant and repeating this for each t in T. We perform the same operation on $\{y_i\}$ while holding s constant for each state s in S. This sampling (where one might use molecular

dynamics simulation in a molecular system) illustrates the "cross-grained" nature of our approach—when sampling on $\{x_i\}$, the information encoded in the state variable t provides sufficient information regarding $\{y_i\}$, the other degrees of freedom are effectively coarse-grained away. The converse applies to sampling $\{y_i\}$. By treating each subspace separately in this manner, we can substantially ease the task of sampling phase space.

To use the example of protein insertion and assembly with the MARTINI force field, the atomistic representation would include the protein and, depending on the coarse-grained state, the surrounding water or membrane. These surroundings encode the coarse-grained state information. The coarse-grained representation contains the full system and encodes the atomistic state by using the protein secondary structure to determine torsional potentials (as is standard in MARTINI) and optionally some additional conformational restrains (c.f. the work of Marrink, Sansom and co-workers [17, 21]).

Given the transition probabilities $P(s_i \rightarrow s_j \mid t, \tau_1)$, $P(t_i \rightarrow t_j \mid s, \tau_2)$, we can then construct a single first-order Markov state model on the joint space (s_i,t_j). If we formulate the transition probabilities on $\tau_3 >= \max(\tau_1, \tau_2)$, $P[(s_i,t_j)\rightarrow(s_k,t_l) \mid \tau_3] = P(s_i \rightarrow s_k \mid t_j, \tau_3) P(t_j \rightarrow t_l \mid s_i, \tau_3)$. This approximation of conditional independence is a key one; transitions that are highly cooperative between coarse-grained and fine-grained degrees of freedom will result in a poor approximation here. This can be partially addressed with a finer sampling of state space, but at a cost of increased sampling. Thus, highly cooperative transitions are an area for further methods development. For most cases, however, this approximation provides an MSM describing the full state of the system, which we can query for equilibrium probabilities, kinetic properties, ensemble experimental observables, etc. much as any single-resolution Markov state model [1, 5, 16].

3. A simple model for the insertion and assembly of a membrane pore

Let us consider a simple system for which phase space can be partitioned into degrees of freedom $\{x_i\}$ and $\{y_i\}$ and that has the following transitions:

$$\begin{aligned} \text{on } \{x_i\}: \quad & A \underset{k_4}{\overset{k_1}{\rightleftharpoons}} I_1 \underset{k_5}{\overset{k_2}{\rightleftharpoons}} I_2 \underset{k_6}{\overset{k_3}{\rightleftharpoons}} B \\ \text{on } \{y_i\}: \quad & C \underset{k_{10}}{\overset{k_7}{\rightleftharpoons}} I_3 \underset{k_{11}}{\overset{k_8}{\rightleftharpoons}} I_4 \underset{k_{12}}{\overset{k_9}{\rightleftharpoons}} D \end{aligned} \qquad (2)$$

and where the states $\{s_1..s_4\} = \{A,I_1,I_2,B\}$ and $\{t_1..t_4\}=\{C,I_3,I_4,D\}$ are Markovian given the joint state of the system (s_i,t_j). These states and the rates given below were chosen to loosely resemble a simple model for the insertion and assembly of a membrane pore, similar to the rough model proposed for the assembly of hemolysin E [22]. The fine-grained state A corresponds to the solution protein conformation, I1 to a hydrophobic-exposed protein conformation in contact with the membrane, I2 to a membrane-inserted and refolded conformation, and B to the protein conformation in the assembled pore. The coarse-grained state C corresponds to the protein distant from the membrane, I3 to the protein at the membrane interface, I4 to the protein inserted in the membrane, and D to the fully assembled pore. A Markov model in full molecular detail for this process would likely have many more states, but this provides an experimentally-motived simple model for our proof of principle. If we have knowledge of the macrostate definitions i.e. the mappings X->S and Y->T but not the transition rates $\{k\}$, we can set up the following sampling approach to estimate the rates:

1. Start n simulation trajectories on $\{x_i\}$ for each macrostate t on $\{y_i\}$; these can be from a single start state (e.g. A) or distributed among start states from a random seeding approach or a more sophisticated sampling scheme such as that described in [23]. Similarly start m simulation trajectories on $\{y_i\}$ for each macrostate s on $\{x_i\}$. Using the state mappings given, this will yield a set of trajectories each of the form $(u_1..u_n \mid t_i)$ where the macrostate ti remains fixed and $(v_1..v_n \mid s_i)$ where s_i remains fixed. Macrostate mappings can also be iteratively refined at this stage.
2. Estimate the transition probabilities for each of these Markovian sub-graphs:

$$P\left(s_i \rightarrow s_j \mid t_k\right) = \frac{Count\left(u_n \in s_i, u_{n+1} \in s_j \mid t_k\right)}{Count\left(u_n \in s_i \mid t_k\right)} \quad (3)$$

3. Construct the transition probabilities for the combined Markovian state model $P[(s_i,t_j)\text{->}(s_k,t_l) \mid \tau_3] = P(s_i\text{->}s_k \mid t_j, \tau_3)P(t_j\text{->}t_l \mid si, \tau_3)$.

We demonstrate this approach by sampling n random trajectories on $\{x_i\}$ for each macrostate $t \in T$ using a transition probability matrix constructed from the rates $\{k\}$ listed in Table 1 for the reaction scheme given above. We similarly take n random trajectories on $\{y_i\}$ for each macrostate $s \in S$. Starting states are randomly selected from a uniform distribution over macrostates. We construct the transition probabilities for the combined Markovian state model as specified above using our "observed" trajectories and compare to the same number of $8n$ total trajectories sampled directly from the combined transition probability matrix. We let n vary between 20 and 100, perform 100 random sampling procedures of this nature for each value of n and compare the convergence of the transition probability matrix eigenvalues as performed previously [23]. Transition probability eigenvalues yield the implied timescales for the system and are hence an important target for validation. Results are plotted in Figure 3.

Table 1. Rate constants used for simple system. First-order rate constants are given in reduced units of time τ.

		Macrostate on $\{y_i\}$			
		C (solution)	I3 (interfacial)	I4 (inserted)	D (pore)
Rate	k_1	1.00E-09	1.00E-05	1.00E-05	1.00E-05
	k_2	1.00E-09	1.00E-05	1.00E-05	1.00E-05
	k_3	1	1.00E-07	1.00E-05	1.00E-05
	k_4	1	1.00E+00	1.00E-05	1.00E-09
	k_5	1	1.00E-05	1.00E-07	1.00E-09
	k_6	1	1.00E-05	1.00E-09	1.00E-09

		Macrostate on $\{x_i\}$			
		A (solution conformation)	I1 (hydrophobic exposure)	I2 (partially refolded)	B (membrane fold)
Rate	k_7	1.00E-05	1.00E-03	1.00E-03	1.00E-03
	k_8	1.00E-09	1.00E-05	1.00E-03	1.00E-03
	k_9	1.00E-09	1.00E-09	1.00E-05	1.00E-05
	k_{10}	1	1.00E-05	1.00E-05	1.00E-05
	k_{11}	1	1.00E-05	1.00E-07	1.00E-09
	k_{12}	1.00E-05	1.00E-09	1.00E-09	1.00E-09

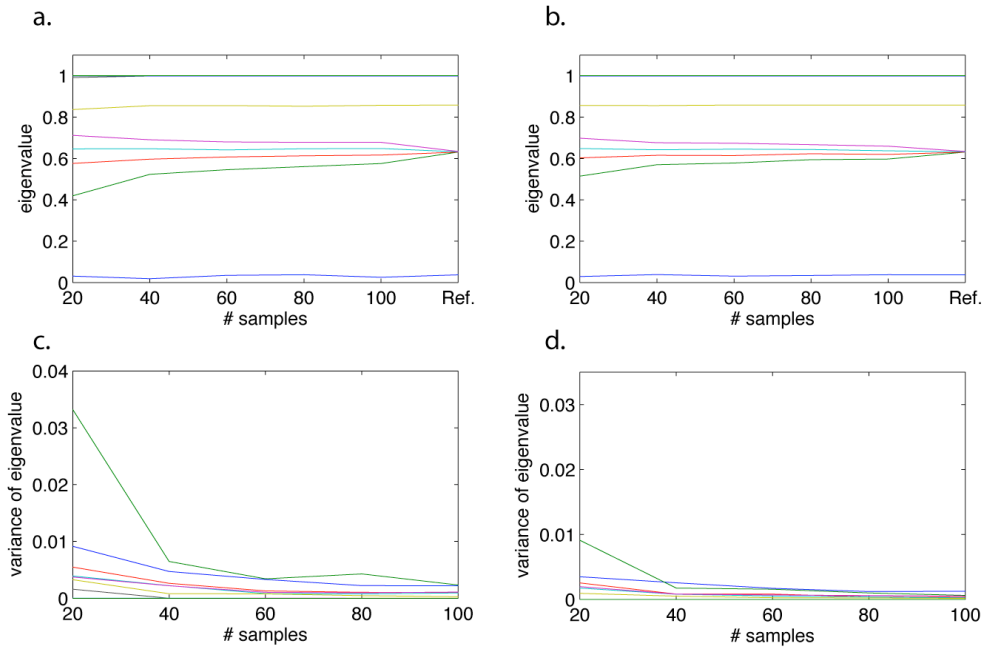


Figure 3. Convergence of eigenvalues for cross-graining. Convergence is compared between cross-grained sampling and direct sampling on the combined transition probability matrix. Panels (a) and (b) show convergence of the mean of each eigenvalue across 100 random samples, while panels (c) and (d) show convergence of the variance.

Convergence of the relaxation timescales yielded by sampling with our "cross-graining" strategy are compared to convergence with an "ideal" multi-scale simulation that can seamlessly mix coarse-grained and atomistic degrees of freedom (mixed CG/AA simulation). Both systems show good convergence over the range 40-100 trajectories of length 1000 τ in each of 8 start state combinations. The clear advantage for cross-graining becomes evident, however, when one considers the compute time required.

We examine computational efficiency in two respects, first with regard to the efficiency of the individual simulations and with regard to the sampling techniques employed. In each of these, cross-graining performs at least as well as the optimal mixed CG/AA simulation. Within the individual simulations, cross-graining allows the coarse-grained and atomistic simulations to be effectively decoupled and run in parallel. This results in optimal scaling across processors and performance equivalent to the hypothetical perfect multi-scale integrator, in which force calculations and coordinate updates can be performed at different timesteps for different parts of the system with no overhead.

Cross-graining also performs at least as well as optimal mixed CG/AA simulation with regard to sampling efficiency. If we have a naive sampling approach where we uniformly sample transitions in the macrostate space (s_i, t_j), cross-graining and mixed CG/AA simulation are equivalent. This equivalence also assumes that for every (s_i, t_j) start state we require even sampling of the s_i -> s_k transitions and the t_j -> t_l transitions. We have taken such an even-sampling approach in the example above, but recent work has shown a substantial advantage for "adaptive sampling" methods that weight the distribution of starting states according to their contribution to the variance of transition matrix eigenvalues [23]. In the case of villin headpiece folding, such a strategy improved convergence by a factor of over 1000. For a given (s_i,t_j) pair, unless we require precisely the same degree of sampling from $s_i \mid t_j$ and $t_j \mid s_i$, we "waste" the sampling across the other degrees of freedom. Since multi-scale simulation typically targets systems where coarse-grained degrees of freedom relate to long-timescale phenomena and atomistic degrees of freedom relate to shorter-timescale phenomena, we end up performing atomistic simulations over the coarse-grained timescales. Using the hemolysin example discussed above, the protein alone comprises 100,000 atoms; if we consider the solution case, simulate water molecules within 1 nm of the protein, and take τ = 1ns, then every

unnecessary trajectory of the resulting 375,000 atom system sampled at length 1000τ would take several months of simulation time using 128 Intel Cloverton cores (benchmarked with Gromacs 4.0 [24]).

4. Outlook

The "cross-graining" approach that we describe is particularly amenable to problems such as insertion, refolding, and assembly of membrane protein oligomers. Membranes and membrane proteins have long been a target for coarse-grained simulation [17, 25, 26], and assembly of folded protein subunits has been studied by several researchers, as has the insertion of pre-formed helices. However, the process of insertion and folding of peptides that have different conformations in solution and in the membrane is more challenging for coarse-grained simulation [27]. Even more difficult are large systems such as E. coli hemolysin (Figure 4) that are are thought to undergo partial unfolding and refolding at the membrane interface prior to assembly in the membrane [22]. One approach to cross-graining such a system is to coarse-grain the hemolysin-membrane-water system with states representing protein in solution, protein at the membrane interface, inserted protein, and various intermediates in assembly. The hemolysin monomer and its immediate surroundings (solution or membrane) are represented atomistically. Sampling the atomistic system will yield Markov state models for the conformational dynamics of the protein in each of the coarse-grained states—data that are difficult to compute accurately in the coarse-grained system alone. Sampling the coarse-grained system will yield the long-timescale behavior of protein insertion and pore formation, data that are not computationally tractable via atomistic systems. Experimental data from crystal structures and biochemical studies can guide the initial state definitions; this approach also provides a means to computationally evaluate structural models for refolding, insertion, and assembly such as those recently proposed for hemolysin [22].

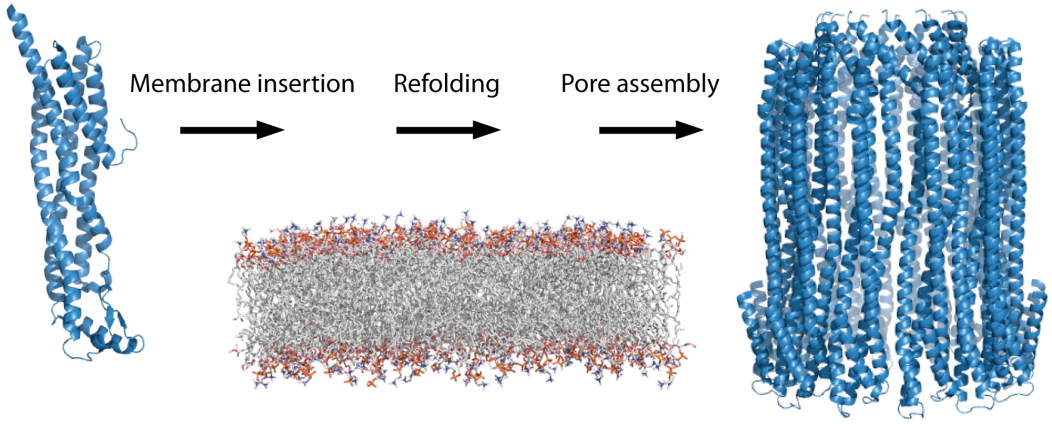


Figure 4. Solution-state and assembled conformations of E. coli hemolysin E.

5. Conclusions

We have presented a means to construct unified multi-scale kinetic models by performing individual simulations at a single scale and combining the simulations and scales in a Markov state model. This is accomplished by partitioning a system into two sets of degrees of freedom that are loosely coupled. This loose coupling and the specification of a discrete number of Markovian "macrostates" allows the approximation of any given transition $P[(s_i,t_j)\text{->}(s_k,t_l)]$ as the product of two orthogonal transitions $P(s_i\text{->}s_k \mid t_j)$ and $P(t_j\text{->}t_l \mid s_j)$. We outline how this method applies to membrane protein simulations, but we anticipate it will prove a much more general method for performing multi-scale simulation, helping to overcome the challenges of simultaneous mixed-scale simulation.

Acknowledgments

PK was supported by a fellowship from the Berry Foundation. The authors would like to thank S. Bacallado, G. Bowman, and D. Ensign for many helpful discussions.

References

1. Swope, W.C., J.W. Pitera, and F. Suits, *Describing protein folding kinetics by molecular dynamics simulations. 1. Theory.* Journal of Physical Chemistry B, 2004. **108**(21): p. 6571-6581.
2. Buchete, N.V. and G. Hummer, *Peptide folding kinetics from replica exchange molecular dynamics.* Phys Rev E Stat Nonlin Soft Matter Phys, 2008. **77**(3 Pt 1): p. 030902.
3. Sriraman, S., I.G. Kevrekidis, and G. Hummer, *Coarse master equation from Bayesian analysis of replica molecular dynamics simulations.* J Phys Chem B, 2005. **109**(14): p. 6479-84.
4. Noe, F. and S. Fischer, *Transition networks for modeling the kinetics of conformational change in macromolecules.* Curr Opin Struct Biol, 2008. **18**(2): p. 154-62.
5. Singhal, N., C.D. Snow, and V.S. Pande, *Using path sampling to build better Markovian state models: Predicting the folding rate and mechanism of a tryptophan zipper beta hairpin.* Journal of Chemical Physics, 2004. **121**(1): p. 415-425.
6. Bowman, G.R., X. Huang, and V.S. Pande, *Using generalized ensemble simulations and Markov state models to identify conformational states.* Methods, 2009.
7. Chodera, J.D., et al., *Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics.* J Chem Phys, 2007. **126**(15): p. 155101.
8. Sansom, M.S., K.A. Scott, and P.J. Bond, *Coarse-grained simulation: a high-throughput computational approach to membrane proteins.* Biochem Soc Trans, 2008. **36**(Pt 1): p. 27-32.
9. Marrink, S.J., et al., *The MARTINI force field: coarse grained model for biomolecular simulations.* J Phys Chem B, 2007. **111**(27): p. 7812-24.
10. Marrink, S.J. and A.E. Mark, *Effect of undulations on surface tension in simulated bilayers.* Journal of Physical Chemistry B, 2001. **105**(26): p. 6122-6127.
11. Faller, R. and S.J. Marrink, *Simulation of domain formation in DLPC-DSPC mixed bilayers.* Langmuir, 2004. **20**(18): p. 7686-93.
12. Marrink, S.J., J. Risselada, and A.E. Mark, *Simulation of gel phase formation and melting in lipid bilayers using a coarse grained model.* Chemistry and Physics of Lipids, 2005.
13. Izvekov, S. and G.A. Voth, *Multiscale coarse graining of liquid-state systems.* J Chem Phys, 2005. **123**(13): p. 134105.
14. Shi, Q. and G.A. Voth, *Multi-scale modeling of phase separation in mixed lipid bilayers.* Biophys J, 2005. **89**(4): p. 2385-94.
15. Ayton, G.S. and G.A. Voth, *Mesoscopic lateral diffusion in lipid bilayers.* Biophys J, 2004. **87**(5): p. 3299-311.
16. Kasson, P.M., et al., *Ensemble molecular dynamics yields sub-millisecond kinetics and intermediates of membrane fusion.* . Proc Natl Acad Sci U S A, 2006. **103**(32): p. 11916-21.
17. Bond, P.J. and M.S. Sansom, *Insertion and assembly of membrane proteins via simulation.* J Am Chem Soc, 2006. **128**(8): p. 2697-704.
18. Carpenter, T., et al., *Self-assembly of a simple membrane protein: coarse-grained molecular dynamics simulations of the influenza M2 channel.* Biophys J, 2008. **95**(8): p. 3790-801.
19. Ayton, G.S., P.D. Blood, and G.A. Voth, *Membrane remodeling from N-BAR domain interactions: insights from multi-scale simulation.* Biophys J, 2007. **92**(10): p. 3595-602.
20. Park, S. and V.S. Pande, *Validation of Markov state models using Shannon's entropy.* J Chem Phys, 2006. **124**(5): p. 54118.
21. Yefimov, S., et al., *Mechanosensitive membrane channels in action.* Biophys J, 2008. **94**(8): p. 2994-3002.
22. Mueller, M., et al., *The structure of a cytolytic alpha-helical toxin pore reveals its assembly mechanism.* Nature, 2009. **459**(7247): p. 726-30.
23. Hinrichs, N.S. and V.S. Pande, *Calculation of the distribution of eigenvalues and eigenvectors in Markovian state models for molecular dynamics.* The Journal of chemical physics, 2007. **126**(24): p. 244101.
24. Hess, B., et al., *GROMACS 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation.* Journal of Chemical Theory and Computation, 2008. **4**: p. 435-447.
25. Shih, A.Y., et al., *Coarse grained protein-lipid model with application to lipoprotein particles.* J Phys Chem B, 2006. **110**(8): p. 3674-84.

26. Marrink, S.J., A.H. de Vries, and A.E. Mark, *Coarse grained model for semiquantitative lipid simulations.* Journal of Physical Chemistry B, 2004. **108**(2): p. 750-760.
27. Marrink, S.J., A.H. de Vries, and D.P. Tieleman, *Lipids on the move: simulations of membrane pores, domains, stalks and curves.* Biochim Biophys Acta, 2009. **1788**(1): p. 149-68.

TOWARD UNDERSTANDING ALLOSTERIC SIGNALING MECHANISMS IN THE ATPASE DOMAIN OF MOLECULAR CHAPERONES

YING LIU AND IVET BAHAR

Department of Computational Biology, School of Medicine, University of Pittsburgh
3064 BST3, 3501 Fifth Avenue
Pittsburgh, PA 15213, USA

The ATPase cycle of the heat shock protein 70 (HSP70) is largely dependent on the ability of its nucleotide binding domain (NBD), also called ATPase domain, to undergo structural changes between its open and closed conformations. We present here a combined study of the Hsp70 NBD sequence, structure and dynamic features to identify the residues that play a crucial role in mediating the allosteric signaling properties of the ATPase domain. Specifically, we identify the residues involved in the shortest-path communications of the domain modeled as a network of nodes (residues) and links (equilibrium interactions). By comparing the calculations on both closed and open conformation of Hsp70 NBD, we identified a subset of central residues located at the interface between the two lobes of the NBD near the nucleotide binding site, which form a putative communication pathway invariant to structural changes. Two pairs of residues forming contacts at the interface in the closed conformation of the NBD are observed to no longer interact in the open conformation, suggesting that these specific interactions may play a switch role in establishing the transition of the NBD between the two functional forms. Sequence co-evolution analysis and collective dynamics analysis with elastic network model further confirm the key roles of these residues in Hsp70 NBD dynamics and functions.

1. Introduction

Heat shock proteins (HSPs), also known as molecular chaperones, are ATP-regulated machines that perform several housekeeping activities in the cell: they assist in folding newly synthesized peptides, or unfolding and refolding partially folded or misfolded proteins, they regulate the intracellular trafficking of proteins, facilitate, in particular the recognition of those to be degraded by the proteasome, and most importantly, assist in the correct folding, and prevent the aggregation, of the proteins denatured in response to heat and other environmental stresses [1,2].

Hsp70 is one of the most ubiquitous members of the HSP family. It exists in almost all organisms [3]. It is composed of two domains: the ATPase domain, also referred as nucleotide binding domain, NBD [4], and the substrate binding domain (SBD) [5]. The two domains regulate the activity of each other via allosteric communication: ATP hydrolysis at the NBD increases the substrate binding affinity of the SBD, thus lowering the substrate exchange rate of the latter; on the other hand, the replacement of the ADP produced upon ATP hydrolysis by a new ATP (*nucleotide exchange*) lowers the binding affinity of the SBD thus enhancing the release and exchange of substrates [3].

The regulation of substrate binding affinity during the ATPase cycle is a crucial aspect of the chaperone activity of Hsp70, and notably, of other HSP family members [6]. The ATPase domain undergoes conformational changes between open and closed forms during the ATPase cycle, which correspond to different nucleotide binding states. The open conformation has been observed in the presence of ATP [7,8] and in the complexes formed with a class of co-chaperones called nucleotide exchange factors (NEFs) [9-11]. NEFs are known to assist nucleotide exchange by stabilizing the open form. Nucleotide exchange efficiency is viewed to be largely dependent on the conformational change to an open state.

In the present study, we examined the type of conformational changes occurring in the ATPase domain, and their influence on inter-residue communication pathways. Our previous examination of another ATP-regulated allosteric machine, the bacterial chaperonin GroEL, showed that the structure has access to intrinsically favored collective dynamics, on the one hand, and to well-defined signal transduction pathways that transmit allosteric effects away from the ATP binding site, on the other [12]. Redistribution of on-pathway interactions during the most cooperative (global) modes of motion of the chaperonin has been proposed to be a mechanism of allosteric regulation [13]. Toward gaining insights into the dynamic aspects of allosteric regulation, this time in the Hsp70 ATPase domain, we adopted here a multi-pronged approach: First, we identified a number of key residues distinguished by their central role in so far as the allosteric signal transduction across the molecule is concerned. This approach takes account of all atom-atom contacts, which are mapped into a low resolution, residue-level

representation of internal interactions. A number of residues lining the cleft between the two lobes of the NBD appear to modulate the opening and closing of the cleft. Second, we analyzed the sequence conservation and co-evolution patterns of these residues. Third, we examined their collective dynamics using a simple elastic network model, the Gaussian network model (GNM) [14,15].

Notably, the presently identified "central" residues belong to two groups in terms of their evolutionary patterns. Residues in the first group are highly conserved across Hsp70 NBD sequence homologs. These residues exhibit little, if any, movement in the global modes predicted by the GNM, serving as hinge centers near the nucleotide binding site. The second group comprises co-evolving residue pairs. These residue pairs tend to concertedly make and break contacts upon closing or opening of the nucleotide-binding cleft. Our results indicate that (i) the GNM-predicted global modes of the Hsp70 ATPase domain entail alterations in inter-residue contact topology, which in turn facilitate nucleotide binding or release; (ii) conserved residues participate in hinge centers in the global modes thus playing a role in maintaining the native contact topology; whereas sequentially variable but correlated residues exhibit a moderate mobility essential to enable functional changes in conformation.

2. Materials and Methods

2.1. *Structural Data*

The Hsp70 ATPase domain consists of two lobes [4]. Lobe I consists of subdomains IA and IB, and lobe II, of subdomains IIA and IIB (Figure 1a). We used the structures of the bovine homolog of Hsp70 (Hsc70) (PDB id: 1hpm [16]) for the closed form of the NBD. For the open form, we considered two structures of the same species complexed with mammalian NEFs: a complex with BAG, and another with Sse1, with respective PDB identifiers of 1hx1 [9] and 3c7n [17]. The structural alignment in Figure 1(b) shows that there is a global change in the relative positions of subdomains IB and IIB, as the structure undergoes a conformational change between closed and open forms.

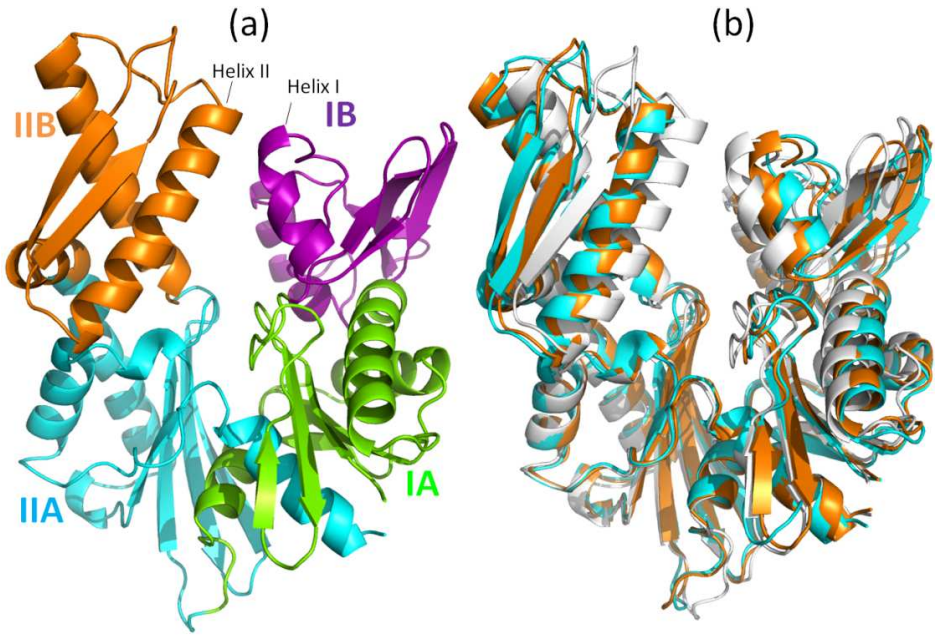


Figure 1. (a) Subdomains of Hsp70 ATPase domain: IA (green; residues 1-39 and 116-188), IB (purple; residues 40-115), IIA (cyan; residues 189-228 and 307-C-terminus) and IIB (orange; residues 229-306). (b) Ribbon diagram of the superimposed closed and open conformations. The closed form (white) is a structure observed in the absence of nucleotide (PDB id: 1hpm) [16]. Two open forms are shown, both observed in the complexes formed with NEFs: In cyan is the structure from the complex with the Sse1 (PDB id: 3c7n) [17]; and in orange is that assumed when complexed with BAG (PDB id: 1hx1) [9]. The NEFs (Sse1 and BAG) are not shown here. The three structures have been aligned using the Kabsch algorithm ([21] as implemented in PyMol).

2.2. *Sequence Data*

We started from the multiple sequence alignment (MSA) of 4839 sequences retrieved from the PFAM database [18] for Hsp70 family (PFAM id: PF00012). This family includes a wide range of subfamilies, some of which have biological functions not represented by the canonical Hsp70 (e.g., the Hsp110 subfamily). We refined the MSA using the consensus sequence of the ATPase domain (380 residues) in the bovine cytosolic homolog of Hsp70. The refinement consists of the following steps: (i) iterative implementation of the Smith-Waterman algorithm (SW) for pairwise alignment [19] using the consensus sequence against each sequence in the MSA, and elimination of those below a threshold SW score (of 300) so as to retain the closest orthologs to the human (Hsc70) and bacterial (DnaK) chaperones, (ii) removal of the MSA columns that correspond to insertions with respect to the consensus sequence, thus restricting the number of amino acids to 380, and (iii) elimination of the sequences that contain more than 10 gaps. The refinement resulted in 1627 sequences, which has been subjected to sequence conservation and correlation analyses. Conserved residues were identified with the WEBLOGO web server [20], and correlated residues by mutual information (MI) analysis.

2.3. *Identification of Central Residues*

We adopted the approach proposed by Nussinov and coworkers [22] to identify the central residues of the ATPase domain. These are the residues that exhibit a high probability of participating in shortest-path communication when all such paths between all residue pairs are examined. The protein is modeled as a network to this aim, each node representing a residue. Nodes corresponding to residues A and B are connected if at least one atom of A is within 4.5 Å distance from an atom of B. In previous studies, such contact-based network models have been pointed out to exhibit properties of small-world networks [23,24]. We utilized as metric the characteristic path length (L) --- a global property of the network defined as the average shortest path length between all pairs of nodes. The shortest path between each pair of nodes is computed using the Dijkstra's algorithm. The centrality of residue k is measured by the difference $\Delta L_k = L_{\text{hyp}}(k) - L$ in characteristic path length with respect to the original network, obtained for the hypothetical network where node k and all connected edges have been removed. If there is a significant increase in the characteristic path length due to removal of residue k, then residue k is considered to be a "central residue" in establishing internode communication. Central residues are hypothesized to play a role in allosteric signal transduction. Indeed, Nussinov and coworkers [22] have applied the method to seven distinct families, to find results consistent with experimental data. It should be noted however, that the identification of central residues may be sensitive to missing residues. In addition, the test set of proteins used in previous work for establishing the centrality contains globular proteins exclusively. The applicability of the method to other proteins remains to be established.

2.4. *Gaussian Network Model (GNM)*

We used the Gaussian network model (GNM) [14,15] for analyzing the equilibrium dynamics of the ATPase domain. Details on the method can be found in our previous studies. In summary, the structure is modeled as an elastic network, the position of each node being identified by that of each α-carbon; and pairs of nodes (C_α) within a cutoff distance of 7.3 Å are connected by elastic springs to account for the tendency of the structure to maintain its inter-residue contact topology under native state conditions. Knowledge of inter-residue contact distribution permits us to construct the Kirchhoff matrix $\mathbf{\Gamma}$, also known as Laplacian in graph theory. Eigenvalue decomposition of $\mathbf{\Gamma}$, or its inverse, $\mathbf{C} = \mathbf{\Gamma}^{-1}$,

$$\mathbf{C} = \sum_k \lambda_k^{-1} \mathbf{u}_k \mathbf{u}_k^{\mathrm{T}} = \sum_k \lambda_k^{-1} \mathbf{C}_k \qquad (1)$$

yields information on the spectrum of collective modes (fluctuations) intrinsically accessible to the structure under equilibrium conditions. Here λ_k and $\mathbf{u}_k$ represent the kth eigenvalue and eigenvector, respectively of $\mathbf{\Gamma}$, the summation is performed over all N-1 nonzero eigenmodes, for a protein of N residues/nodes and $\lambda_k^{-1}\mathbf{C}_k$ is the contribution of the kth mode to $\mathbf{C}$. The diagonal elements of $\mathbf{C}_k$ represent the normalized square fluctuations of the N residues, also called the *mobility profile*, induced by mode k, and the off-diagonal elements scale with the cross-

correlations between residue fluctuations. Each mode contribution is weighted by the inverse eigenvalue (which is proportional to the square root of the mode frequency) such that the lowest frequency modes make the largest contribution to **C**. We focus here on the modes in the low frequency regime, as the major determinants of potentially functional movements, and examine the weighted average mobility profiles $\Sigma_k \lambda_k^{-1} \mathbf{C}_k / \Sigma_k \lambda_k^{-1}$ for $1 \leq k \leq 10$.

2.5. *Mutual Information (MI)*

We adopted the mutual information (MI) content as a measure of the degree of co-evolution between residue pairs [25,26]. Accordingly, each of the N columns in the MSA generated for a protein of N residues is considered as a discrete random variable X_i ($1 \leq i \leq N$) that takes on one of the 20 amino-acid types with some probability. The MI associated with the random variables X_i and Y_j corresponding to the ith and jth positions is defined as

$$I(X_i, Y_j) = \sum_{\mathbf{all}x_i} \sum_{\mathbf{all}x_j} P(X_i; Y_j) \log \frac{P(X_i; Y_j)}{P(X_i) P(Y_j)} \qquad (2)$$

where $P(X_i ; Y_j)$ is the joint probability of observing amino acid type X at position i, and Y at position j; $P(X_i)$ and $P(Y_j)$ are the marginal/singlet probabilities for amino acids of types X and Y at the two respective positions. Eq (2) permits us to evaluate the $N \times N$ MI matrix **I** with elements $I_{ij} = I(X_i , Y_j)$. I_{ij} varies in the range $[0, I_{\max}]$, with the lower and upper limits corresponding to uncorrelated and most correlated pairs of residues. The MI is a classical concept from information theory; however, like other methods based on sequence information, the performance of the MI metric also relies on the quality of MSA.

3. Results and Discussion

3.1. *Centrality Profile of Different Conformations of NBD*

We calculated the centrality profile of all the three structures, the NBD in the closed state and the two others, in the open state. The results are shown in Figure 2 for the unbound (panel (a) and Sse1-bound (panel (b)) of the ATPase domain. The centrality profile for the BAG-bound form exhibits patterns similar to those observed in panel (b). From the characteristic path length and the RMSD calculated from structural alignment, we infer that the lobes of the Sse1-bound NBD are further apart than the BAG-bound form.

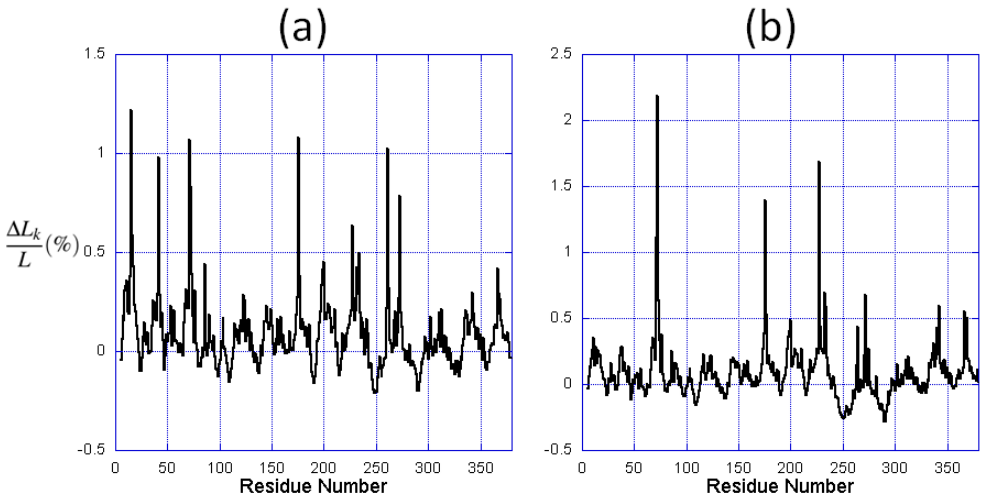


Figure 2. The centrality profile for NBD residues. The profiles are calculated for (a) the closed form, and (b) the open form in the Sse1-bound NBD. The abscissa represents the fractional change $\Delta L_k/L$ in characteristic path length compared to the original network, maxima referring to nodes that have a strong impact on communication efficiency, if removed.

Comparison of panels (a) and (b) shows that the two profiles exhibit similar features (i.e., peaks and minima at the same regions), while the relative heights of the peaks vary. In particular, the peaks near residues located at the inter-lobe interface, that is residues 257-276 (helix II) and residues 10-60, are suppressed in the open form. In contrast, some residues located at the nucleotide binding pocket (e.g., Arg342 and Asp366) are increased in the open form (note that the ordinate scales are different in the two panels). The increase in centrality suggests that they assume an enhanced role in establishing the communication away from the active site in the open form.

We consider the top ranking (top 2%, or equivalently, top 8 residues) residues in the centrality profile in each case and refer to them as the central residues in the following text. Among them four are distinguished as central residues in all of three structures, regardless of the open or closed state of the NBD: Lys71, Arg72, Glu175 and His227; in contrast, the other four residues vary with the conformation (Table 1).

Table 1. Central residues in the closed and two open conformations of Hsp70 NBD.

PDB id	**L(Å)**	**Binding NEF**	**conformation**	**Central residues(*)**	
				shared by all	*Specific to examined structures*
1hpm	14.06	None	closed	**Lys71 Arg72 Glu175** His227	Tyr15 Tyr41 Arg261 Arg272
1hx1	14.67	BAG	open		Ala60 Arg261 **Arg342** Asp366
3c7n	15.26	Sse1	open		Leu73 Asp232 **Lys271 Arg342**

(*) fully conserved residues are written in boldface (see Fig. 4a). Note that Leu73 and Asp232 are also highly conserved.

3.2. *Central Residues Have Different Structural and Sequence Variation Characters*

We examined the position of the central residues on the structure (Figure 3), and performed sequence analyses to examine their conservation profile and co-evolutionary properties (Figure 4).

The four residues that are invariant to conformational changes are colored cyan in Figure 3, and are labeled in Figure 3b. Interestingly, these (sequentially separated) residues appear to form a (spatially contiguous) communication path across the lobes, starting from His227 and ending at Glu175. Indeed, Lys71 and Glu175 serve as catalytic residues [27,28] and are believed to regulate a proline switch that regulates the inter-domain allosteric interactions. Studies by Johnson and McKay also show that mutations of Lys71 and Glu175 impede the functional conformational change of ATPase domain, which appears to block the signal transduction between subdomains IIB and IA [29]. His227 is located at the calcium binding site and its mutation significantly weakens the catalytic activity [30,31]. Since the central residues are found to be mediating allosteric communications in a variety of protein families [22], we propose these residues to play not only a catalytic role, but also a signaling role in communicating the nucleotide exchange events to the other regions of the NBD, including for example the interface with the substrate-binding domain.

These four residues are also highly conserved. The sequence logo [20] shown in Figure 4a indicates that residues Lys71, Arg72 and Glu175 are fully conserved. His227, although not conserved, can only be substituted by phenylalanine, although histidine probability is much higher, suggesting that a large aromatic group may be functional at this position. The interaction of Arg72 and His227, as can be seen from Figure 3e, can be viewed as a highly conserved amino-aromatic interaction [32], which is presumably maintained when histidine is replaced by phenylalanine. So even though His227 tolerates a mutation to phenylalanine, its interaction with Arg72 is conserved. In the following text we will refer to these 4 residues as the shared central residues (SCR).

The other central residues also exhibit patterns relevant to functional changes in NBD conformation. In the closed form, these residues (Tyr15, Tyr41, Arg261 and Arg272) are distributed along the cleft formed by lobes I and II to form two closely interacting pairs: Arg272---Tyr15 and Arg261---Tyr41. These pairs serve as two bridges that connect subdomain IIB with IA (Arg272---Tyr15) and with IB (Arg261---Tyr41). Bukau and coworkers [33] have shown that the salt bridges formed between helix I and helix II, labeled in Figure 1a, affect the nucleotide exchange of NBD. We speculate that among the residues located on these two helices, these two pairs arginines and tyrosines, also involved in amino-aromatic interactions, play a key role in controlling the subdomain closure and opening that in turn ensure efficient nucleotide stabilization or release, respectively. Moreover, since the central residues are supposed to be the most “indispensable” residues in establishing the shortest-path communications, the two pairs we

identified might be the “anchors” that maintain the closed conformation of NBD. Indeed, this conjecture is reinforced by the collective dynamics of the NBD in the next section.

Interestingly, residues at these four positions also tend to co-evolve as may be seen from the MI map in Figure 4b. By examining the sequence logo (Figure 4a), we found that the variation of amino acids at these residues primarily arises from the difference between the Hsp70 mammalian homolog Hsc70 and the Hsp70 bacterial homolog Dnak. The interactions between the two lobes of DnaK, as well as its interaction with NEF (GrpE in this case) primarily consist of hydrophobic contacts; whereas in Hsc70, there is a prominence of electrostatic interactions. The co-evolution of these central residues is in line with the specificity of their interactions in different organisms.

In the BAG-bound NBD, which assumes a less open conformation between the two NEF-bound structures, there still remains a contacting residue pair between the tips of subdomains IB and IIB (Arg261---Ala60, see Figure 3c), but this interaction can hardly account for the interface between the lobes. On the other hand, Arg342 and Asp366 are both conserved and form the nucleotide binding pocket. Their interactions are crucial for maintaining the conformation of the active site. In the Sse1-bound NBD, because subdomains IB and IIB have undergone a rotation, Asp232 becomes in contact with Lys227, which implies a putative extension of the SCR to subdomain IIB. Similarly, Leu73 extends SCR to subdomain IB. Lys271 and Arg342 are both conserved residues at the active site, and mutagenesis study showed that Arg342 is crucial for sulfogalactolipid recognition [34].

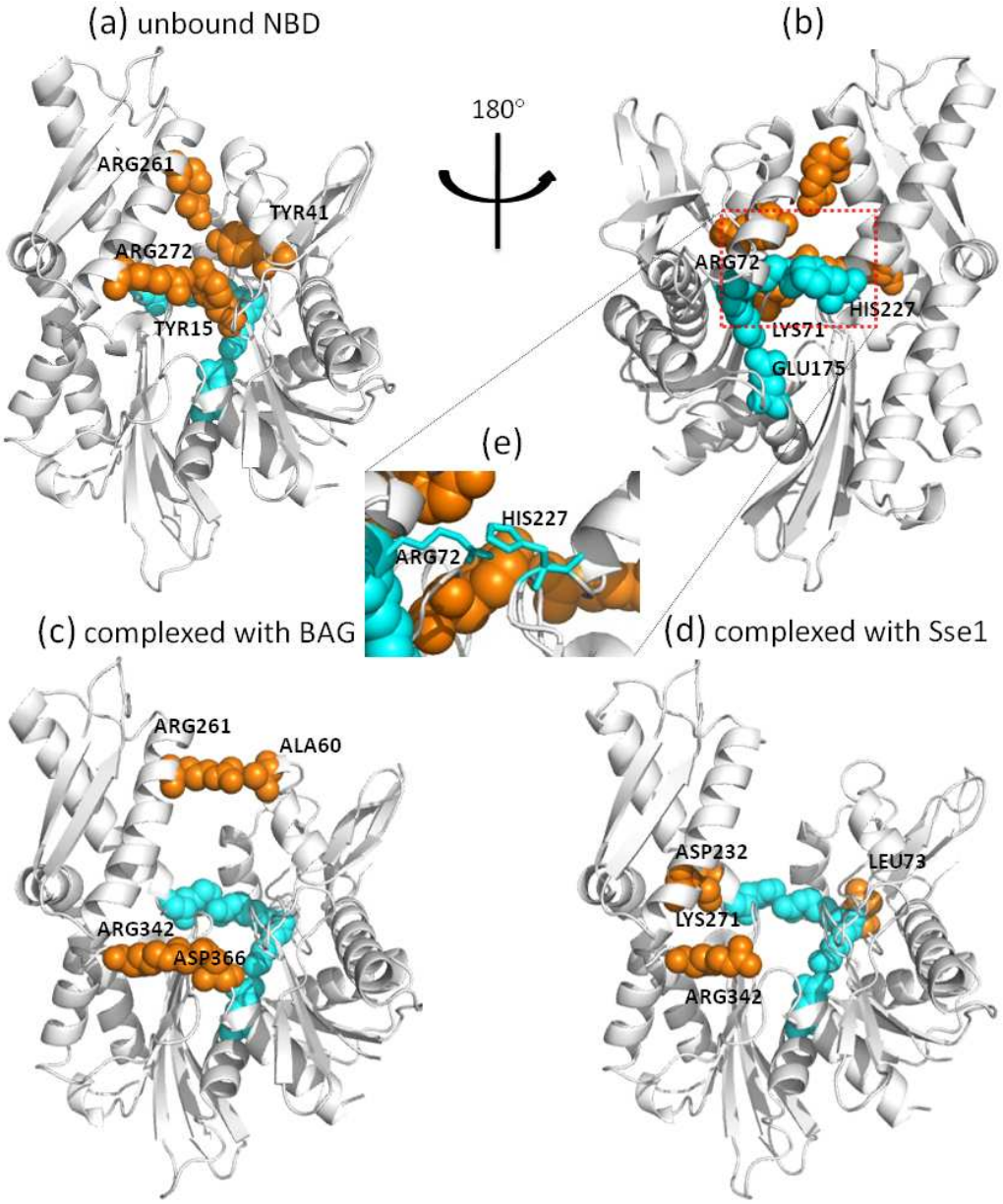


Figure 3. Position of Hsp70 NBD central residues. Panels (a) and (b) display the positions on the NBD closed form, (c) on the NBD open form bound to BAG and (d) on NBD open form bound to Sse1. Panel (e) highlights the interaction of Arg72 and His227. The conserved central residues are colored cyan; other central residues are colored orange. Panel (b) is the rotated view of panel (a), and the conserved central residues are only labeled in panel (b). The diagram was generated with PyMOL. (http://pymol.sourceforge.net).

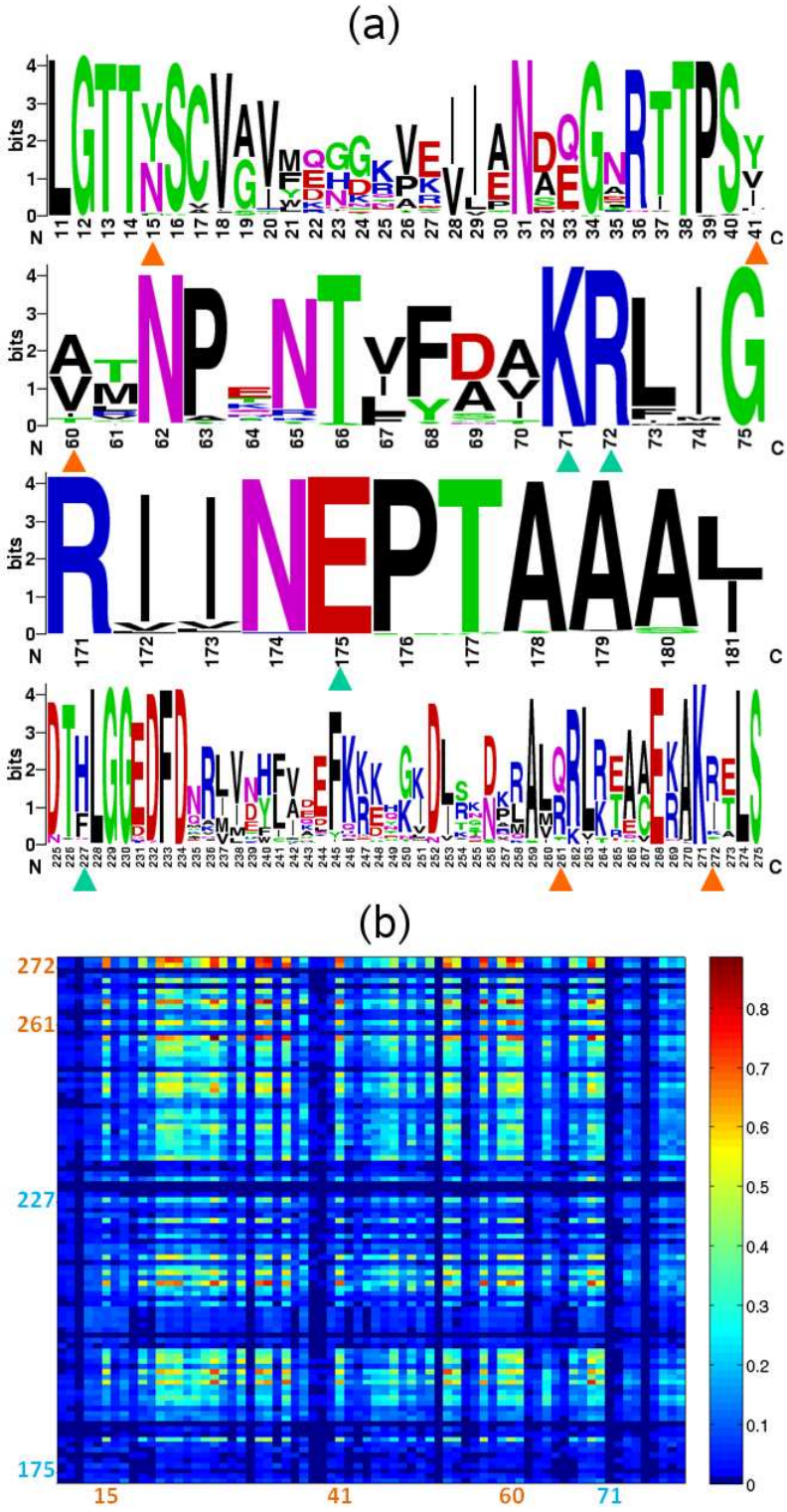


Figure 4. Sequence analysis of central residues. (a) Sequence logo of the SCRs (marked with cyan triangles) and other central residues (marked with orange triangles) identified in the closed and open (BAG-complexed) ATPase domain. (b) Mutual information map between residues 173-274 and 10-80, which includes the central residues located at the lobe interface in the closed state. The SCRs' residue numbers are written in cyan, whereas others are written in orange.

3.3. *Global Dynamics of ATPase Domain and Role of Central Residues*

As suggested in [22], the central residues generally relate to the system fragility; that is, these residues ought to remain "stable" to maintain the biological function of the molecule. From the sequence perspective, this requires sequence conservation; from the structural dynamics perspective, one might expect to see little variations, if any, in their spatial positions. In order to critically examine their dynamical characters, we examined the equilibrium dynamics of the ATPase domain using the GNM. We focused in particular on the low frequency end of the spectrum of modes, given that these modes are usually highly cooperative and relevant to function [35,36]. We compared the centrality profile and the mobility profile resulting from the weighted average of the 10 slowest modes of the closed-form NBD (Figure 5). Strikingly, the mobility profile (which represents the normalized distribution of square fluctuations in residue positions driven by these modes) exhibits minima at the peaks of the centrality profile, and vice versa. Minima in the mobility profile represent sites that act as hinges (or anchors) in the collective modes. Notably, all the central residues coincide with minima (Figure 5a), which is indicative of their key role in the global motions of the NBD. Arg261 and Arg272 are of particular interest: first, their mobility is higher than that of other central residues, suggesting a lower energy barrier for them to dissociate from lobe I to facilitate the cleft opening; second, helix II as the linkage between two most mobile regions of NBD, is implicated in functional motions.

Overall, the centrality profile and the slow modes curve are negatively correlated, which can be observed in Figure 5b. Figure 5a indicates the correspondence between the peaks of one curve and the valleys of the other, in most cases. In Figure 5b, the residues with high centrality (≥ 0.05) are characterized with low mobility, except for Asp86 (labeled in italic in Figure 5b). Indeed, Asp86 is located in an exposed helix that accounts for the rotation of subdomain IB and forms a salt bridge with Arg72, which in turn is one of the shared central residues presently identified. It appears that the salt bridge between Asp86 and Arg72 is critical to the motion of the exposed helix. On the other hand, the residues with negative centrality are usually located at the ends or tips of the structure, consistent with their high mobility.

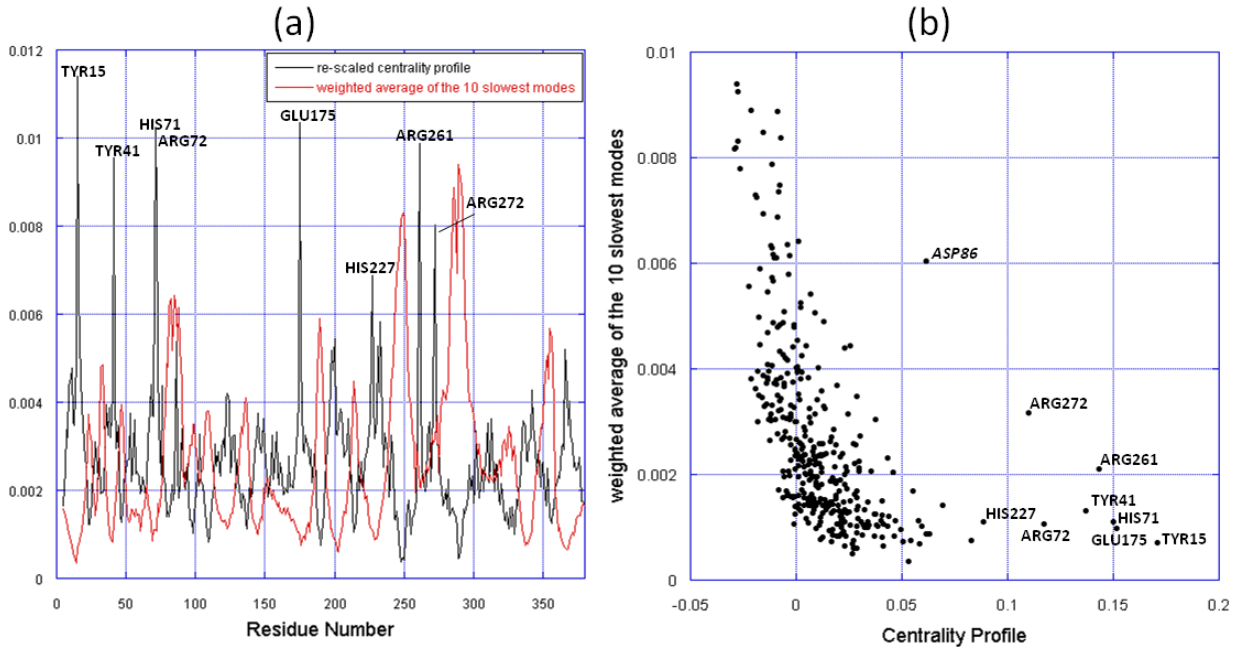


Figure 5. Comparison of the slowest modes and the centrality profile. (a) Mobility profile resulting from the weighted average of the 10 slowest modes and the re-scaled centrality profile of the NBD closed form. The centrality profile is re-scaled for visual comparison. The peaks corresponding to central residues are labeled, in addition to another outlier, Asp86, distinguished by its high mobility. (b) Mobility versus centrality for all residues in the NBD closed form. The points corresponding to central residues are labeled. The ordinate and abscissa values are taken from the two curves in panel a, for each residue.

3.4. *Summary of Different Types of Central Residues*

Here is an overview of the central residues based on our findings in this study (illustrated in Figure 6), and it remains to be seen how these results apply to central residues in other families of proteins. We can group the central residues into three categories depending on their location on the structure and/or their role in the structural dynamics:

1. The hinge point
2. bridging point at the interface near the cleft (or contact interface)
3. stretched linker (or long helix/loop stretching out)

In the first case, the central residues (e.g., SCR) connect two parts, at least one of which is highly mobile. These residues mediate the communications between different parts of the molecule, and transmit the information necessary for the proper functioning of the molecule. Perturbations at these residues are most likely to impede function. These residues are also highly conserved and serve as hinge points not only with respect to the two structural elements that are directly connected, but in the global dynamics of the entire NBD. In the second case, the central residues serve as linkages at the interface between substructures that have intrinsically access to alternative (e.g. open and closed) conformations. They act as the "anchoring point" of the interface, and can be the determinants of the motions of the moving parts. These residues are more exposed to the environment and more tolerant to mutations compared to the first case. Yet, their important role is signaled by correlated mutations that take place which presumably aim at restoring the key role (that of locking the closed form in this case). For residues in the third category, although we did not observe any such residue in this study, they have been observed in other systems. For example, the inter-domain linker between the NBD and SBD of the Hsp70 possess such residues, which evidently play a key role in establishing the allosteric communication between the two domains [37].

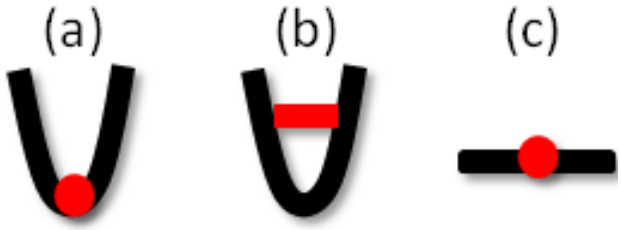


Figure 6. Three scenarios for the central residue's location on the structure.

3.5. *Comparison with results from other methods.*

We note that similar studies have been conducted for identifying allosteric residues and communication pathways, including the work of Tang and coworkers [38] who developed the AlloPathFinder package, the statistical coupling analysis (SCA) of Ranganathan and coworkers [39], or the pair-to-pair correlations analysis of Eyal *et al.* [40]. The performance of different methods for detecting sequence correlations has been comparatively studied by Fodor and Aldrich [41] and by Eyal *et al*. [42]. Calculations performed here for Hsp70 with AlloPathFinder indicated a number of pathways starting from His227 to Lys71/Glu175 which are composed of a series of residues deeply buried at the interface of subdomains IA and IIA (e.g, His227→Leu228→Leu200→Val337→Ala179→Glu175→Lys71). When the destination was set to Arg72, however, AlloPathFinder identified the pathway established by the contact between His227 and Arg72. The difference in the identified pathway(s) may be attributed to the fact that AlloPathFinder employs evolutionary information for weighting the individual steps along the communication paths, whereas our present approach for identifying pathways is exclusively based on topology. On the other hand, the application of SCA on Hsp70 indicates that Lys71 and Arg72 take part in a given cluster of highly coupled residues, while Glu175 belongs to another cluster, and His227does not belong to any such cluster (private communication with Dr. Lila Gierasch).

4. Conclusion

We presented here a computational study of the ATPase domain of Hsp70 to identify and analyze the residues that are crucial for efficient transduction of signals within this domain. This particular domain serves as a signal transduction module not only in molecular chaperones but many other proteins, as well, and understanding the position and role of key residues in establishing allosteric communication is a topic of broad interest. We identified a subset of central residues across sub-domains, which form a communication pathway invariant to structural change between open and closed forms. We also identified two pairs of interacting residues bridging the lobes in the closed conformation but no longer interacting in the open conformation.

The analysis of sequence correlations and collective dynamics assisted in assessing the key role of these residues in mediating domain movements relevant to function, supporting the functional character of central residues in allosteric systems..The findings independently obtained by centrality profile based on graph-theoretical methods, GNM based on statistical mechanical principles, and sequence analysis provide complementary perspectives on the allosteric potential, intrinsic dynamics and sequence evolution properties, respectively, of the examined system. These three pieces of data have been advantageously combined here to extract a uniform picture of the structural and dynamic aspects of Hsp70 NBD function. Further investigation of the detailed mechanism of transition between the NBD closed and open conformations and its coupling to the SBD, using the adaptive ENM [43], holds promise toward gaining deeper insights about, and further establishment of, the functional significance of particular residues and interactions inferred from the present study.

References

1. F. U. Hartl and M. Hayer-Hartl, *Science* **295**, 1852 (2002).
2. F. U. Hartl and M. Hayer-Hartl, *Nat Struct Mol Biol.* **16**, 574 (2009).
3. B. Bukau and A. L. Horwich, *Cell* **92**, 351 (1998).
4. K. M. Flaherty, C. Luca-Flaherty, and D. B. McKay, *Nature* **346**, 623 (1990).
5. X. T. Zhu, X. Zhao, W. F. Burkholder, A. Gragerov, C. M. Ogata, M. E. Gottesman, and W. A. Hendrickson, *Science* **272**, 1606 (1996).
6. M. M. Ali, S. M. Roe, C. K. Vaughan, P. Meyer, B. Panaretou, P. W. Piper, C. Prodromou, and L. H. Pearl, *Nature* **440**, 1013 (2006).
7. A. Bhattacharya, A. V. Kurochkin, G. N. Yip, Y. Zhang, E. B. Bertelsen, and E. R. Zuiderweg, *J. Mol. Biol*. **388**, 475 (2009).
8. E. B. Bertelsen, L. Chang, J. E. Gestwicki, and E. R. Zuiderweg, *Proc. Natl. Acad. Sci. U. S. A.* **106**, 8471 (2009).
9. H. Sondermann, C. Scheufler, C. Schneider, J. Hohfeld, F. U. Hartl, and I. Moarefi, *Science* **291**, 1553 (2001).
10. S. Polier, Z. Dragovic, F. U. Hartl, and A. Bracher, *Cell* **133**, 1068 (2008).
11. C. J. Harrison, M. HayerHartl, M. DiLiberto, F. U. Hartl, and J. Kuriyan, *Science* **276**, 431 (1997).
12. C. Chennubhotla and I. Bahar, *Molecular Systems Biology* **2**, 36 (2006).
13. C. Chennubhotla, Z. Yang, and I. Bahar, *Mol. Biosyst.* **4**, 287 (2008).
14. L. W. Yang, A. J. Rader, X. Liu, C. J. Jursa, S. C. Chen, H. A. Karimi, and I. Bahar, *Nucleic Acids Research* **34**, W24 (2006).
15. I. Bahar, A. R. Atilgan, and B. Erman, *Folding & Design* **2**, 173 (1997).
16. S. M. Wilbanks and D. B. McKay, *J. Biol. Chem.* **270**, 2251 (1995).
17. Schuermann JP, Jiang J, Cuellar J, Llorca O, Wang L, Gimenez LE, Jin S, Taylor AB, Demeler B, Morano KA, Hart PJ, Valpuesta JM, Lafer EM, and Sousa R, *Molecular Cell* **31**, 232 (2008).
18. R. D. Finn, J. Mistry, B. Schuster-Bockler, S. Griffiths-Jones, V. Hollich, T. Lassmann, S. Moxon, M. Marshall, A. Khanna, R. Durbin, S. R. Eddy, E. L. L. Sonnhammer, and A. Bateman, *Nucleic Acids Research* **34**, D247 (2006).
19. T. F. Smith and M. S. Waterman, *J. Mol. Biol*. **147**, 195 (1981).
20. G. E. Crooks, G. Hon, J. M. Chandonia, and S. E. Brenner, *Genome Res.* **14**, 1188 (2004).
21. W. Kabsch, *Acta Crystallographica Section A.* **32**, 922 (1976).
22. A. del Sol, H. Fujihashi, D. Amoros, and R. Nussinov, *Molecular Systems Biology* **2**, 2006.0019 (2006).

23. D. J. Watts and S. H. Strogatz, *Nature* **393**, 440 (1998).
24. L. H. Greene and V. A. Higman, *J. Mol. Biol.* **334**, 781 (2003).
25. Y. Liu, E. Eyal, and I. Bahar, *Bioinformatics* **24**, 1243 (2008).
26. N. D. Clarke, *Protein Science* **4**, 2269 (1995).
27. M. C. O'Brien, K. M. Flaherty, and D. B. McKay, *J. Biol. Chem.* **271**, 15874 (1996).
28. M. Vogel, B. Bukau, and M. P. Mayer, *Molecular Cell* **21**, 359 (2006).
29. E. R. Johnson and D. B. McKay, *Biochemistry* **38**, 10823 (1999).
30. M. Sriram, J. Osipiuk, B. Freeman, R. Morimoto, and A. Joachimiak, *Structure* **5**, 403 (1997).
31. X. Wu, M. Yano, H. Washida, and H. Kido, *Biochem. J.* **378**, 793 (2004).
32. S. K. Burley and G. A. Petsko, *FEBS Lett.* **203**, 139 (1986).
33. D. Brehmer, S. Rudiger, C. S. Gassler, D. Klostermeier, L. Packschies, J. Reinstein, M. P. Mayer, and B. Bukau, *Nature Structural Biology* **8**, 427 (2001).
34. D. Mamelak and C. Lingwood, *J. Biol. Chem.* **276**, 449 (2001).
35. I. Bahar, A. R. Atilgan, M. C. Demirel, and B. Erman, *Physical Review Letters* **80**, 2733 (1998).
36. I. Bahar and A. J. Rader, *Curr. Opin. Struct. Biol.* **15**, 586 (2005).
37. J. F. Swain, G. Dinler, R. Sivendran, D. L. Montgomery, M. Stotz, and L. M. Gierasch, *Molecular Cell* **26**, 27 (2007).
38. S. Tang, J. C. Liao, A. R. Dunn, R. B. Altman, J. A. Spudich, and J. P. Schmidt, *J. Mol. Biol.* **373**, 1361 (2007).
39. G. M. Suel, S. W. Lockless, M. A. Wall, and R. Ranganathan, *Nature Structural Biology* **10**, 59 (2003).
40. E. Eyal, S. Pietrokovski, and I. Bahar, *Bioinformatics* **23**, 1837 (2007).
41. A. A. Fodor and R. W. Aldrich, *J. Biol. Chem.* **279**, 19046 (2004).
42. E. Eyal, M. Frenkel-Morgenstern, V. Sobolev, and S. Pietrokovski, *Proteins* **67**, 142 (2007).
43. Z. Yang, P. Majek, and I. Bahar, *PLoS Comput. Biol.* **5**, e1000360 (2009).

3D-BLAST: 3D PROTEIN STRUCTURE ALIGNMENT, COMPARISON, AND CLASSIFICATION USING SPHERICAL POLAR FOURIER CORRELATIONS

LAZAROS MAVRIDIS

DAVID W. RITCHIE

INRIA Nancy – Grand Est, LORIA, 615 rue du Jardin Botanique
54506 Vandoeuvre-lès-Nancy, France

This paper presents a novel sequence-independent method of aligning protein structures using three-dimensional spherical polar Fourier (SPF) representations of protein shape. The approach is demonstrated by clustering subsets of the CATH database for each of the four main CATH fold types, and by searching the entire CATH database of some 12,000 structures using several protein structures as queries. Overall, the automatic SPF clustering approach agrees very well with the expert-curated CATH classification, and ROC-plot analyses of the database searches show that the approach has very high precision and recall. Database query times can be reduced considerably by using a simple rotationally-invariant pre-filter in tandem with a more sensitive rotational search with little or no reduction in accuracy. Hence it should soon be possible to perform on-line 3D structural searches in interactive time-scales.

1. Introduction

The BLAST and FASTA sequence alignment programs are probably considered by most biologists as the standard tools for searching genomic nucleotide or amino acid sequence databases. However, it is known that in nature protein structure is more conserved than protein sequence. Hence, structural alignments can provide significant insights about protein function and can help classify protein families into functional super-families [1]. Considering the large and rapidly growing number of three-dimensional (3D) protein structures available in the Protein Databank (PDB [2]), it is important to develop new methods to align and compare protein structures [3]. Current protein structure alignment methods such as SSM [4], CE [5], VAST [6], SSAP [10], and DALI [1] typically use graph-matching and dynamic programming techniques to identify and align cliques of backbone Cα atoms or secondary structural features. However, these approaches are slow compared to conventional sequence alignment methods. Furthermore, finding the best way to perform 3D structural alignments remains an open question [7]. The most widely used protein structure classifications are the CATH [8] and SCOP [9] databases, both of which are curated by human experts. In CATH, the classification is initially performed using SSAP, whereas SCOP relies more on visual inspection by the curators. In both cases, it would be desirable to be able to assemble and update structural classifications in a more automated way.

In a significant step towards performing protein structure comparisons more efficiently, Mak *et al.* [11] and Sael *et al.* [12] showed that the 3D shapes of large protein molecules could be compared and classified very rapidly using special 3D pose-invariant descriptors derived from spherical harmonic (SH) and Zernike polynomials [13]. Hence this kind of approach offers the possibility of being able to search a 3D database of protein structures very rapidly. Similarly, it has been shown previously that the related spherical polar Fourier (SPF) representation provides a fast way to perform protein-protein docking correlations [14,15]. However, until now, the SPF approach has not been used to superpose and compare protein shapes. Conceptually, the use of Zernike descriptors has close parallels with the SPF representation, although Mak *et al*. and Sael *et al*. did not exploit the special rotational properties of the SH basis functions. Hence their approaches can detect similar protein shapes, but cannot superpose them. There is growing interest in the use of SH-based shape representation techniques [16-25]. The novelty of our SPF representation compared to other SH-based approaches is its use of orthonormal Laguerre-Gaussian (GL) radial functions to give a 3D density-based representation of molecular shape. This removes the requirement that molecular surfaces should be star-like with respect to the chosen coordinate origin [26], and it allows both rotational an translational correlation expressions to be calculated analytically [27]. In this paper, we demonstrate that SPF correlations may be used to superpose and compare protein structures in an efficient and completely sequence-independent manner. We present results obtained by clustering multiple protein structures selected from the CATH

database, and we demonstrate the utility of our approach by performing queries of single protein structures against the entire CATH database of some 12,000 protein structures.

2. Methods

2.1. *Spherical Polar Fourier Shape Density Functions*

In the SPF approach, protein shapes are represented as 3D density functions expressed as expansions of orthonormal basis functions:

$$R(\mathbf{r}) = \sum_{n=1}^{N} \sum_{l=0}^{n-1} \sum_{m=-l}^{l} a_{nlm} R_{nl}(r) y_{lm}(\vartheta, \varphi) \qquad (1)$$

where N is the order of the expansion, $R_{nl}(r)$ are Laguerre-Gaussian radial functions, $y_{lm}(\vartheta, \varphi)$ are real spherical harmonics, and a_{nlm} are the expansion coefficients which are calculated numerically as described previously [14]. Figure 1 shows the SPF representations of a pair of similar nitrogenase domains at several expansion orders.

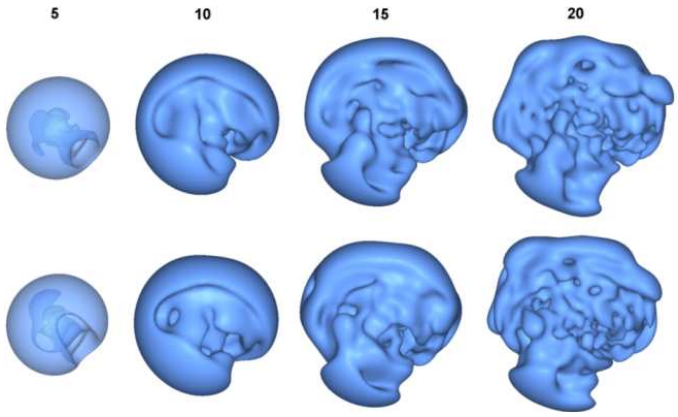


Figure 1. Two similar nitrogenase proteins, represented as SPF expansions at expansion orders N=5, 10, 15, and 20. The protein in the top row is from azotobacter vinlandii (PDB code 2MIN). The protein in the bottom row is from clostridium pasteurianum (PDB code 1MIO).

In order to superpose a pair of protein structures we calculate a rotation-dependent Carbo-like similarity score S_{ROT} using:

$$S_{ROT} = \frac{\sum_{nlm}^{N} a_{nlm} b_{nlm}}{\left(\sum_{nlm}^{N} a^2{}_{nlm} \right)^{1/2} \left(\sum_{nlm}^{N} b^2{}_{nlm} \right)^{1/2}} \qquad (2)$$

Conceptually, one protein is held fixed and a six-dimensional (6D) rotational/translational search over positions of the second protein is performed. However, in practice it is more efficient to implement the search using one translational and five Euler angle rotational coordinates [14].

2.2. *Rotationally Invariant Fingerprints*

Protein superpositions can be calculated relatively quickly by using FFT techniques to accelerate the 6D search [14]. However, it is necessary to develop even faster comparison techniques in order to search very large 3D structural databases. Noting that expansion coefficients with the same values of m transform amongst themselves under rotation, it is natural to use the vector interpretation of SH coefficients to construct rotationally invariant (RI) fingerprints (RIFs) as:

$$A_{nl} = \sqrt{\sum_{m=-l}^{l} a_{nlm}^2} \qquad (3)$$

If the coefficients a_{nlm} define the shape density of a protein, then the rotation-invariant descriptors A_n together encode the protein's radial mass distribution. By analogy to Eq 2, the RIF similarity score is written as:

$$S_{RIF} = \frac{\sum_{n=1}^{N} A_n B_n}{\left(\sum_{n=1}^{N} A_n\right)^{1/2} \left(\sum_{n=1}^{N} B_n\right)^{1/2}} \qquad (4)$$

2.3. *Implementation*

Our approach has been implemented in a C program called 3D-Blast. We have used 3D-Blast to calculate and store SPF coefficients for all the proteins of the CATH database. However, the approach could equally be used to store SPF descriptions of other protein structure databases such as SCOP, for example. Given a protein query structure in PDB format, 3D-Blast will calculate its SPF representation and then use only the resulting SPF expansion coefficients to search its database.

2.4. *Data Preparation*

In order to evaluate the SPF approach, clustering experiments were performed on selected proteins from the CATH database [8,10], and the entire CATH database was searched using SPF density functions as queries. In CATH, proteins are assigned to a super-family according to their fold class, architecture, topology, and homology. This classification scheme is essentially hierarchical, with the top-level class consisting of four possible fold types: All-α, All-β, α+β, and irregular. Each of the four levels in the CATH hierarchy is identified by a numeric code. Additionally, CATH names each protein according to its four-letter PDB code, its chain letter, and the number of domains (e.g. 1IOMA02). These naming conventions are followed here. For each clustering experiment, five or six super-families with the same architecture were selected in such a way as to give around 30 protein structures for each CATH fold class. Hence the aim of these experiments is to assess how well our approach can identity proteins with the same topology and homology within a given fold architecture. The details of the super-families used here are shown in Table 1.

For the initial database search experiment, asparagine synthetase (PDB code 12AS, CATH super-family 3.30.930.10) was selected as the query structure. The 3.30.930.10 super-family has 27 members, and these were treated as "positives" while the remaining proteins in the database were treated as "negatives" with respect to the query. If a scoring function were to reproduce exactly the CATH classification, the 27 positives would appear at the top of the ranked list. However, such an ideal outcome is seldom observed in practice. Hence Receiver-Operator-Characteristic (ROC) curves [28] are used to analyse objectively the precision/recall characteristics of the scoring functions. In a ROC analysis, each element of the ranked list is considered in turn, and the number of positives and negatives in the sublists to each side of the current element are counted. Here, we call the high similarity sublist the "hit list". A true positive (TP) is assigned when an element in the hit list contains an original positive, and a false positive (FP) is assigned if that element contains a negative. Conversely, a true negative (TN) is assigned when an original negative falls outside the hit list, and a false negative (FN) is assigned if that position is occupied by a positive member. ROC plots are produced by plotting the true positive rate (TPR) on the y-axis against the false positive rate (FPR) on the x-axis, where TPR and FPR are given by:

$$TPR = \frac{TP}{TP + FN} \qquad (5)$$

$$FPR = \frac{FP}{FP + TN} \qquad (6)$$

The quality of a scoring function can quickly be assessed from the shape of a ROC plot. For example, a random scoring function would give a diagonal line (TPR=FPR), whereas a perfect scoring function would give a horizontal line (TPR=1) with a maximum value for the area under the curve (AUC=1).

Table 1. The 23 CATH super-families used in the four clustering experiments, grouped according to class and architecture. For each super-family the name of each representative protein is provided according to the CATH naming convention.

Class + Architecture	Topology + Homology	Protein Name and Function	Representative member
All-α Orthogonal Bundle (1.10)	230.10	Cytochrome p450-Terp; Domain 2	1iomA02
	120.10	Bifunctional Trypsin/Alpha-Amylase Inhibitor	1beaA00
	225.10	NK - Lysin	1l9lA00
	167.10	Regulator of G-Protein Signalling 4; Domain 2	1dk8A02
	30.10	DNA Binding (I) , subunit A	1qrvA00
All-β Ribbon (2.10)	109.10	Umud Fragment, subunit A	1jhfB00
	150.10	Urease, subunit B	1ejxB00
	110.10	Cysteine Knot Cytokines, subunit B	1g47A00
	77.10	Hemagglutinin; Chain A, domain 2	1jsdA02
	160.10	Vascular Endothelial Growth Factor - 165, Heparin - binding Domain	1kmxA00
	10.10	Seminal Fluid Protein PDC - 109 (Domain B)	1h8pA02
α+β Roll (3.10)	130.10	P-30 Protein	1dy5A00
	170.10	Elastate; Domain 1	1u4gA01
	150.10	DNA Polymerase III; Chain A, Domain 2	1ok7A01
	110.10	Ubiquitin Conjugating Enzyme	2grrA00
	120.10	Flavocytochrome B2; Chain A; Domain 1	1cyoA00
Irregular (4.10)	280.10	MYOD Basic-Helix-Loop-Helix Domain, subunit B	1nlwE00
	290.10	Bacteriorhodopsin Fragment	1bctA00
	410.10	Factor Xa Inhibitor	1g6xA00
	490.10	High-Potential Iron-Sulfur Protein; Chain A	1iuaA00
	400.10	Low-density Lipoprotein Receptor	2fcwB02
	320.10	Dihydrolipoamide Transferase	1w85I00

3. Results

3.1. *Expansion Resolution and Protein Superposition*

It was shown previously that SPF expansions of order $N \geq 25$ are required for protein-protein docking correlations [14]. However, from our previous results on small-molecule virtual screening [29], we expected that considerably lower order expansions would be sufficient to calculate satisfactory protein shape-density superpositions. In order to test this hypothesis, we selected four example protein pairs with sequence identities ranging from 28% to 43% previously identified by Levitt and Gerstein [30], and we superposed them using different expansion orders. The root mean square deviation (RMSD) for each superposed pair was calculated between corresponding Cα atoms identified by ProFit [31]. Figure 2 shows the RMSD as a function of the expansion order *N*.

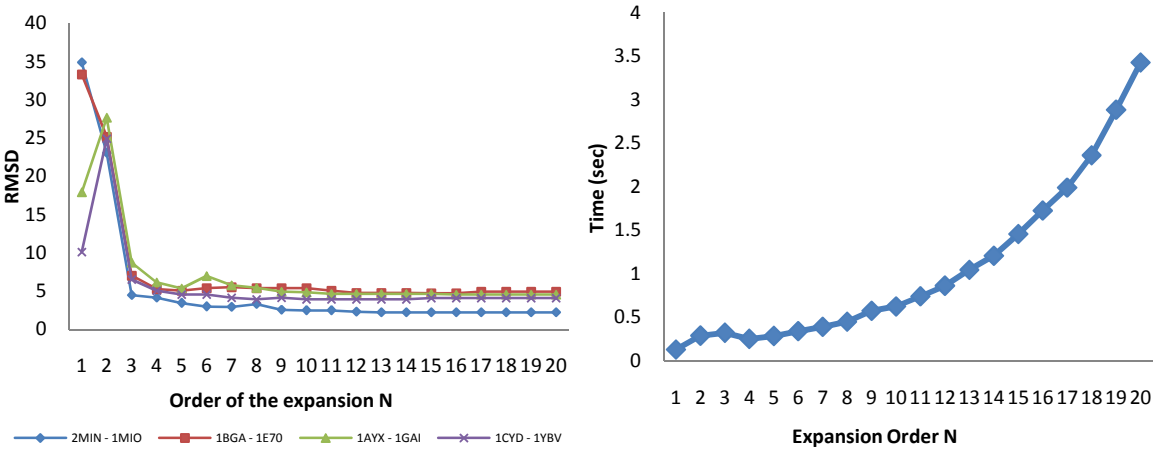


Figure 2. On the left, the RMSD values of four protein pairs are plotted against the expansion order *N*. On the right, the time per pair-wise search is plotted against the expansion order *N*.

Figure 2 shows that using an expansion order of $N \geq 3$ is generally sufficient to obtain good superpositions (with RMSDs ranging from 2Å to 5Å), requiring about 0.25 seconds per superposition. However, we choose to use expansions to order N=6 in order to disambiguate cases where flipping about an axis might give similar scores [29]. Figure 2 shows that calculating N=6 correlations is only marginally more expensive than using N=3. Figure 3 shows the *N=6* superposition of the nitrogenase proteins shown in Figure 1.

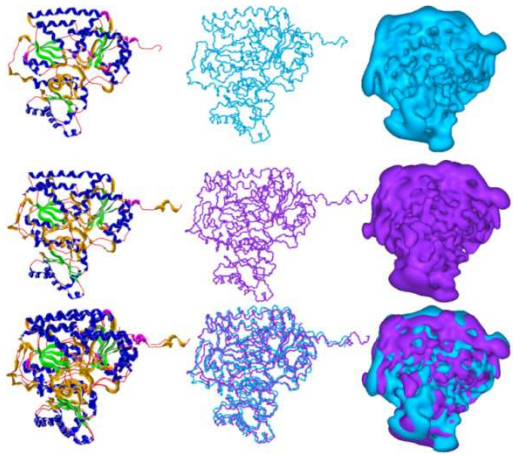


Figure 3. The superposition of a pair of nitrogenase proteins calculated using N=6 SPF correlations, shown as ribbon cartoons (left), backbone traces (middle), and as 3D SPF density expansions to order N=25 (right). The protein in the top row is from azotobacter vinlandii (PDB code 2MIN). Top row: PDB code 2MIN; middle row PDB code 1MIO; bottom row their superposed orientation. The two proteins have a sequence identity of 43%.

3.2. *Clustering Protein Structures*

For the All-α clustering experiment, five super-families were selected, as listed in Table 1. For each pair of proteins in this set, a correlation search was performed to find the orientation that gives the maximum Carbo similarity score (Eq 2). Ward's agglomerative clustering [32] was then applied to the resulting table of pair-wise similarity scores. The clustering results in Figure 4 show that the 1.10.230.10 and 1.10.167.10 super-families are correctly assigned to two separate groups. Although there exist some differences in the clusters produced for the remaining super-families, there is still a very good agreement between the calculated SPF clusters and the CATH hierarchy. The most notable exception is that suposin (PDB code 1N69) is grouped with the 1.10.30.10 super-family. From visual inspection of Figure 4 it can be seen that the overall fold of suposin is much closer to that of the 1.10.30.10 super-

family than the CATH assignment of 1.10.225.10. This suggests that the automatic SPF classification could potentially help the CATH curators resolve unusual or ambiguous cases.

For the All-β class, six super-families were selected. As can be seen in Figure 5, SPF clustering correctly distinguishes all six groups, but two proteins are misplaced according to the CATH classification. These are the carboxy-terminal LIM domain (PDB code 1CTL) and the influenza virus hemagglutinin (PDB code 2VIR) which are grouped with the singleton heparin-binding domain (PDB code 1KMX). This seems to occur because 1CTL and 2VIR are calculated to be less similar to the other members of their respective CATH super-families, and they are grouped with 1KMX largely because all three proteins have similar steric bulk.

For the α + β class, five super-families were selected. From these five super-families the 3.10.130.10 and 3.10.120.10 super-families are correctly assigned into two groups, as shown in Figure 6. The other three super-families (3.10.110.10, 3.10.150.10, and 3.10.170.10), present a similar case to the All-α results, whereby one super-family group (3.10.110.10) is split into two sub-groups and two super-family groups (3.10.170.10 and 3.10.150.10) are merged into one. Nonetheless, despite these differences, the overall consistency of the SPF clustering with the CATH hierarchy is clearly very good.

For the irregular class, six super-families were selected. As in the All-β example, SPF clustering is completely consistent with the CATH hierarchy. However, two proteins are misplaced with respect to the CATH classification, namely bikunin from the inter-alpha-inhibitor complex (PDB code 1BIK) and the tick anticoagulant peptide (PDB code 1D0D), which are both grouped with the 4.10.490.10 super-family. This seems to be due to the difference in size between those proteins and the rest of their super-family of factor XA inhibitors. For example, Figure 7 shows that bikunin has a repeat of the same motif as the other factor XA inhibitors. Hence, it is sterically too large to be clustered with the other XA inhibitors, and is instead placed with the larger proteins of the 4.10.490.10 super-family.

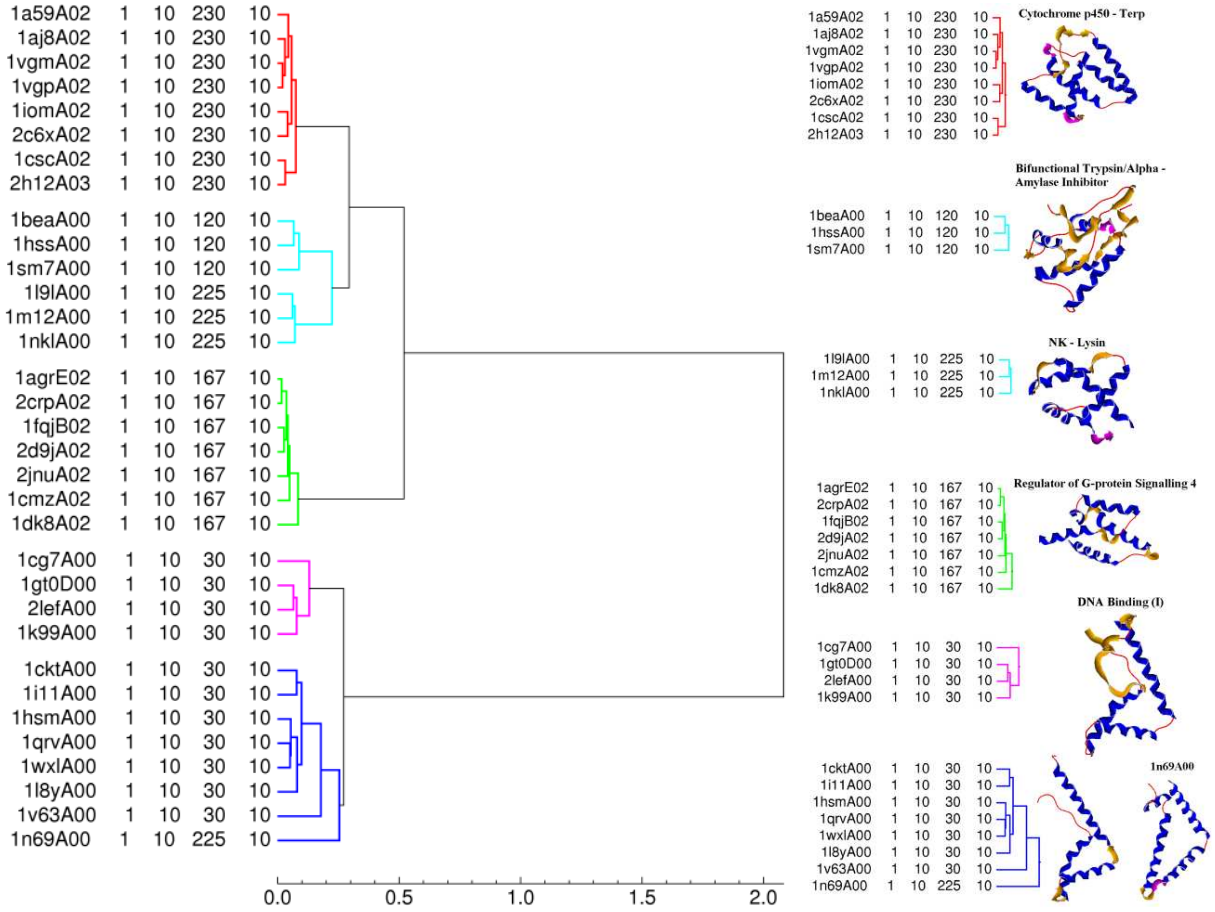


Figure 4. SPF clustering results of the All-α class. Left: the dendrogram obtained using *N=6* with five clusters; Right: the corresponding groups and the representative proteins for each group.

3.3. *Database Searching*

Because it is relatively time-consuming to calculate high order comparisons, different rotational and rotation-invariant parameters were tested to explore the extent to which the speed of queries on a large database can be improved while still performing accurate searches. In each case, the asparagine synthetase structure (PDB code 12AS, CATH super-family 3.30.930.10) was superposed onto each protein in the database using $N=6$ SPF correlation searches, and the database structures were ranked in order to similarity to the query. Figure 8 shows the resulting ROC curves obtained for a range of expansion orders when querying the CATH database, from which it can be seen that our approach gives very good precision and recall. Figure 9 shows the 27 members of this super-family, which were treated as true positives with respect to the query. To analyse the results further, the TPs were clustered into 5 groups. The query belongs to group 1 and when using an expansion order of $N \geq 6$, all members of this group were found in the top 10 hits. groups 2 and 3 have similar β-sheet structures to group 1, but different arrangements of α-helices. All proteins in groups 2 and 3 are ranked in the top 20% of the database. All proteins in group 4 are ranked in the top 30%. The singleton group 5 is an obvious outlier due to its extra α-helical domain.

Figure 8 also shows results for the RIF scoring function. Compared to the rotation-dependent scoring function, the RIF function generally performs remarkably well. However, the two functions behave rather differently on the first percentages of the database. For example, the rotational searches give a TPR of around 40% on the first 0.1% of the database, whereas the RIF searches give a TPR of only around 10%. Hence, the RIF function is not sufficiently sensitive to be used on its own but it could usefully be used as a fast pre-filter on the database so that the more expensive rotation-dependent function is only applied to the most promising candidates. In order to test the notion, the CATH database was searched using the RIF and rotational scoring functions in tandem using several protein structures as queries: asparagine synthetase, ALF4-activated Giα1 protein (PDB code 1AGR), chicken cysteine-rich protein (PDB code 1B8T), dihydrolipoyllysine-residue acetyl transferase (PDB code 1W4E), and UbcH7 (PDB code 1C4Z). Using a RIF pre-filter similarity threshold of 0.99, which selects from 2% to 15% of the database for rotational re-scoring, each tandem search takes less than 10 minutes compared to 75 minutes for full rotational searches on a 2.3GHz Pentium Xeon processor. Figure 10 shows the resulting ROC plots for the rotational, RIF, and the tandem searches. This figure shows that tandem searches achieve the same high level of performance as the rotational searches. The very high precision/recall values further support the utility of the SPF scoring functions.

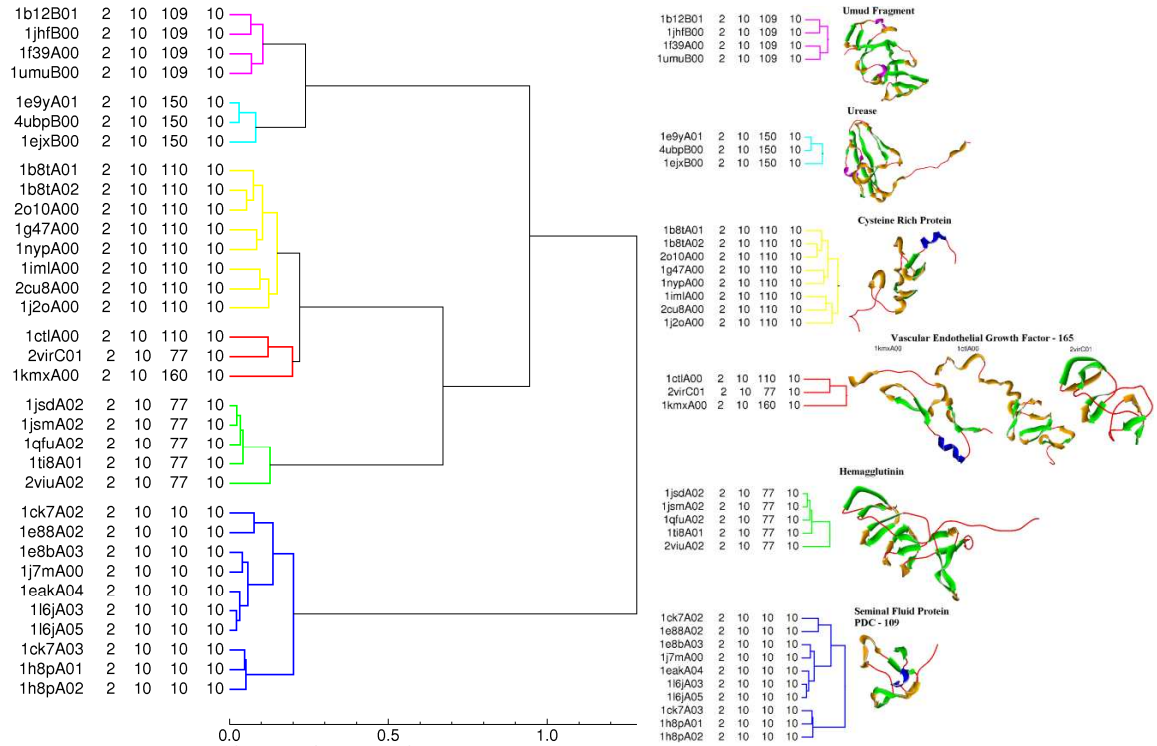


Figure 5. SPF clustering of the All-β class. Left: the dendrogram obtained using $N=6$ with six clusters; Right: the corresponding groups and the representative proteins for each group.

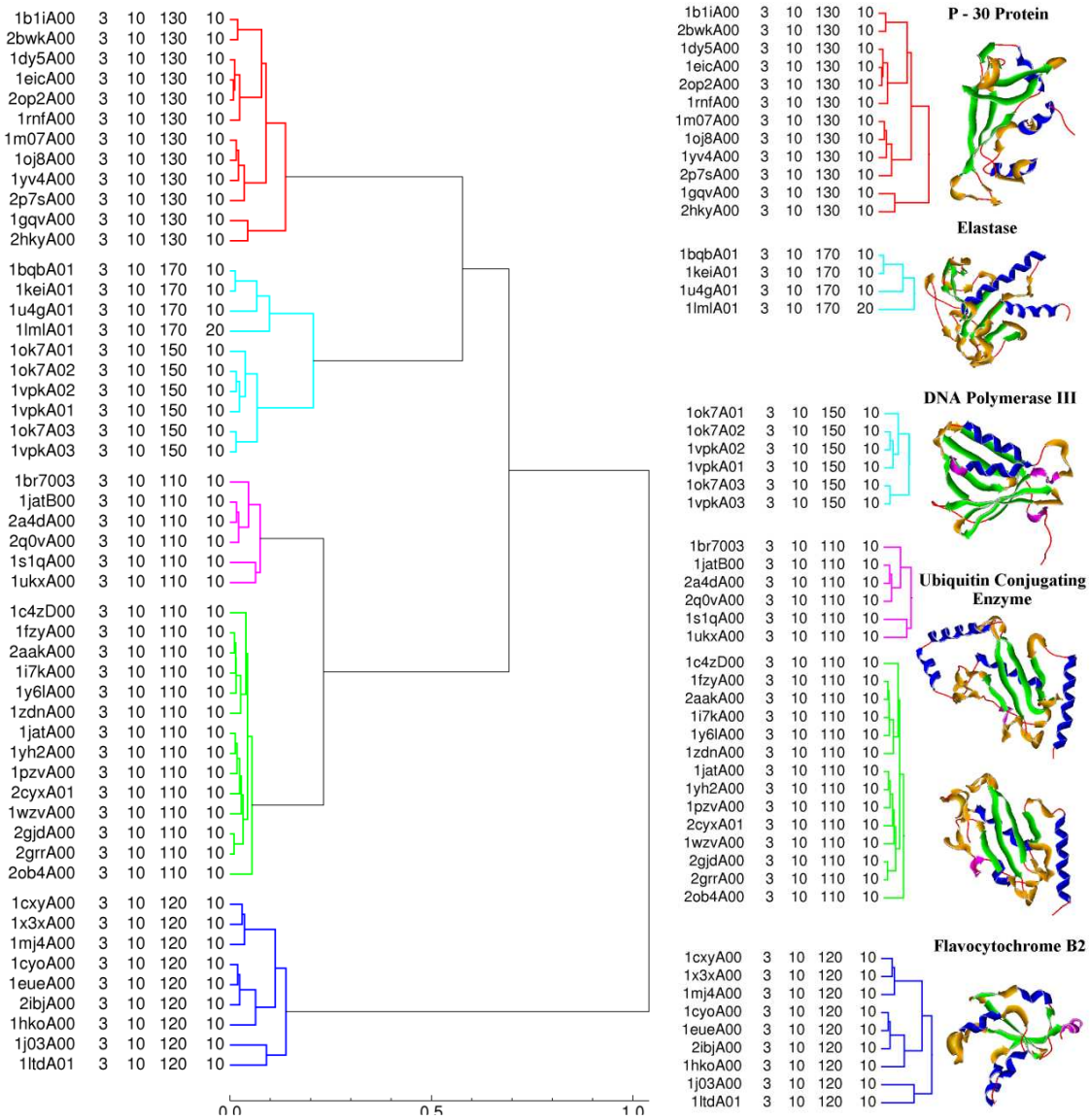


Figure 6. SPF clustering of α + β class. Left: the dendrogram obtained using $N=6$ with five clusters; Right: the corresponding groups and the representative proteins for each group.

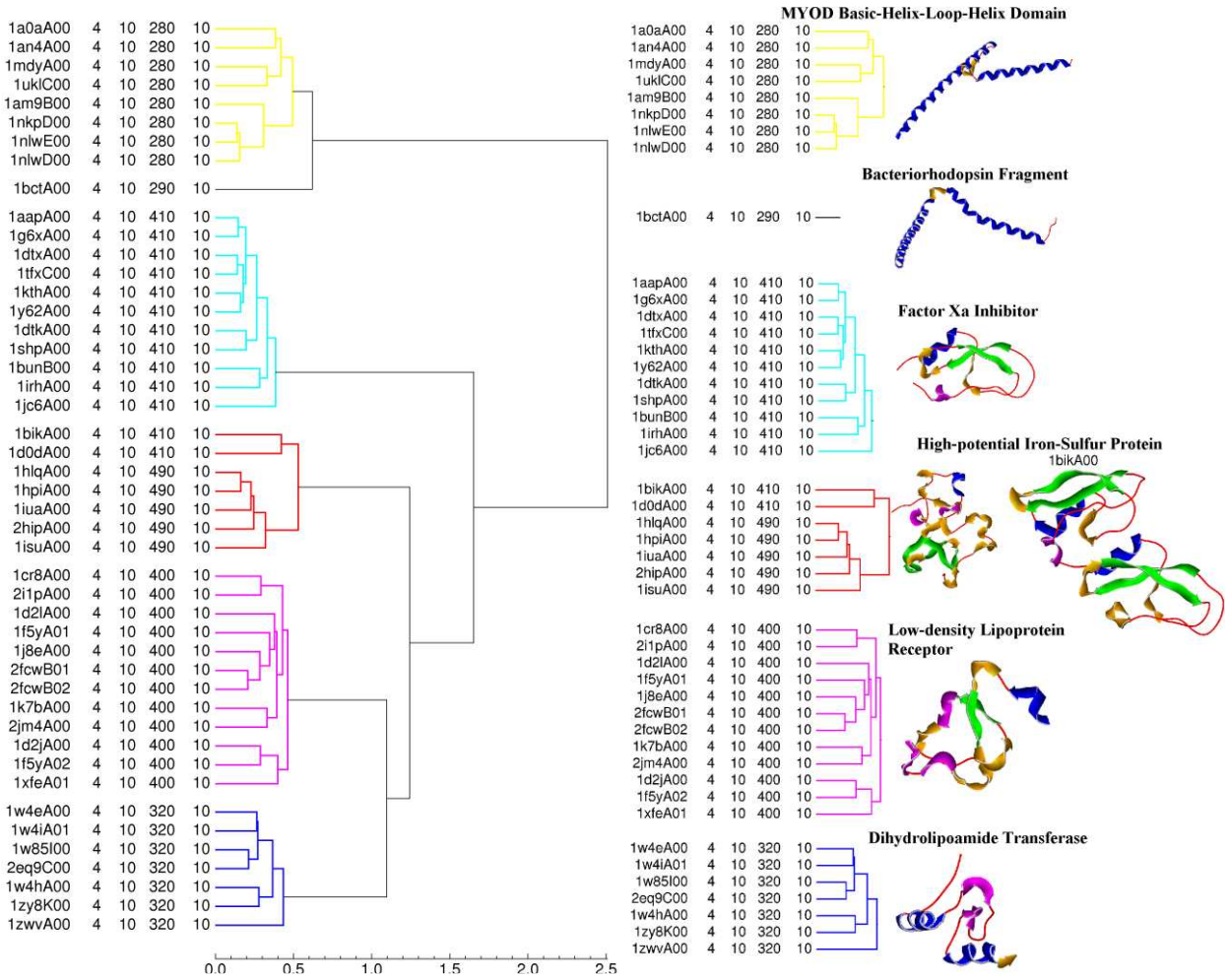


Figure 7. SPF clustering of the irregular class. Left: the dendrogram obtained using $N=6$ with six clusters; Right: the corresponding groups and the representative proteins of each group.

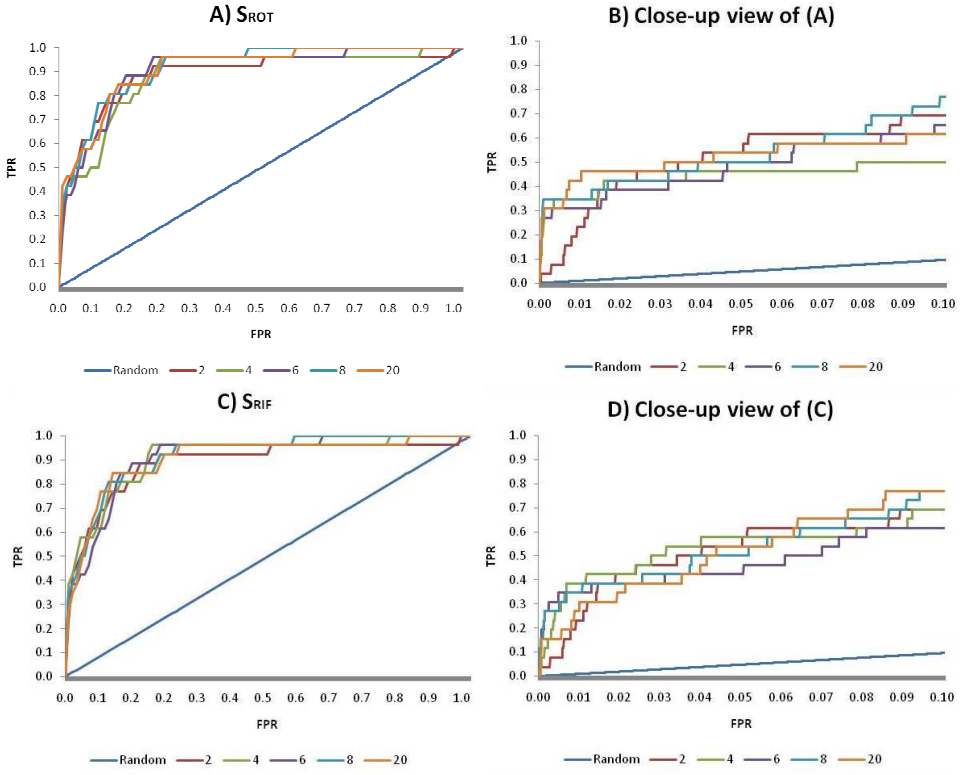


Figure 8. ROC plot analyses obtained when querying the entire CATH database with the 12asA00 structure. A: rotation-dependent scoring function (Eq 2); B: close-up view of (A); C: RIF scoring function (Eq 4); D: close up view of (C).

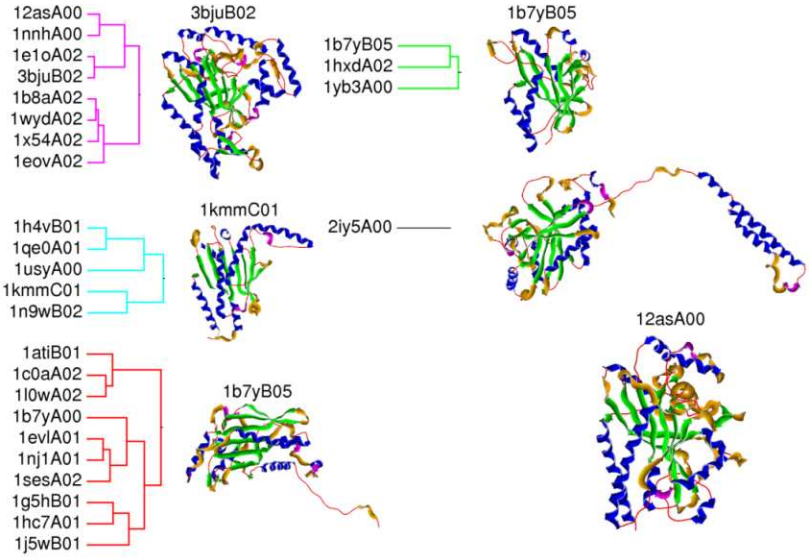


Figure 9. Clustering the CATH super-family 3.30.930.10 into five groups. The representative members of each group are illustrated along with the query protein 12asA00.

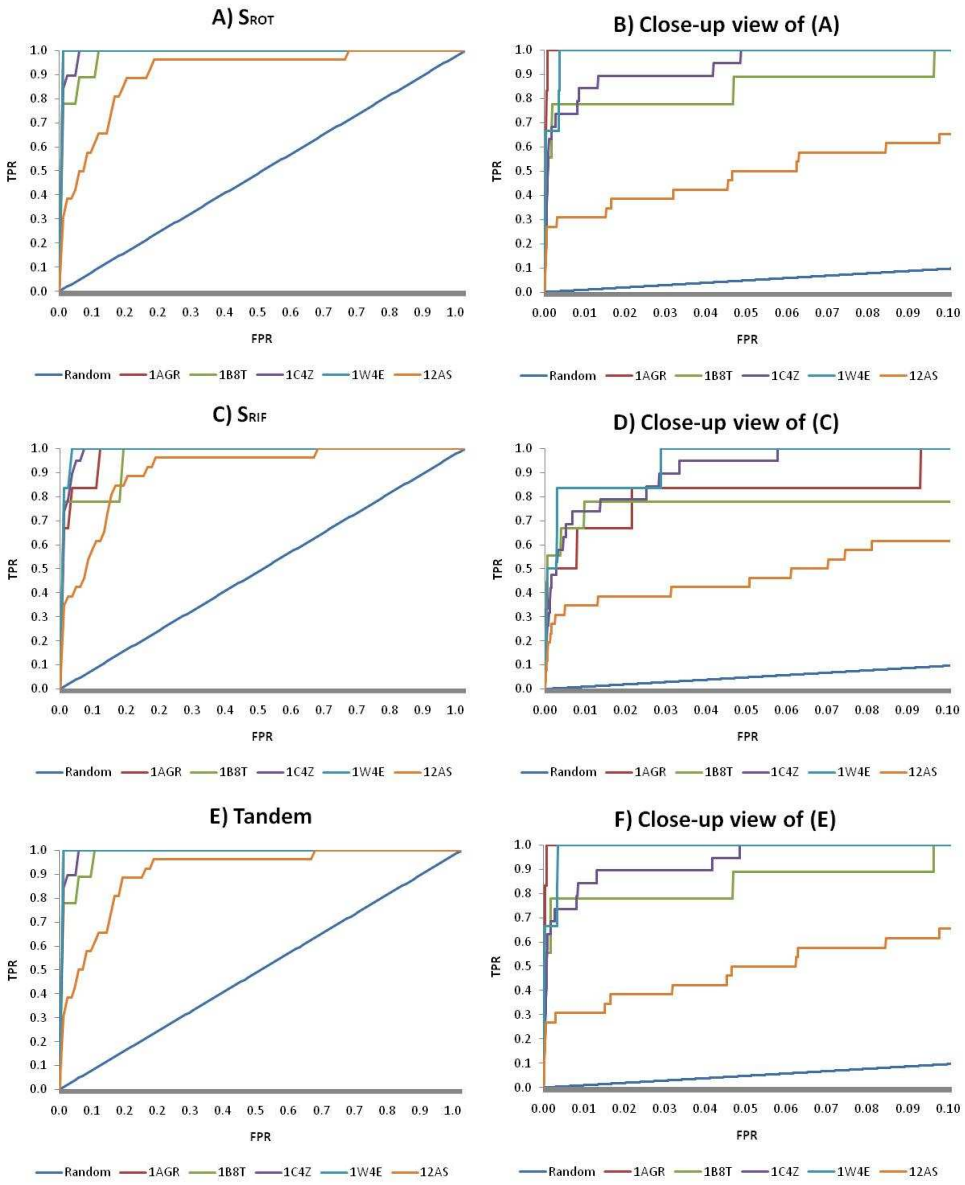


Figure 10. ROC plot analyses obtained when querying the entire CATH database using the 1AGR, 1B8T, 1W4E, 1C4Z and 12AS, structures. A: rotation-dependent scoring function (Eq 2); B: close-up view of (A); C: RIF scoring function (Eq 4); D: close up view of (C); E: Tandem scoring; F: close up view of (E).

4. Discussion and Conclusions

It has been shown that low resolution SPF expansions provide a reliable and fast sequence-independent way to superpose and compare protein structures. The clustering results of the four CATH super-families show that SPF clustering generally agree with the CATH classification. The database search results show that a large protein structure database can be queried accurately using SPF expansions to order $N=6$. In order to accelerate further the scoring calculation, the use of a simple rotationally invariant scoring function was investigated. Our RIF scoring function is significantly faster to calculate, but it is less precise than a rotational search. Nonetheless, the RIF function may still be used effectively as an initial filter when searching large databases. Using the RIF function in tandem with the rotational function to re-score only the top matches of the database gives very promising results.

Thus tandem searches are much faster than full rotational searches, yet the recall/precision is almost equivalent. Currently a limitation of our approach is that proteins larger than a typical domain of around 100-150 residues are not represented well due to the exponential decay of the GL functions at large radial distances. However, we are working to extend the resolution range of our approach to be able to describe proteins of any size, and we are enhancing the approach to be able to detect the presence of symmetrical domains and repeated motifs, for example. We also aim to develop a web-based interface for general use by the biological community. As well as being able to calculate rapidly sequence-independent protein structure superpositions, we believe that the SPF approach could provide an automatic and objective way to enhance the quality of protein structure classifications.

Acknowledgements
This work is funded by the Agence Nationale de la Recherche, grant reference: ANR-08-CEXC-017-01.

References

1. L. Holm and C. Sander. *Trends in Biochemical Sciences.* **20**, 478 (1995).
2. H.M. Berman *et al. Acta Cryst.* **D58**, 899 (2002).
3. R. Kolodny, P. Koehl, and M. Levitt. *J. Mol. Biol.* **346**, 1173 (2005).
4. E. Krissinel and K. Henrick. *Proceedings of the Fifth international Conference on Molecular Structural Biology, Vienna*. 88 (2003).
5. I.N. Shindyalov and P.E. Bourne. *Protein Engineering* **11**, 739 (1998).
6. T. Madej, J.F. Gibrat and S.H. Bryant. *Proteins: Struct, Func. Bioinf.* **23**, 356 (1995).
7. M.J. Sippl and M. Widerstein. *Bioinformatics.* **24**, 426 (2008).
8. A.L. Cuff, I. Sillitoe, T. Lewis, O.C. Redfern, R. Garratt, J. Thornton, and C.A. Orengo. *Nucleic Acids Research.* **37,** 310 (2008).
9. A.G. Murzin, S.E. Brenner, T. Hubbard, and C. Chothia. *J. Mol. Biol.* **247**, 536 (1995).
10. C.A. Orengo and W.R. Taylor. *Methods Enzymol.* **266**, 617 (1996).
11. L. Mak, S. Grandison, and R.J. Morris. *J. Mol. Graph. Model.* **26**, 1035 (2008).
12. L. Sael, B. Li, D. La, Y. Fang, K. Ramani, R. Rustamov, and D. Kihara. *Proteins: Struct, Func. Bioinf.* **72**, 1259 (2008).
13. M. Novotni and R.J. Klein. *J. Comp.-Aided Mol. Des.* **36**, 1047 (2004).
14. D.W. Ritchie and G.J.L. Kemp. *Proteins: Struc., Funct., Genet.* **39**, 178 (2000).
15. D.W.Ritchie, D. Kozakov, and S. Vajda. *Bioinformatics.* **24**, 1865 (2008).
16. S.E. Leicester, J. L. Finney and R.P. Bywater, *JMG.* **6**, 104 (1988).
17. S.E. Leicester, J. Finney and R. Bywater. *J. Math. Chem.* **16**(3-4), 315 (1994).
18. S.E. Leicester, J. Finney and R. Bywater. *J. Math. Chem.* **16**(3-4), 343 (1994).
19. N.L. Max and E.D. Getzoff. *IEEE Comput. Graphics Appl.* **8**(4), 42 (1988).
20. N.L. Max. *JMG.* **6**(4) 210 (1988).
21. B.S. Duncan and A.J. Olson. *Biopolymers.* **33**, 219 (1993).
22. B.S. Duncan and A.J. Olson. *Biopolymers.* **33**, 231 (1993).
23. B.S. Duncan and A.J. Olson. *JMG.* **13**, 258 (1995).
24. B.S. Duncan and A.J. Olson. *JMG.* **13**, 250 (1995).
25. A. Gramada and P.E. Bourne, *BMC Bioinformatics*, **7**, 242 (2006).
26. R.J. Morris, R.J. Najmanovich, A. Kahraman and J.M. Thornton, *Bioinformatics*, **21**(10), 2347 (2005).
27. D.W. Ritchie. *J. Appl. Cryst.* **38**, 808 (2005).
28. J.P. Egan. *Academic Press, New York* (1975).
29. L. Mavridis, B.D. Hudson and D.W. Ritchie, *J. Chem.Inf. Model.*, **47**(5), 1787 (2007).
30. M. Gerstein and M. Levitt, *Protein Science*. **7**(2), 445 (1998).
31. A.D. McLachlan. *Acta Cryst.* **A38**, 871 (1982).
32. J.H. Ward. *J. Am. Stat. Assoc.* **58**(301), 236 (1963).

STRUCTURAL PREDICTION OF PROTEIN-RNA INTERACTION BY COMPUTATIONAL DOCKING WITH PROPENSITY-BASED STATISTICAL POTENTIALS

LAURA PÉREZ-CANO[1], ALBERT SOLERNOU[1], CARLES PONS[1,2], JUAN FERNÁNDEZ-RECIO[1]

[1] *Life Sciences Department, Barcelona Supercomputing Center (BSC),*

[2] *Computational Bioinformatics, National Institute of Bioinformatics (INB),*

Jordi Girona 29, Barcelona, 08034, Spain

Despite the importance of protein-RNA interactions in the cellular context, the number of available protein-RNA complex structures is still much lower than those of other biomolecules. As a consequence, few computational studies have been addressed towards protein-RNA complexes, and to our knowledge, no systematic benchmarking of protein-RNA docking has been reported. In this study we have extracted new pairwise residue-ribonucleotide interface propensities for protein-RNA, which can be used as statistical potentials for scoring of protein-RNA docking poses. We show here a new protein-RNA docking approach based on FTDock generation of rigid-body docking poses, which are later scored by these statistical residue-ribonucleotide potentials. The method has been successfully benchmarked in a set of 12 protein-RNA cases. The results show that FTDock is able to generate near-native solutions in more than half of the cases, and that it can rank near-native solutions significantly above random. In practically all these cases, our propensity-based scoring helps to improve the docking results, finding a near-native solution within rank 100 in 43% of them. In a remarkable case, the near-native solution was ranked 1 after the propensity-based scoring. Other previously described propensity potentials can also be used for scoring, with slightly worse performance. This new protein-RNA docking protocol permits a fast scoring of rigid-body docking poses in order to select a small number of docking orientations, which can be later evaluated with more sophisticated energy-based scoring functions.

1. Introduction

Understanding the molecular mechanism of protein-RNA recognition in order to understand and predict such interactions is one of the grand challenges in structural biology. In recent years, the growing awareness for the importance of RNA in the context of protein-RNA interactions, together with the publication of the 50S and 30S ribosome subunits,[1,2] have increased the volume of data on these complexes. However, in spite of this, the level of structural knowledge of protein-RNA association is quite poor in comparison to that of other biomolecules.

Given that experimental determination of protein-RNA complexes at high resolution is challenging (being crystallization one of the main bottle-necks), computational approaches for structural modeling at different resolution levels are increasingly needed in order to complement existing experimental data on protein-RNA interactions of interest. One promising tool is computational protein-RNA docking, which can provide structural models at residue and even atomic resolution level, but there are still very few reported methods as compared to protein-protein or protein-ligand docking, and certainly there are no systematic studies on large data sets. To our knowledge, the largest reported benchmark set for protein-RNA docking consisted on five cases.[3] In this context, the Critical Assessment of PRediction of Interactions (CAPRI) experiment (http://www.ebi.ac.uk/msd-srv/capri), a blind international docking competition to evaluate the performances of protein-protein computational docking methods, proposed recently the first protein-RNA complex target. The experiment nicely showed how some docking methods can be adapted to predict the tridimensional structure of a protein-RNA complex. Indeed, our pyDock scoring protocol,[4] which achieved excellent results for protein-protein docking in past CAPRI tests,[5] identified the second best model among all participants in the scorer experiment, with excellent ligand RMSD of 3.8 Å with respect to the X-ray structure of the complex. However, this experiment also highlighted the limitations of current methods. In addition to better treatment of flexibility, new scoring parameters specifically adapted for protein-RNA binding are needed.

In this sense, a number of studies have used available structural data of real protein-RNA interfaces in order to understand this type of interaction and extract better parameters for predictions.[6-16] Some of these studies reported individual propensities and all-atom statistical potentials for the characterization of modeled protein-RNA interactions at atomic level. For instance, all-atom hydrogen-bond statistical potentials have been applied to identify

near-native docking solutions.[3] Other all-atom statistical potentials found interesting details of the protein-RNA interaction.[12-16] However, coarse-grained statistical potentials at residue-nucleotide level have the advantage of low computational cost for their application to larger benchmark sets in order to develop new docking methods (something that is needed before considering more detailed functions). In a previous work,[6] we extracted and successfully used single residue interface propensities for protein-RNA to identify RNA-binding sites in proteins. In the present study, we have extracted new pairwise residue-ribonucleotide propensities for protein-RNA with the goal of being of predictive value for docking. For that, we have used a standard FFT-based approach to generate protein-RNA rigid-body docking poses, and then we have used the new propensities to successfully score these docking poses. We have also tested other reported single residue or residue-ribonucleotide propensities,[6,9,15,16] in order to check the capability of these statistical potentials for the scoring of rigid-body docking poses.

2. Methods

2.1. *Pairwise residue-ribonucleotide interface propensities from protein-RNA structural data*

We extracted protein-RNA pairwise interface propensities using the same training data set that we previously used to extract individual interface propensities, which was composed of 282 non-redundant protein-RNA interactions.[6] These propensities can be calculated from the observed frequency of the specific residue-ribounucleotide pairs of type *pq* (p = 1...20 for amino acid residues; and q = 1...4 for ribonucleotides) at the protein-RNA interfaces, as compared with the expected frequency of these pairs according to the protein and ribonucleotide surface composition, as it is shown by equations 1-4:

$$P^I_{pq} = \frac{N^I_{pq} / \sum_{pq} N^I_{pq}}{N^S_p / \sum_p N^S_p \times N^S_q / \sum_q N^S_q} \qquad (1)$$

where N^I_{pq} is the number of pairs between residue type p and ribonucleotide type q at the protein-RNA interfaces (the pairs were defined by having at least one atom within 4 Å distance from each other), ΣN^I_{pq} the total number of residue-ribonucleotide pairs at protein-RNA interfaces, N^S_p and N^S_q the number of surface residues and ribonucleotides of type p and q respectively (surface residue or ribonucleotide were defined as those with accessible surface area ASA > 0.1 Å^2), and ΣN^S_p and ΣN^S_q the total number of surface residues and ribonucleotides, respectively.

Then, propensities P^I_{pq} were easily converted to free-energy estimates or statistical potentials by equation 2:

$$\Delta G^{stat}_{pq} = -RT\ln(P^I_{pq}) \qquad (2)$$

The statistical potential ΔG^{stat}_{pq} thus represents the empirical energy of forming a pair between a residue of type p and a ribonucleotide of type q at the interface, given their frequencies at the protein and ribonucleotide surfaces, being R the gas constant and T the absolute temperature (we have used here RT as 0.59 kcal/mol). Therefore, negative statistical potential values indicate favorable binding energies.

2.2. *Protein-RNA rigid-body docking and scoring by propensity-based statistical potentials*

We used FTDock[17] to generate 10.000 protein-RNA docking poses. We used the same FTDock version as we previously used for testing our pyDock method for scoring of protein-protein docking, that is, with no electrostatics and 1.2 Å grid resolution.

We evaluated all generated docking poses by a very fast algorithm that scored solutions based on the existing contacts at interface. For every residue-ribonucleotide pair at the interface of the docking pose (that is, those that have at least one atom within 4 Å distance from each other), the corresponding propensity value according to its type was assigned. The propensity-based values of all pairs were summed and formed the final score of the given docking pose *i*, as in equation 3:

$$\Delta G_i^{stat} = \sum_{pq} \Delta G_{pq}^{stat} \qquad (3)$$

Finally, all docking solutions were ranked according to these propensity-based scores. For comparison, we also tested other previously described propensity values, either pairwise residue-ribonucleotide or single residue propensities. In the case of pairwise residue-ribonucleotide propensities, we converted the reported values (usually observed and expected frequencies) to statistical potentials as above described. In the case of single residue propensities, we summed the corresponding values of all interface residues according to their types.

2.3. *Benchmarking the method on known protein-RNA complex structures*

In order to benchmark our method, we compiled a set of non-redundant protein-RNA complexes of known structure, in which there is an available unbound structure for at least one of the two components. This produced a total of 12 cases, two of which had available structure for both unbound protein and RNA molecules, five had available only the unbound protein structure, and the remaining five had available structure for only the unbound RNA (Table 1). In order to avoid redundancy we ensured there was no more than 70% of sequence identity between any pair of proteins within the data set. On the other hand, we considered as unbound proteins those with more than 95% of sequence identity with respect to the bound protein, and as unbound RNAs those with more than 85% of sequence identity with respect to the bound RNA structures in the protein-RNA complexes.

Table 1. Structural data set of protein-RNA interactions used in this study. For each molecule the PDB and chain identifiers are shown. The RMSD in Å between the receptor or ligand used here and the bound structure is also shown in brackets.

Name	Complex PDB	Receptor PDB (RMSD)	Ligand PDB (RMSD)
Tyrosyl-tRNA synthetase splicing factor / group I intron RNA	2RKJ_a:c	1Y42_x (0.9)	1Y0Q_a (3.0)
Ct domain of elongation factor SelB / SECIS RNA	1WSU_a:e	1LVA_a (0.7)	1MFK_a (3.1)
NF-Kb / anti-NFKb RNA aptamer	1OOA_a:c	1OOA_a (0.0)	2JWV_a (5.4)
Stnthetic Fab / P4-P6 ribozyme domain	2R8S_l:r	2R8S_l (0.0)	1HR2_a (4.3)
Elongation factor SelB from E.Coli / SECIS RNA	2PJP_a:b	2PJP_a (0.0)	1MFK_a (3.1)
SRP 19 / 7S.S SRP RNA	1LNG_a:b	1LNG_a (0.0)	1Z43_a (2.1)
SRP ribonucleoprotein core Variant 6 / RNA	2PXV_a:b	2PXV_a (0.0)	1CQL_a (8.1)
RNA-binding protein 15.5 K complexed / RNA	1E7K_a:c	2JNB_a (3.2)	1EK7_c (0.0)
HutP / Hut mRNA	1WPU_a:c	1WPV_a (0.2)	1WPU_c (0.0)
Norwakl Virus Polymerase with CTP / RNA	3BSO_a:p	1SH0_b (1.3)	3BSO_p (0.0)
Pp7 Coat protein dimer in complex / RNA hairpin	2QUX_a:c	2QUD_a (0.8)	2QUX_c (0.0)
Structure of 9-subunit archaeal exosome / RNA	2JEA_a:c	2JE6_a (0.4)	2JEA_c (0.0)

This set of structures was used to benchmark the docking results. We compared the predicted docking poses with the real protein-RNA complex structures by superimposing protein alpha-carbons of predicted and real complexes, and then calculating the RMSD between the RNA molecules (considering all atoms). A near-native solution was defined as a docking pose with RNA RMSD, calculated as described above, smaller than 10 Å, which is in line with the criteria used in the CAPRI experiment. This was calculated for the 10,000 docking poses generated by FTDock,[17] and we computed success rates as the percentage of cases in which at least a near-native solution was found within a given number of docking poses as scored by the docking algorithm. The success rates expected by random were calculated by randomly shuffling the scores of the docking solutions (the process was repeated 100 times and the average was calculated).

3. Results

3.1. *New pairwise residue-ribonucleotide interface propensities for protein-RNA*

We computed pairwise residue-ribonucleotide interface propensities from a set of protein-RNA complex structures, and then we converted them to statistical binding potentials (see Methods). The resulting values for all residue-ribonucleotide types are shown in Figure 1. We found in this analysis that the most populated residue-ribonucleotide pairs at protein-RNA interfaces are those composed of the amino acid residues arginine (R), lysine (K) and histidine (H). On the contrary, the least favored pairs are composed of the following residues: aspartic acid (D), glutamic acid (E), cysteine (C), valine (V), leucine (L) and isoleucine (I). In most of the cases, for a given residue type, we do not see significant differences in the pairwise propensity values with regards to the four ribonucleotide types. This can be clearly seen in Figure 1, in which the major differences can be found among the residue types and not among the ribonucleotide types. These results are consistent with our previously reported single residue propensities for protein-RNA,[6] and show the important role of electrostatic forces in protein-RNA binding, with negative RNA charge playing a determinant role in RNA-binding areas in proteins. Interestingly, the important role of electrostatics in protein-RNA binding underlines a major difference with protein-protein association, where desolvation and hydrophobic effect seem much more important.

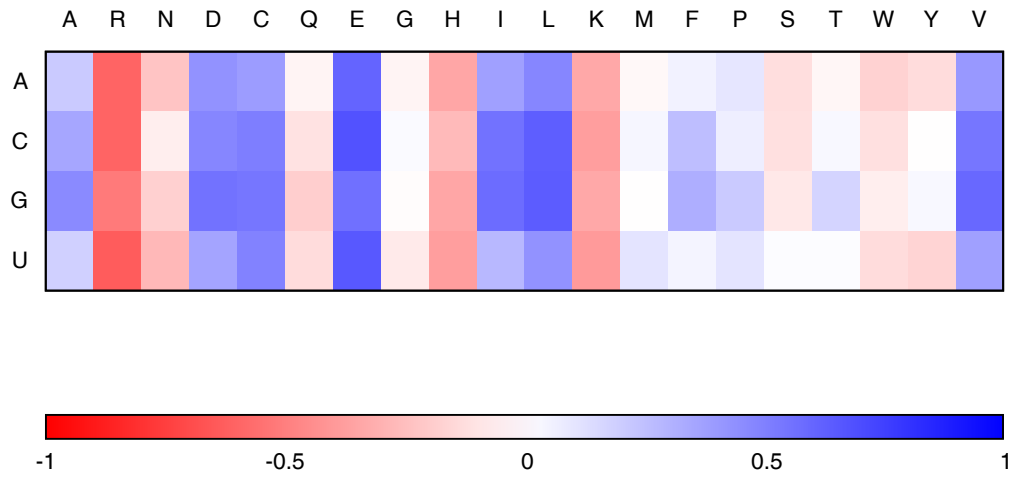


Figure 1. Pairwise protein-RNA statistical potentials (favorable pairs are in red; disfavored in blue).

3.2. *Protein-RNA rigid-body docking and scoring by propensity-based statistical potentials*

For seven out of the 12 cases, the FTDock rigid-body docking generated at least a near-native solution within the total 10,000 docking poses. The results are shown in Table 2. In two cases, the near-native solutions are ranked below 10 by FTDock, but the rest of cases there is no near-native solution ranked below 100. We can also see that the propensity-based scoring alone improves the best rank of a near-native solution in three of the cases, as compared with the original FTDock scores. As a consequence, in three cases a near-native solution is found with rank below 100. Moreover, when the propensity-based scores are used in combination with the original FTDock scores (simply by adding the scores, no weight optimization has been attempted here to avoid overfitting given the small number of cases), in practically all cases the near-native rank values improve with respect to the original FTDock scores (except 1WPU, which slightly worsens from 159 to 170). Actually, in four cases a near-native solution is found with rank below 100. This indicates a clear predictive value of the pairwise propensity scores. Interestingly, the FTDock and the propensity values are quite complementary: in three of the cases, a near-native solution is found with rank below 10 by either FTDock or propensity-based scoring.

Table 2. Results of protein-RNA docking and scoring. The total number of near-native solutions in the docking set is shown. The best rank of a near-native solution (RMSD < 10Å) is shown, after scoring by FTDock, by propensity-based potentials, and by combined score (in brackets is given the RMSD in Å of the near-native solution with respect to the x-ray complex structure, in addition to the fraction of native f_{nat} and non-native $f_{non\text{-}nat}$ contacts as defined in CAPRI).

Complex PDB	Number of near-native solutions	Best near-native rank by FTDock scoring (RMSD) (f_{nat}; $f_{non\text{-}nat}$)	Best near-native rank by propensity-based scoring (RMSD) (f_{nat}; $f_{non\text{-}nat}$)	Best near-native rank by FTDock + propensity scoring (RMSD) (f_{nat}; $f_{non\text{-}nat}$)
1WSU	5	2015 (8.9) (0.23; 0.84)	70 (9.0) (0.23; 0.88)	1049 (9.0) (0.23; 0.88)
2PJP	9	1590 (9.3) (0.43; 0.75)	213 (9.7) (0.43; 0.75)	763 (8.9) (0.43; 0.75)
1LNG	4	131 (5.3) (0.55; 0.22)	660 (5.3) (0.55; 0.22)	92 (5.3) (0.55; 0.22)
1E7K	46	7 (9.6) (0.05; 0.90)	778 (8.7) (0.18; 0.67)	7 (9.6) (0.05; 0.90)
1WPU	44	159 (8.1) (0.35; 0.57)	1989 (9.8) (0.19; 0.74)	170 (8.1) (0.35; 0.57)
2QUX	14	157 (8.1) (0.45; 0.70)	1 (8.5) (0.45; 0.67)	60 (8.1) (0.45; 0.70)
2JEA	17	9 (9.1) (0.18; 0.95)	61 (8.3) (0.18; 0.96)	7 (9.1) (0.18; 0.95)

3.3. *Examples of successful predictions*

It is remarkable that for the 2QUX case (unbound protein vs. bound RNA), the scoring by the new pairwise propensities is able to find a near-native solution ranked 1. As can be seen in Figure 2, the predicted RNA orientation on the protein surface is very close to that in the x-ray structure. The best rank obtained by FTDock scoring was 157, so the effect of using the new propensity-based potentials on the final scoring is dramatic in this case.

Another example of successful application of protein-RNA docking can be found in a recent CAPRI blind experiment. Targets 33 and 34 were a protein-RNA case. In target 33, both molecules (protein and RNA) needed to be modeled since there was no available x-ray structure. That was an extremely difficult case for which no group was able to submit an acceptable model. Target 34 was the same complex, but with the bound structure of the RNA provided by the organizers (although with random orientation). For this target, we generated docking poses with FTDock and with RotBUS (and in-house developed program for rigid-body docking; see upcoming publication). The docking poses were scored by our protein-protein scoring function pyDock,[4] with desolvation parameters for

RNA adapted from those used by us in protein-protein docking. In addition, we applied distance restraints to one residue-ribonucleotide pair and six ribonucleotides that, according to literature, were likely to be at the interface. The result was an acceptable model within the ten submitted models (it was ranked 885 before applying restraints, and rank 3 after restraints). Moreover, in the scorers experiment, our method identified the second best model among all participants, with excellent ligand RMSD of 3.8 Å with respect to the X-ray structure of the complex.

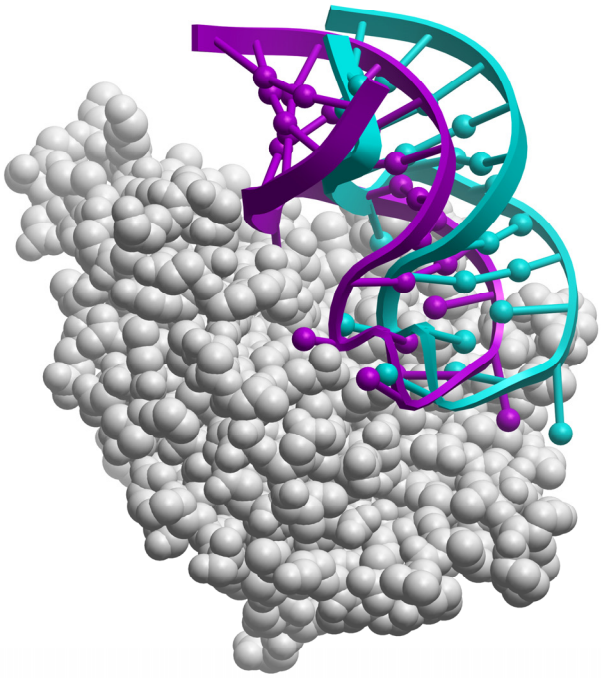


Figure 2. Solution ranked 1 after the new propensity-based scoring in the docking of Pp7 Coat protein dimer with RNA hairpin (protein in white CPK; RNA in cyan ribbon). For comparison, the x-ray structure (PDB 2QUX) of the protein-RNA complex is shown (RNA in magenta ribbon, after superimposing protein molecules).

4. Discussion

4.1. *New pairwise propensities for protein-RNA interaction. Comparison to other reported propensities*

We can compare our new propensities with other reported protein-RNA propensities, all of them derived from smaller data sets.[9,15,16] In practically all of the studies, the favored residues are R and K, but their pairwise preferences for the ribonucleotides change. In our work here and in others,[9,16] there are no significant differences on the preferences of R and K for any ribonucleotide, but in some studies they see preferred pairs as R-U,[11,12,15] or K-A.[11,12] Interestingly, in our study we can see H residues with higher pairwise propensities, while this is not observed in any of the previous studies. Histidine residue can act as positively charged, depending on the environment, so this can explain its high propensities. This was already discussed in our previous study on single residue interface propensities.[6] On the other side, while in our study the aromatic residues are not amongst the most preferred pairs, in other studies they are.[9,16] We have seen before that this can strongly depend on the data set used for deriving the propensities. For instance, we checked that there are more aromatic residues involved in the interaction with single chain RNA molecules, so the proportion of this type of RNA in the data set could modify the propensity values.[6] Interestingly, we see that although the global propensities of pairs involving W and Y residues are small (favorable,

but small), they have slightly higher preferences for A and U ribonucleotides (especially in the case of Y residue). While all the reported propensities may reflect different characteristics of the protein-RNA interaction, we are more confident in the general applicability of the propensities described in this work, given that they were extracted from the largest data set so far (282 non-redundant protein-RNA interactions).[6]

4.2. *Use of other reported propensities (single and pairwise) for scoring*

We have seen above that our new pairwise residue-ribonucleotide propensities improve the scoring of rigid-body protein-RNA docking poses. In order to check whether this represents an advance over other reported protein-RNA propensities, we have also used these other propensities in the same conditions to check their results in docking. For this, we selected different protein-RNA propensities from bibliography[9,15,16] and from an own previous work.[6] We used both pairwise residue-ribonucleotide interface propensities[6,9,15,16] as well as single residue interface propensities.[6,9,16] We compiled a total number of seven different protein-RNA propensity matrices, five of which were pairwise propensities, and two single residue interface propensities. Because the propensity definition varied among the different studies, we considered for all of them the reported expected (F_{obs}) and observed (F_{exp}) frequency values, which we transformed into propensities ($P = F_{obs} / F_{exp}$) and then into their respective statistical potentials following our definition (eq. 2 and 3).

The results of using different propensity values can be seen in Figure 3. In general, the pairwise propensities described in this work, especially when is combined with FTDock score, give the best success rates. Actually, most of the other propensities give success rates that are not significantly above random. However, the pairwise propensities from Westhof laboratory[16] had reasonable good results, indeed significantly above random. Moreover, it is interesting that single residue propensities from our previous work[6] and from Westhof laboratory[16] obtained quite good success rates. Two important conclusions can be derived from this: *i)* the determinant for protein-RNA specificity lies mostly on the protein residues, and *ii)* the predictive value of the statistical potentials depend strongly on the size and composition of the database used to derive the propensities.

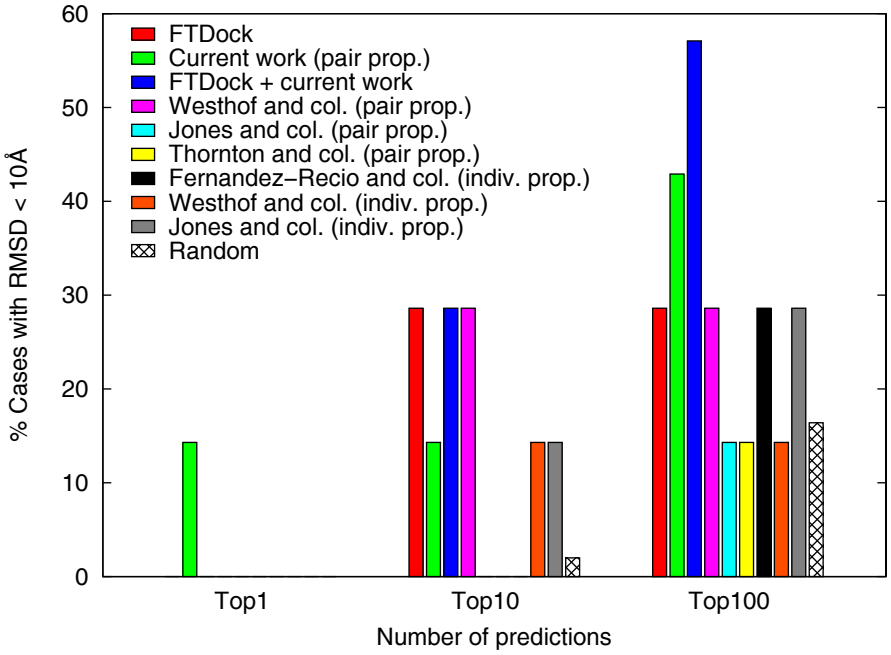


Figure 3. Success rates of protein-RNA docking after scoring by different propensity-based statistical potentials.

4.3. *FTDock docking and scoring performance is better than in protein-protein docking*

While the number of protein-RNA cases for which a near-native solution is found by FTDock (58%) is inferior to that in protein-protein docking (where a near-native solution is found in 77% of the cases), it is interesting that for some cases, the number of near-native solutions is quite high (Table 2). Especially in the cases of 1EK7 and 1WPU, more than 40 near-native solutions are found, a number clearly above the performance in protein-protein docking. Moreover, the success rate of FTDock scoring for the top 10 solutions is well above random (in protein-protein, success rates for FTDock scoring, run in the same conditions as here, was always similar to random).[4] This could indicate that geometry complementarity is more important in protein-RNA than in protein-protein (electrostatics should also be important in the interaction, but the FTDock version we are running here has no electrostatics, so this should not affect the scoring). This higher importance of surface complementarity is consistent with the fact that no statistical preferences are found within the ribonucleotides. That is, while protein residues have different preferences derived from their different chemico-physical characteristics, the ribonucleotides are more similar in thermodynamic behaviour, and perhaps the contribution of RNA to specificity lies more on conformational aspects. Actually, in the cases where FTDock has the best results (1EK7, 2JEA), the RNA molecule is in the bound conformation, and thus the rigid-body approach can take full advantage of the geometry complementarity.

5. Conclusions

In summary, we have proposed here a new protocol for protein-RNA docking, based on FFT rigid-body docking followed by scoring with new pairwise residue-ribonucleotide interface propensities derived from protein-RNA complex structures. The docking approach and new propensities have been tested in the largest protein-RNA benchmark, to our knowledge. The results show that FTDock can be successful if RNA conformation is in the bound conformation, and that the new propensities help to improve the rank of the near-native docking poses in virtually all cases.

From the results shown here and our experience in the blind CAPRI experiment, we can envisage a possible strategy for protein-RNA docking. First, rigid-body docking between protein and RNA based on FFT algorithms or on any other efficient approach. Then, we can use a combination of geometry-based scoring and propensity statistical potentials as a filter to select a few hundred docking poses, which later will be evaluated with more complete energy function. Finally, the use of a minimal information that can be integrated as distance restraints can dramatically improve the results. One of the main bottlenecks for continuing development in protein-RNA docking is the lack of cases in which both the unbound protein and RNA structures are simultaneously available. Therefore, it seems that future developments will need to focus on the use of homology-based models of RNA in docking.

Although beyond the scope of current work, it would be interesting in the future to check the capabilities of our protein-RNA statistical potentials for the prediction of protein-DNA interactions. However, reported data on protein-RNA and protein-DNA propensities highlight the specific differences between these types of interfaces, specifically that in DNA-binding the phosphate group is determinant for the interaction, so protein charged residues are preferred, while in RNA-binding the ribose and nucleotides are more relevant, being the protein aromatic residues key for specificity.[7,8,10,12,14] These findings, together with preliminary tests (data not shown) indicate that our protein-RNA propensities are specific for protein-RNA binding. As future work, derivation of new protein-DNA propensities, in the same fashion as that described here for protein-RNA, could be successfully applied for DNA-binding interface prediction and docking studies.

Acknowledgments

This work is supported by grant BIO2008-02882 from Plan Nacional I+D+i (Spanish Ministry of Science). LP-C is recipient of an FPU fellowship from the Spanish Ministry of Science.

References

1. N. Ban, P. Nissen, J. Hansen, P.B. Moore and T.A. Steitz, Science 289:905-920 (2000).
2. B.T. Wimberly, D.E. Brodersen, W.M. Clemons, R.J. Morgan-Warren, A.P Carter, C. Vonrhein, T. Hartsch and V. Ramakrishnan, Nature 407:327-339 (2000).
3. Y. Chen, T. Kortemme, T. Robertson, D. Baker and G. Varani, Nucleic Acid Res. 32:5147-5162 (2004).
4. T.M Cheng, T.L Blundell and J. Fernández-Recio, Proteins 68:503-15 (2007).
5. S. Grosdidier, C. Pons, A. Solernou and J. Fernández-Recio, Proteins 69:852-8 (2007).
6. L. Pérez-Cano and J. Fernández-Recio, Proteins, *in press* (2009) *[published online: 7 Jul 2009]*.
7. R.P. Bahadur, M. Zacharias and J. Janin, Nucleic Acids Res 36:2705-2716 (2008).
8. C.M. Baker and G.H. Grant, Biopolymers 85:456-470 (2007).
9. J.J. Ellis, M. Broom and S. Jones, Proteins 66:903-911 (2007).
10. N. Morozova, J. Allers, J. Myers and Y. Shamoo, Bioinformatics 22:2746-2752 (2006).
11. O.T. Kim, K. Yura and N. Go, Nucleic Acid Res 34:6450-6460 (2006).
12. D. Leujene, N, Delsaux, B. Charloteaux, A. Thomas and R. Brasseur, Proteins 61:258-271 (2005).
13. E. Jeong, H. Kim, S. Lee and K. Han, Mol. Cells 16:161-167 (2003).
14. J. Allers and Y. Shamoo, J. Mol. Biol. 311:75-86 (2001).
15. S. Jones, D. Daley, N. Luscombe, H. Berman and J. Thornton, Nucleic Acid Res. 29:943-954 (2001).
16. M. Treger and E. Westhof, J. Mol. Recogn. 14:199-214 (2001).
17. H.A. Gabb, R.M. Jackson, M.J. Stemberg, J. Mol. Biol. 272:106-120 (1997).

PERSONAL GENOMICS

Can Alkan

Department of Genome Sciences, University of Washington.
and the Howard Hughes Medical Institute.
Seattle, WA, 98195, USA

Michael Brudno

Department of Computer Science, University of Toronto.
Toronto, ON, M5S 3G4, Canada

Evan E. Eichler

Department of Genome Sciences, University of Washington.
and the Howard Hughes Medical Institute.
Seattle, WA, 98195, USA

Maricel G. Kann

Department of Biological Sciences, University of Maryland Baltimore County.
Baltimore, MD, 21250, USA

S. Cenk Sahinalp

School of Computing Science, Simon Fraser University.
Burnaby, BC, V5A 1S6, Canada

1. Introduction

Until just recently, bioinformaticians and genomicists did not have access to genomic data from multiple human individuals or many different species due to labor intensiveness and limitations of targeted approaches such as array comparative genomic hybridization (arrayCGH), PCR, SNP microarrays and similar assays, and the cost of cloning based traditional sequencing technology. This restricted our understanding of the evolution of species as well as normal and disease-causing genetic variation between different individuals of the same species. Recent improvements in sequencing methods introduced high-throughput, low-cost and cloning-free (thus less labor-intensive) technologies such as *pyrosequencing* (Roche 454), *sequencing by synthesis* (Illumina), *sequencing by ligation* (AB SOLiD), *single-molecule sequencing* (HeliScope), and many others. The revolution in DNA sequencing opened many possibilities for researchers working in the fields of evolution, genetic variation, diseases of genomic origin, and even personalized medicine. The reduced cost for whole-genome resequencing prompted a large-scale genome variation study called the 1000 Genomes Project* that aims to sequence the genomes of approximately more than 1000 individuals from different populations to build the most extensive genetic variation database to date. The new sequencing technologies can also be employed to discover functional landscape of the human genome as part of the ENCODE Project†, such as epigenetic variation (methylation patterns and histone modification) and protein-DNA interaction. Further uses of the high-throughput sequencing technologies include transcriptome analysis, non-coding RNA discovery, gene expression profiling, rapid testing of genotype-phenotype associations, and identification of pathogens.

Our genetic identity not only determines our physical differences, but it also defines our susceptibility against diseases. Several groups are working on various methods to exploit the power of cost efficient se-

*http://www.1000genomes.org
†http://www.genome.gov/10005107

quencing technologies as well as more traditional genome analysis approaches (SNP microarrays, arrayCGH, etc.) to better perform genotype-phenotype associations, in particular to identify susceptibility to disease, and eventually diagnose disease at its early stages. The ultimate goal is to vastly improve the field of pharmacogenomics, which can broadly be defined as the study of the relationship between genotype and drug response and how the drugs affect our metabolism. The abundance of new sequence data gives many opportunities to advancing our understanding of how to optimize drug combinations for each individual's genetic makeup. The underlying computational tools for such studies analyze available sequence data to identify differences between a reference genome and high-throughput sequenced genomes and perform sequence oriented clustering and classification to obtain both normal and disease-related phenotype associations.

This session focuses on the development of novel computational methods in all aspects of Personal Genomics including genetic and epigenetic variation discovery, genotype-phenotype associations, indexing and cataloguing both normal and disease-related variation, exome capture and resequencing, and personalized medicine. This session has a broad target audience that includes algorithm developers working on sequence analysis, genomics researchers, pharmacogeneticists, and medical geneticists.

2. Session Summary

This session includes an invited talk, five reviewed oral presentations, two additional accepted papers and a tutorial. The studies presented in this session focus on the development of computational methods to analyze genomic data generated with various types of methods.

2.1. *Papers*

The paper by **Garten *et al.*** targets an important pharmacogenomics problem that is an essential first step of personalized medicine. This work presents methods to automatically curate a network of drug-gene relationships through text mining. The authors propose that accurate curation of drug-gene relationships together with their previously described algorithm that ranks potential pharmacogenes will help identify genes that can explain variation in drug response.

The paper by **Li, Chen, and Li** addresses the problem of detecting genome-wide haplotype polymorphism. The authors present a computational framework combining the efforts of recombination detection, zero-recombinant haplotype inference and haplotype local structure clustering to jointly use the pedigree and population information. The methods presented in this work accurately reveal the haplotype structure in human populations on a genome-wide level.

The paper by **Greene *et al.*** addresses the problem of test of interaction between genes (epistasis). The authors present a novel permutation test that allows the effects of nonlinear interactions between multiple genetic variants to be specifically tested in a manner that is not confounded by linear additive effects. This method for explicitly testing epistasis or gene-gene interaction effects will likely be complimentary to genome-wide association studies (GWAS) improving our understanding of biological and statistical epistasis and their roles in human health and disease.

Li, Iakoucheva, Mooney, and Radivojac investigate the influence of disease associated mutations on known post-translational modifications. The authors provide a statistical analysis method to estimate statistical confidence of the observed trends of post-translational modification sites and amino acid substitutions.

The paper by **Yavas *et al.*** describes a new software that utilizes SNP microarray data to predict rare copy number variants (CNVs) from raw copy number. Accurate and inexpensive detection of CNVs are particularly important in disease studies where large number of patients with genetic disease can be genotyped. The authors also describe an algorithm based on simulated annealing to refine the breakpoints of the detected CNVs.

In addition to the oral presentations, the Personal Genomics session also contains two more valuable studies published in the proceedings. The paper by **Grady *et al.*** describes a computational framework

composed of a variety of filters to identify a subset of interesting SNPs from a much larger set for efficient interaction analysis in genome-wide association study (GWAS) data. Finally, **Thomson *et al.*** presents the application of the "sequence feature variant type" (SFVT) method to analyze the HLA genetic association in Juvenile Idiopathic Arthritis.

We are excited by the breadth of research in the field of Personal Genomics, and are hopeful that our session will help bring together researchers in these areas. The five papers presented at our session, and the additional two papers in the proceedings were selected with the help of several reviewers, whose help we gratefully acknowledge.

3. Acknowledgments

We would like to thank all the authors who submitted their work to the Personal Genomics Session. We are also indebted to the anonymous reviewers who contributed their time and expertise to evaluate the submitted papers.

IMPROVING THE PREDICTION OF PHARMACOGENES USING TEXT-DERIVED DRUG-GENE RELATIONSHIPS

YAEL GARTEN[§]
Stanford Biomedical Informatics Training Program,
Stanford University, Stanford, CA 94305, USA

NICHOLAS P TATONETTI[§]
Stanford Biomedical Informatics Training Program,
Stanford University, Stanford, CA 94305, USA

RUSS B ALTMAN[†]
Departments of Bioengineering & Genetics
Stanford University, Stanford, CA 94305, USA

A critical goal of pharmacogenomics research is to identify genes that can explain variation in drug response. We have previously reported a method that creates a genome-scale ranking of genes likely to interact with a drug. The algorithm uses information about drug structure and indications of use to rank the genes. Although the algorithm has good performance, its performance depends on a curated set of drug-gene relationships that is expensive to create and difficult to maintain. In this work, we assess the utility of text mining in extracting a network of drug-gene relationships automatically. This provides a valuable aggregate source of knowledge, subsequently used as input into the algorithm that ranks potential pharmacogenes. Using a drug-gene network created from sentence-level co-occurrence in the full text of scientific articles, we compared the performance to that of a network created by manual curation of those articles. Under a wide range of conditions, we show that a knowledge base derived from text-mining the literature performs as well as, and sometimes better than, a high-quality, manually curated knowledge base. We conclude that we can use relationships mined automatically from the literature as a knowledgebase for pharmacogenomics relationships. Additionally, when relationships are missed by text mining, our system can accurately extrapolate new relationships with 77.4% precision.

1. Introduction

Individuals have variable response to drug treatment[1,2]. The assumption underlying personalized medicine and pharmacogenetics is that an individual's genotype can be used to predict variable drug response[3]. Understanding and describing this variation is an essential first step of personalized medicine[2,4,5]. Pharmacogenomics investigates how genes and their variation impact drug response. Such research has historically been pharmaco*genetic*, focusing on smalls set of genes or proteins[6]. However, in this new age of high throughput technologies, the research has become increasingly pharmaco*genomic,* involving multiple genes. Pharmacogenomics (PGx) knowledge has expanded rapidly, as we uncover new connections between genes and the effects of their variants on drug response. Simply determining the genes that are important for drug response is a critical requirement. Recently, high throughput technologies such as genome wide association studies have yielded important new insights, however these technologies are plagued with high false positive rates, and statistical analysis of the data does not take advantage of existing biomedical knowledge[7,8]. Hansen et al. recently described an algorithm that uses existing knowledge in order to rank 12,460 genes in the genome on the basis of their potential relevance to a specific drug of interest[9]. This algorithm can prioritize genes in high throughput data sets, thus removing some false positives. The Hansen algorithm, called PGxPipeline, uses two knowledge bases of known drug-gene relationships, the Pharmacogenomics Knowledge Base (PharmGKB)[10] and DrugBank[11]. While these knowledge bases are extremely useful for pharmacogenomics they are also created manually by a staff of curators, who read the literature and annotate the PGx information. Thus, they are expensive to maintain and difficult to update, particularly as the volume of pharmacogenomic literature increases.

Therefore, there is a need for a scalable, inexpensive way to generate a comprehensive knowledge base of drug-gene relationships that can be used as input to the PGxPipeline algorithm. PharmGKB contained knowledge about

[§] First author, [†] Corresponding author, russ.altman@stanford.edu

404 drugs and 585 genes at the time of download, however the literature contains an order of magnitude more of both drugs and genes[10]. Automatic methods of monitoring this space are necessary.

Text mining techniques allow us to survey the literature in an automated fashion, and extract information from the unstructured scholarly literature[12], into a structured format in a database. As described by Hunter and Cohen, today's interdisciplinary research scientist has an increasingly overwhelming amount of literature to assimilate[13]. Only through efficient text mining techniques can the data in the literature be extracted and rendered most useful. We previously described Pharmspresso[14], which performs the task of extracting pharmacogenomic relationships from sentences. In this work we combined Pharmspresso and PGxPipeline to assess the suitability of automatically derived knowledge in training a gene-ranking algorithm. Thus, we can compare the performance of the text-mining-based knowledge source to the curation-based knowledge source utilized by Hansen, et al. If successful, we can contemplate using a continuously updated and expanded network of drug-gene relationships as the literature expands. This will clearly improve the results as Hansen et al. showed that the performance of the ranking algorithm depends critically on the size of the set of input drug-gene relationships[9]. Additionally, this work also serves as an external validation for the Pharmspresso automated text-mining algorithm, which, until now, has only been validated on a small set of relationships.

2. Methods

2.1. *Generation of corpus for text mining algorithm*

We used the QUOSA desktop application[15] to automatically download the full text PDFs of all articles that were manually curated by PharmGKB curators. At the time the PharmGKB relationships were extracted from the database, 2202 articles had been curated. Of these, 1731 articles had available full text and this set was used as our corpus.

2.2. *Generation of PharmGKB set of drug-gene relationships for training*

We extracted drug-gene relationships from the core tables of PharmGKB[10], for all 1731 articles for which we had full text. Of these articles, 964 contained drug-gene relationships (the remaining 767 contained drug-disease or gene-disease relationships). A total of 1782 unique drug-gene relationships are found in these 1731 articles. For articles that contain more than one gene or more than one drug we relate all possible combinations of genes and drugs.

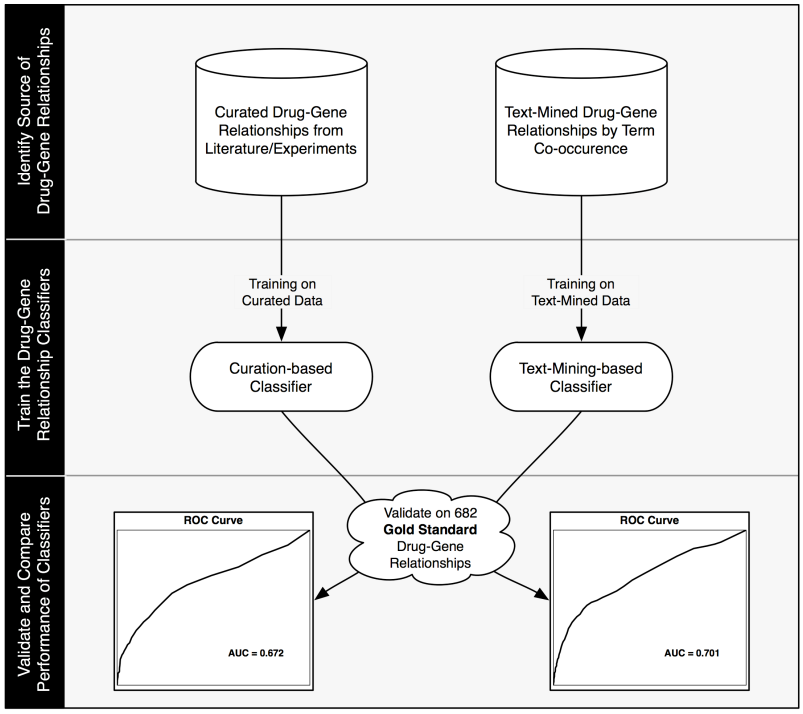


Figure 1. Methods overview: A knowledge base of drug-gene relationships is extracted from a curated source (PharmGKB and DrugBank) as well as from an automatic text-mining source (Pharmspresso). One classifier is trained using each of the two types of knowledge sources and then validated against a gold standard set of drug-gene relationships to allow comparison of the two sources.

2.3. *Generation of gold standard drug-gene relationships*

The PharmGKB staff curate the literature at the article level, annotating the genes and drugs discussed in the article. In order to obtain a gold standard of manually curated relationships at the single relationship level, we used only those articles that mention at most one gene and one drug, to ensure a direct pharmacogenomic relationship between gene and drug. There were 916 such articles, containing a total of 682 unique drug-gene relationships. This was used as our gold standard for evaluation. We use this validation set derived from PharmGKB data to compare the performance of a classifier trained on PharmGKB data and a classifier trained on text-mined data; it is important to note that this validation set is not included when training either classifier.

2.4. *Extraction of drug-gene relationships from text by Pharmspresso*

We used the Pharmspresso system described previously[14] to extract the drug-gene relationships from the corpus. Pharmspresso extracts all sentences that contain co-occurrences of a gene and a drug (see Figure 2). To allow direct comparison of performance of the PGxPipeline using the text-mining-based drug-gene network to the curation-based drug-gene network, we used only genes and drugs found in the PharmGKB database when running the Pharmspresso algorithm: a total of 585 genes and 404 drugs.

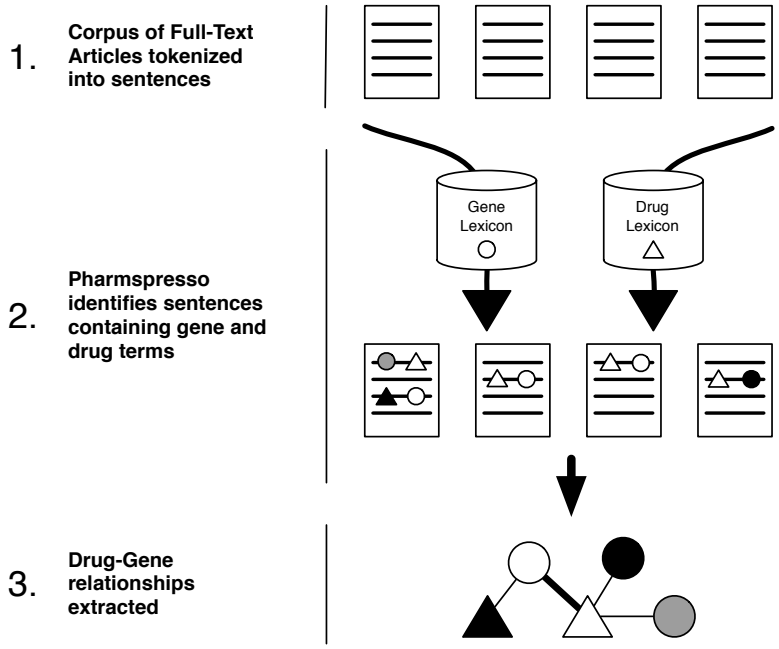


Figure 2. Description of Pharmspresso system for relationship extraction at the sentence level. A corpus of full-text articles is first tokenized into sentences. Pharmspresso then marks up the sentences by identifying terms associated with genes and drugs. A drug-gene network is then created by drawing edges between genes and drugs that co-occur at the sentence level. The width of the edge corresponds to the number of articles that support the relationship.

2.5. *Generation of scores for drug-gene relationships by PGxPipeline algorithm*

The PGxPipeline algorithm, as presented by Hansen et al., assigns scores to 12,460 genes representing their propensity to modulate drug response for a query drug. Figure 3 illustrates this method. Briefly, the algorithm derives the scores by using two knowledge bases, (1) a gene-gene interaction network and (2) a drug-gene relationship network. These two networks are combined to make a gene-gene-drug network. For a query drug, the PGxPipeline scores each gene by comparing the query drug to drugs connected to that gene in the gene-gene-drug network, and assigning a score based on this similarity. Drug similarity is measured by both structural similarity and similarity of indications. As described in Hansen et al. structural drug similarity is defined as the Tanimoto coefficient of 166 structural features. The Tanimoto

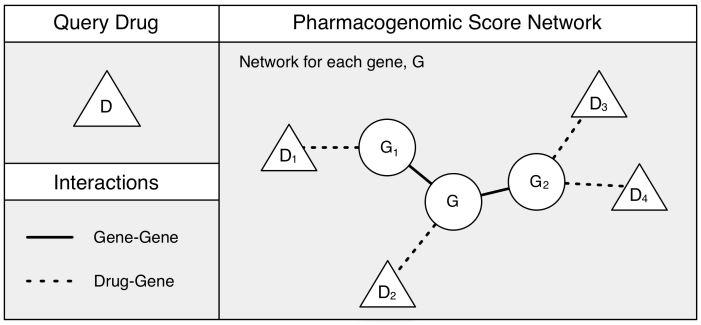


Figure 3. The Pharmacogenomics Pipeline. Given a drug, D, each gene in the genome is scored based on the similarity of the neighboring drugs to the query drug. A neighboring drug may interact directly with the gene (D2) or indirectly (D1, D3, D4) through neighboring genes (G1, G2).

coefficient is also used as a metric to compare the similarity of the indication sets of two drugs. We trained a logistic regression classifier on the input positive examples and random negative examples using these two types of features, structural similarity and indication set similarity. The more similar the drugs in the local network of the gene are to the query drug, the higher the score the gene will receive.

The original PGxPipeline[9] used PharmGKB as a source of "genetic" drug-gene relationships, DrugBank as a source of "physical" drug-gene interactions, and the InWeb interactome[16] as a source of gene-gene interactions. InWeb is a protein-protein interaction network created with data from experiments. In this work, we explored the use of drug-gene relationships mined from the literature, as an alternative knowledge source to the algorithm, in place of using the combination of relationships provided by the PharmGKB curation process and the physical interactions from DrugBank.

2.6. *Refining the source of negative relationships*

In order to provide negative relationships to the logistic classifier during training, the PGxPipeline matches each positive relationship from its gold standard with relationships between that same drug and three randomly selected genes. The set of genes it samples from is the entire set of genes in InWeb, PharmGKB, and DrugBank (12,460 genes). The PGxPipeline knowledge base contains approximately 400 drugs and 12,460 genes. However, only approximately 1,000 of those genes have relationships (genetic or physical) with drugs. Given a gene chosen at random from the set 12,460 genes, the chance that the gene will have any drug relationships is quite low. The consequence of this is that it can be relatively easy to differentiate between a positive and negative relationship, since most negative examples have no important relationships with any drugs.

In order to make our classification task more challenging we select negative examples from among known pharmacogenes—genes that in fact have at least some known relationship with a drug. Therefore, we replaced the pool of genes from which negative examples are selected with only the set of genes that exists in PharmGKB (585 genes), a much smaller gene set of known pharmacogenes. This allows us to more stringently evaluate the classifier while still maintaining the power to predict potential drug relationships with unknown pharmacogenes.

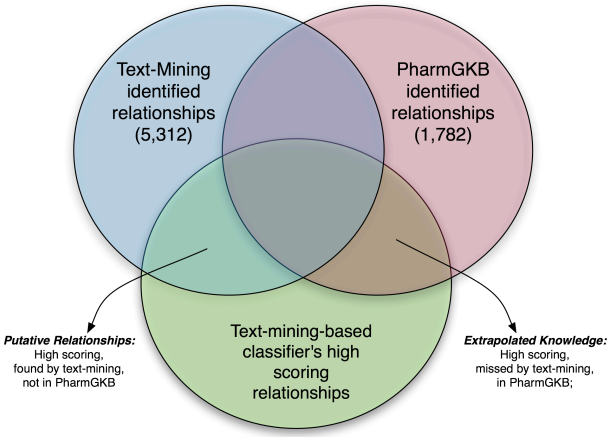


Figure 4. The intersection of drug-gene interactions identified by Pharmspresso text-mining or by PharmGKB curators, and those interactions receiving high scores when applying the text-mining-based classifier. Pharmspresso identified 5,312 pharmacogenomic interactions, PharmGKB contained 1782 interactions, with an overlap of 1,157 between the two sources.

2.7. *Comparison of the two drug-gene knowledge sources: Curated versus Text-Mined*

To facilitate comparison between the text-mining-based classifier and the curation-based classifier we trained a logistic regression classifier in a similar manner, and validated with fivefold cross-validation. Therefore, in each of the folds, all knowledge about the relationships in the validation set (1/5 of the data) is dropped from the training set (the 4/5 of the data used for training). The performance of each classifier on this task is a metric of how accurate the model is in classifying known pharmacogenetic relationships.

2.8. *Using text-mining-derived relationships combined with PGxPipeline scores to extrapolate; discovering additional drug-gene relationships*

Pharmspresso identified 5,312 pharmacogenomic relationships, PharmGKB contained 1,782 relationships, with an overlap of 1,157 between the two sources (Figure 4). As expected and previously described[14], Pharmspresso is a very sensitive test for pharmacogenomic relationships, while PharmGKB is a highly specific one. There are 625 relationships in PharmGKB that Pharmspresso does not identify when searching for co-occurrence at the sentence level (the lexical names of the gene and drug may not occur in the same sentence). To test whether we can use the PGxPipeline scores to recognize true relationships not directly found in literature by Pharmspresso, we did the following: We trained the classifier with the 5,132 drug-gene relationships found by text-mining and applied the classifier to all of the 625 drug-gene relationships in PharmGKB that were not found by text-mining, to get a pharmacogene score for each relationship. For comparison we also applied the classifier to a randomly generated set of drug-gene relationships to get a pharmacogene score for each relationship. We then investigated our ability to use the pharmacogene score to distinguish between the relationships that were in PharmGKB versus the randomly created relationships. (To find relationships in region titled "Extrapolated Knowledge" in Figure 4.)

2.9. *External validation: New relationships registered by the PharmGKB staff*

During the time since we first downloaded the pharmacogenomic relationships from PharmGKB an additional 1,462 articles were curated resulting in an additional 1,636 drug-gene relationships. This set of relationships was used as an external validation set. For each drug-gene relationship we used the trained classifier to score the relationship and randomly sampled three more genes to pair with the drug as a source of negative relationships.

3. Results

3.1. *Comparison of the two drug-gene knowledge sources: Curated versus Text-Mined*

To evaluate the use of a text-mining based network as a pharmacogenomic relationship knowledge base we compared the performance of the text-mining-based classifier with that of the curation-based classifier (Methods 2.7) using 5-fold cross validation on the gold standard set of drug-gene relationships (Methods 2.3). We find that the text-mining-based classifier out-performs the curation-based classifier, with receiver operator characteristic (ROC) curves with area under the curve (AUC) of 0.701 and 0.672 respectively (see Figure 5). Besides having an overall AUC that is slightly higher, the text-mining-based classifier achieves high sensitivity in the region of high specificity (FPR $\leq$ 0.2). Achieving a greater AUC in this area alone is often desirable by experimentalists as the algorithm can ensure a very low false positive rate, even though it may not have high recall. In addition we tested the two classifiers under the exact conditions described by Hansen (negative set genes selected from InWeb and broader definition of gold standard from PharmGKB data). This yields ROC curves with AUC values of 0.814 and 0.799 for the curation-based classifier and the text-mining based classifier respectively, and so under those conditions the performance of the two classifiers is comparable. The 0.814 AUC of the curation-based classifier is slightly

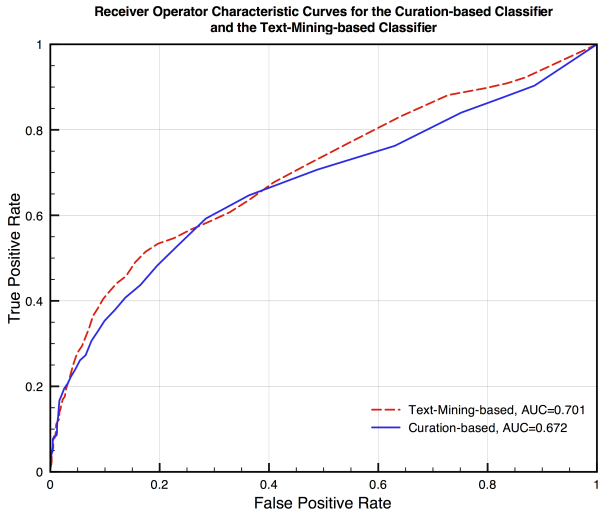


Figure 5. The ROC curves for the curation-based classifier and text-mining-based classifier validated on the gold standard. The text-mining-based classifier out-performs the curation-based classifier.

lower than the 0.82 as reported by Hansen et al., presumably because the input knowledgebase is smaller—it is based on the subset of 1731 articles for which we obtained full text to allow fair comparison with Pharmspresso.

3.2. *Using text-mining-derived relationships combined with PGxPipeline scores to extrapolate; discovering additional drug-gene relationships*

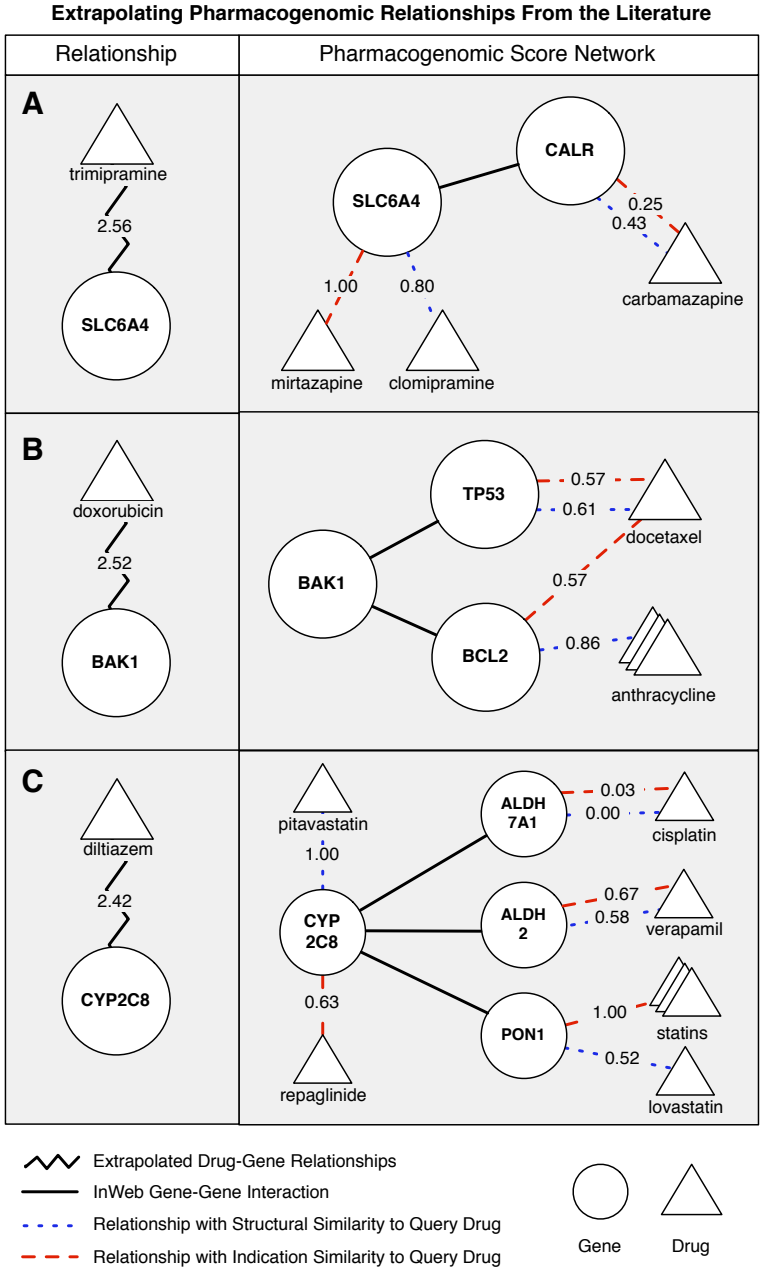


Figure 6. Examples of extrapolation of drug-gene interactions using the text-mining-based PGxPipeline classifier. All examples are in fact found in PharmGKB; meaning there is literature support for these relationships recorded manually by curators. Although Pharmspresso misses them, they are recovered by the PGxPipeline scoring mechanism. Zigzag line: suggested, positive-score interaction (left panel). This interaction is not found directly in the literature by Pharmspresso, but receives a positive PGxPipeline score (score appears on the line). Solid lines: gene-gene relationships from InWeb. Dashed/dotted lines: drug-gene relationship found in literature by Pharmspresso. Dashed red - indication similarity, Dotted blue - structural similarity. The score shown on the edge represents the similarity score of the edge's drug, to the query drug.

We observe that there are relationships in PharmGKB that Pharmspresso does not discover when searching for co-occurrence at the sentence level and thus not used to train the text-mining-based classifier. We explored whether we can detect these relationships by using the scores assigned by the classifier. As described in Section 2.8, we selected a balanced set of positive and negative relationships and tested which relationships lie in the region titled "Extrapolated Knowledge" in Figure 4 (relationships positively scored by the classifier, not found by text-mining, and in PharmGKB). We call these "extrapolated" since they were not identified by the text-mining algorithm and so not part of the input knowledge base of drug-gene relationships. We validated against the PharmGKB relationships and found 3.44-fold enrichment (134/39) with a cutoff score of zero. That is, of the set of 173 relationships that score positively, we have 134 true positives that are in PharmGKB and presumably 39 false positives, a false discovery rate (FDR) of 22.5%.

Figure 6 describes the contribution of the local network to the score for a given pharmacogenomic relationship extrapolated from the text-mining-derived relationships by the text-mining-based classifier, for three examples. These examples represent a known pharmacogenetic relationship (they appear in PharmGKB) where the drug and gene did not co-occur at the sentence level in the literature, yet the text-mining-based classifier assigns the relationship a high score.

Figure 6A shows the underlying evidence for the predicted relationship between the drug trimipramine and gene

SLC6A4, based on the text-mining-derived network of drug-gene relationships, and the gene-gene interactions in InWeb. SLC6A4 is a sodium-dependent serotonin transporter. The drug trimipramine is a tricyclic antidepressant[17]. The support for the predicted relationship between SLC6A4 and trimipramine stems from the similarity to the two drugs directly related to the gene, in the text-mining-derived network (mirtazapine and clomipramine), and from the similarity to the drug carbamazapine, which is related in the text-mining-derived network to the CALR gene, found to interact with SLC6A4 in the InWeb network. Mirtazapine is an antidepressant used for the treatment of moderate to severe depression, and has a very similar set of indications as trimipramine [18]. Of the drugs that co-occur in the literature with SLC6A4, the one that is most similar in structure to trimipramine is clomipramine. Of the drugs related to the CALR gene in the text-mining-derived network, the one that is most structurally similar to trimipramine, the query drug, is carbamazapine. It is also the most similar in its indications: both carbamazapine and trimipramine are used to treat depression, the indirect connection to carbamazapine via CALR boosts the prediction[10].

Figure 6B shows the support for the relationship between doxorubicin and BAK1, a BCL2-antagonist/killer 1. The InWeb interactome connects BAK1 to two genes, TP53 and BCL2, each of which has a literature co-mention with docetaxel, an anti-mitotic chemotherapy medication. Both doxorubicin and docetaxel are cancer treatments as well, and so the similarity of indications plays a role in uncovering the relationship between BAK1 and doxorubicin. Doxorubicin is a type of anthracycline, which is the most active group of cytotoxic agents for the treatment of breast cancer. Docetaxel with anthracyclines are sometimes used together and share structural similarity[19]. BAK1 "borrows" drug relationships from its neighbors to boost the likelihood of sharing a relationship with doxorubicin.

Figure 6C shows the predicted relationship between diltiazem and CYP2C8. Diltiazem is a calcium channel blocker, a member of the benzothiazepine class that reduce blood pressure through vasodilation[20]. It is used to treat hypertension and rhabdomyolysis, as is verapamil. InWeb connects CYP2C8 to 3 genes: ALDH7A1, ALDH2, and PON1. Each of these interact with drugs that have substantial structural or indication overlap with diltiazem[10]. CYP2C8 itself is found in the literature with 2 drugs; pitavastatin has the highest indications similarity to diltiazem and repaglinide has the highest structural similarity to diltiazem.

3.3. *External Validation of the text-mining-based pharmacogene classifier*

As an external validation of the text-mining-based classifier 1,636 drug-gene relationships added to the PharmGKB after we established the training set as well as three times that many randomly chosen drug-gene relationships were scored. The ROC curve has an area under the curve of 0.78, as shown in Figure 7. The text-mining-based classifier had comparable performance to the curation-based classifier on the same external validation set, which produced a ROC curve with an AUC of 0.8.

There are relationships found by Pharmspresso that do not appear in PharmGKB. We scored each of these relationships by leaving out the knowledge about the relationship during training of the text-mining based classifier. The relationships that receive positive scores appear in the intersection of the green and blue

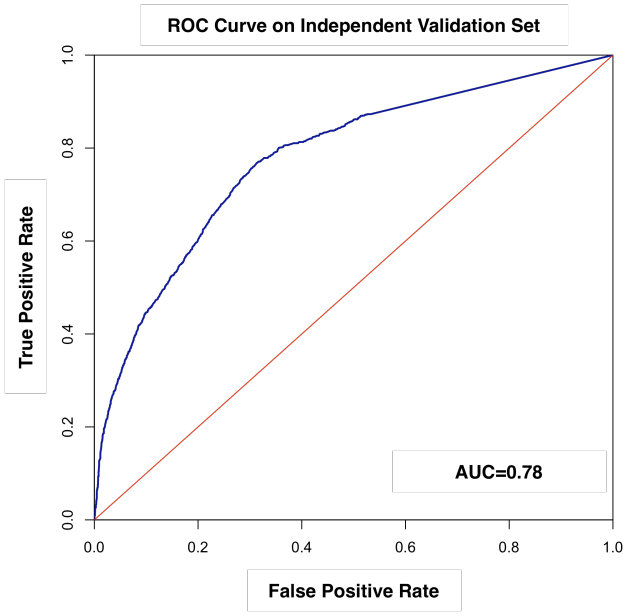


Figure 7. The ROC curve for the literature-based classifier on the external validation set of 1,636 drug-gene interactions not included in the training set. This performance was achieved under the same conditions as presented in the Hansen paper.

circles of Figure 4 – that is, the intersection of regions "Text-mining relationships" and "Text-mining based high scoring relationships". Within this intersection, those that do not appear in PharmGKB are titled "Putative Relationships" and are sent to the curators for potential insertion into PharmGKB. For example, the relationship between *CYP3A5* and cyclosporine scored highly, was not in the original set of PharmGKB relationships (thus appears in region titled "Putative Relationships" in Figure 4), and in fact PharmGKB now has three articles supporting this relationship [21-23].

4. Discussion

In this work, we explored the use of a text-mining-derived network of drug-gene relationships, as a knowledge base to replace human-curated literature relationships in the PGxPipeline. The PGxPipeline uses the knowledge base to predict pharmacogenes for an input query drug. While the human curated data are high quality, they are much less abundant. In this application, it is apparent that the improved coverage afforded by automatic detection outweighs the introduction of noise and errors because of imperfect text-mining extraction. Of course, the benefits are substantial: curation is a very expensive process (in terms of time and money), whereas text-mining is inexpensive and scalable[14]. In addition, PharmGKB staff curators often only read the abstract of articles because of the large volume of papers they must annotate. Abstracts do not necessarily contain the pharmacogenomic drug-gene relationship reported by the article, whereas the Pharmspresso system analyzes the full text of an article.

The task of Pharmspresso is really to identify relationships between genes and drugs, within the small scope of a single sentence. This is not what PharmGKB curators have been tasked with; they curate articles with respect to the genes and drugs that are mentioned without specifically asserting which genes and drugs relate. Therefore, it is not surprising that when using the high-quality gold standard the text-mining-based classifier actually performs slightly better than the curation-based classifer (0.701 AUC vs. 0.672 respectively, Figure 5). These results demonstrate that the drug-gene network derived by Pharmspresso can be used in place of manually curated data in the PGxPipeline algorithm, which may allows us to enlarge the drug-gene network to millions of articles in the scientific literature. Our results also provide an independent, large-scale, external validation of the usefulness and accuracy of Pharmspresso. Pharmspresso had previously been validated on a small evaluation set. Finally, we have demonstrated that the scores assigned by the PGxPipeline can be used to detect new relationships.

The Hansen et al. algorithm relied on a manually-curated network of pharmacogenomic drug-gene relationships derived from experimental or clinical data, as reported in the literature. Our results show, however, that a text-mined sentence co-occurrence drug-gene network can perform as well and even better under some circumstances. We acknowledge that the co-occurrence drug-gene network contains noise. Nonetheless, it is more likely that a co-occurrence is a meaningful relationship than a random one. Text mining allows us to generate a large network of relationships, thereby increasing the signal-to-noise ratio. This implies that the likelihood of a pharmacogenomic drug-gene relationship increases in proportion to the number of similar drugs that co-occur in the literature with the gene or genes in its pathway. We can therefore expect that as our knowledge base increases by mining more pharmacogenomic articles, so will our power to predict pharmacogenomic relationships. Additionally, because the method is general, this logic may apply to other types of pharmacogenomic relationships such as drug-SNP relationships, a hypothesis which we are currently investigating.

4.1. *Limitations*

While we have shown that we can predict pharmacogenomic drug-gene relationships based on a corpus of pharmacogenomic articles, it not yet clear that this methodology will work for other interesting biological problems, such as deriving drug-SNP relationships or gene regulatory networks. The generality of our methodology may be limited since our analysis is based on a corpus that is highly enriched for pharmacogenomics articles. We plan to

investigate how dependent the performance of the algorithm is on this specialized corpus. One other limitation of using simple co-occurrence is the inability to derive the type of drug-gene relationship text-mined from the literature[24]. For example, it would be advantageous to know if a drug and gene have a positive or negative relationship or whether the gene is pharmacokinetic or pharmacodynamic for the drug. This type of characterization of edges in the network requires more sophisticated text mining.

Our next step is to expand the corpus of articles available to Pharmspresso to include a larger pharmacogenomics literature. Methods that classify publications likely to contain pharmacogenomic information, such as MScanner, can be used to filter Medline in order to identify pharmacogenomic articles[25]. This expansion of the drug-gene relationship network will greatly improve the performance of the PGxPipeline. The PGxPipeline relies on other types of relationship networks in addition to the drug-gene network, namely a gene-gene network and a drug-disease network. Mining these relationships from the literature may also increase the predictive power of the algorithm as well as keep the knowledge base scalable and up-to-date. Finally, we plan to incorporate high-scoring predictions into the curation pipeline at PharmGKB, to prioritize these predictions for curator review and subsequent insertion into the knowledge base.

Pharmacogenomics is not only concerned with the important genes but also with their particular variations that impact drug response. For example, variations in the *VKORC1* and *CYP2C6* genes are critical for determining warfarin dose, and can be used to predict the optimal dose of warfarin[26-29]. The Pharmspresso algorithm can detect genetic variations, and can be used to create a network linking specific variations to specific drugs[14]. Such a network might be useful in refining the PGxPipeline to weight pharmacogene predictions based on this additional knowledge source. The text-mining-based PGxPipeline classifier produced a substantial number of high scoring drug-gene relationships that were not found to be in PharmGKB. A high-throughput biological assay could be employed to test these relationships for their validity.

Author Contributions

YG and NPT designed the study and carried out the analysis. YG wrote the manuscript. NPT contributed to the manuscript. RBA provided critical guidance on the study design, interpretation of results, and preparation of the manuscript.

Acknowledgements

YG and NPT are supported by the Graduate Training in Biomedical Informatics grant (T15 LM007033) from the National Library of Medicine. YG and RBA are supported by NIH/NIGMS Pharmacogenetics Research Network and Database and the PharmGKB resource (NIH U01GM61374). The authors would like to thank R. Whaley for PharmGKB data. We thank our anonymous reviewers for their constructive comments.

References

1. Evans & Mcleod. Pharmacogenomics -- Drug Disposition, Drug Targets, and Side Effects. *N Engl J Med* **348**, 538 (2003).
2. Swen et al. Translating Pharmacogenomics: Challenges on the Road to the Clinic. *PLoS Med* **4**, e209 (2007).
3. Weiss et al. Creating and evaluating genetic tests predictive of drug response. *Nature reviews Drug discovery* **7**, 568 (2008).
4. Davis et al. The microeconomics of personalized medicine: today's challenge and tomorrow's promise. *Nature reviews Drug discovery* **8**, 279 (2009).
5. Kirchheiner et al. Pharmacogenetics-based therapeutic recommendations - ready for clinical practice? *Nature reviews Drug discovery* **4**, 639 (2005).
6. Goldstein et al. Pharmacogenetics goes genomic. *Nat Rev Genet* **4**, 937 (2003).
7. Dollery, C.T. Beyond genomics. *Clin Pharmacol Ther* **82**, 366-70 (2007).

10

8. McCarthy, M.I. et al. Genome-wide association studies for complex traits: consensus, uncertainty and challenges. *Nat Rev Genet* **9**, 356-69 (2008).
9. Hansen, N.T., Brunak, S. & Altman, R.B. Generating genome-scale candidate gene lists for pharmacogenomics. *Clin Pharmacol Ther* **86**, 183-9 (2009).
10. Klein, T.E. et al. Integrating genotype and phenotype information: an overview of the PharmGKB project. Pharmacogenetics Research Network and Knowledge Base. *The Pharmacogenomics Journal* **1**, 167-70 (2001).
11. Wishart, D.S. et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Research* **36**, D901-6 (2008).
12. Müller, H.M., Kenny, E.E. & Sternberg, P.W. Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol* **2**, e309 (2004).
13. Hunter, L. & Cohen, K.B. Biomedical language processing: what's beyond PubMed? *Mol Cell* **21**, 589-94 (2006).
14. Garten, Y. & Altman, R.B. Pharmspresso: a text mining tool for extraction of pharmacogenomic concepts and relationships from full text. *BMC Bioinformatics* **10 Suppl 2**, S6 (2009).
15. Quosa | The total solution for efficiently managing and monitoring scientific literature, http://www.quosa.org. (2009).
16. Lage, K. et al. A human phenome-interactome network of protein complexes implicated in genetic disorders. *Nat Biotechnol* **25**, 309-16 (2007).
17. Hébert, C., Habimana, A., Elie, R. & Reader, T.A. Effects of chronic antidepressant treatments on 5-HT and NA transporters in rat brain: an autoradiographic study. *Neurochem Int* **38**, 63-74 (2001).
18. Marcus, S.C., Hassan, M. & Olfson, M. Antidepressant switching among adherent patients treated for depression. *Psychiatric services (Washington, DC)* **60**, 617-23 (2009).
19. Jones, S. et al. Docetaxel With Cyclophosphamide Is Associated With an Overall Survival Benefit Compared With Doxorubicin and Cyclophosphamide: 7-Year Follow-Up of US Oncology Research Trial 9735. *J Clin Oncol* **27**, 1177-83 (2009).
20. Neuvonen, P.J., Niemi, M. & Backman, J.T. Drug interactions with lipid-lowering drugs: mechanisms and clinical relevance. *Clin Pharmacol Ther* **80**, 565-81 (2006).
21. Kreutz, R. et al. CYP3A5 genotype is associated with longer patient survival after kidney transplantation and long-term treatment with cyclosporine. *The Pharmacogenomics Journal* **8**, 416-22 (2008).
22. Kreutz, R. et al. The effect of variable CYP3A5 expression on cyclosporine dosing, blood pressure and long-term graft survival in renal transplant patients. *Pharmacogenetics* **14**, 665-71 (2004).
23. Fanta, S. et al. Pharmacogenetics of cyclosporine in children suggests an age-dependent influence of ABCB1 polymorphisms. *Pharmacogenet Genomics* **18**, 77-90 (2008).
24. Ahlers, C.B., Fiszman, M., Demner-Fushman, D., Lang, F.-M. & Rindflesch, T.C. Extracting semantic predications from Medline citations for pharmacogenomics. *Pacific Symposium on Biocomputing Pacific Symposium on Biocomputing*, 209-20 (2007).
25. Poulter, G.L., Rubin, D.L., Altman, R.B. & Seoighe, C. MScanner: a classifier for retrieving Medline citations. *BMC Bioinformatics* (2008).
26. Rieder et al. Effect of VKORC1 Haplotypes on Transcriptional Regulation and Warfarin Dose. *N Engl J Med* **352**, 2285 (2005).
27. Takeuchi et al. A genome-wide association study confirms VKORC1, CYP2C9, and CYP4F2 as principal genetic determinants of warfarin dose. *PLoS Genet* **5**, e1000433 (2009).
28. Rajanayagam & Rajanayagam. Pharmacogenetics: Optimizing warfarin therapy. *Nature Reviews Cardiology* **6**, 324 (2009).
29. Cooper, G.M. et al. A genome-wide scan for common genetic variants with a large influence on warfarin maintenance dose. *Blood* **112**, 1022-7 (2008).

FINDING UNIQUE FILTER SETS IN PLATO: A PRECURSOR TO EFFICIENT INTERACTION ANALYSIS IN GWAS DATA

BENJAMIN J. GRADY, ERIC TORSTENSON, SCOTT M. DUDEK, JUSTIN GILES, DAVID SEXTON, AND MARYLYN D. RITCHIE[†]

Center for Human Genetics Research, Department of Molecular Physiology & Biophysics, Vanderbilt University Nashville, TN 37232, United States

The methods to detect gene-gene interactions between variants in genome-wide association study (GWAS) datasets have not been well developed thus far. PLATO, the Platform for the Analysis, Translation and Organization of large-scale data, is a filter-based method bringing together many analytical methods simultaneously in an effort to solve this problem. PLATO filters a large, genomic dataset down to a subset of genetic variants, which may be useful for interaction analysis. As a precursor to the use of PLATO for the detection of gene-gene interactions, the implementation of a variety of single locus filters was completed and evaluated as a proof of concept. To streamline PLATO for efficient epistasis analysis, we determined which of 24 analytical filters produced redundant results. Using a kappa score to identify agreement between filters, we grouped the analytical filters into 4 filter classes; thus all further analyses employed four filters. We then tested the MAX statistic put forth by Sladek et al. [1] in simulated data exploring a number of genetic models of modest effect size. To find the MAX statistic, the four filters were run on each SNP in each dataset and the smallest p-value among the four results was taken as the final result. Permutation testing was performed to empirically determine the p-value. The power of the MAX statistic to detect each of the simulated effects was determined in addition to the Type 1 error and false positive rates. The results of this simulation study demonstrates that PLATO using the four filters incorporating the MAX statistic has higher power on average to find multiple types of effects and a lower false positive rate than any of the individual filters alone. In the future we will extend PLATO with the MAX statistic to interaction analyses for large-scale genomic datasets.

1. Introduction

1.1. *Dissecting the Genetic Architecture of Complex Traits in GWAS*

In the quest for disease susceptibility genes, genome-wide association studies (GWAS) have become the standard approach utilized by many investigators, with the promise of finding genes. Innovation is needed for the analysis and interpretation of GWAS data, as we are headed for a calamity. In the past, there have been problems replicating single-locus candidate genes studies[2]. Soon, we will be faced with many 500K and 1M (1-million) SNP datasets with failure to replicate, as well as a flood of publicly available data that could be a gold mine, if the appropriate analysis strategy is utilized. Currently, no single analytical method will allow us to extract all information from a GWAS; in fact, no single method *can* be optimal for all datasets, especially when the genetic architecture for a given disease is not well understood. Therefore, an integrative platform is needed to accommodate the multitude of sophisticated analytical methods being developed in the field for analysis as we learn more about the genetic architecture as well as which methodologies are successful for GWAS analyses. As a resolution to this crisis, we have developed a system, the Platform for the Analysis, Translation, and Organization of Large Scale data (**PLATO)**, for the analysis of GWAS data that will incorporate numerous analytic approaches as filters. The use of multiple filters that can be used in a modular way will allow a flexible analytical strategy that can be tailored to each investigation. In particular, these filters will be critical in the search for complex interactions among genes and/or the environment. It is already feasible to search all individual effects on even 1M single nucleotide polymorphisms (SNPs); however, once interactions between SNPs are considered, the problem becomes much less tractable. Most GWAS contain at least 500,000 or 1M SNPs and sometimes include environmental or clinical factors. Many common diseases are believed to be multifactorial, having multiple genetic and/or environmental disease susceptibility factors that may or may not have statistically detectable main effects [3,4]. Interaction effects have been discovered as influential in conditions such as hypertension [5], Hirschsprung's disease [6], and cystic fibrosis[7,8]; examples such as these demonstrate the importance of considering interactions during analysis of GWAS data. Efficiently exploring the search space when considering interactions in GWAS data becomes challenging very quickly, considering that looking for an interaction between just two variables among 500,000 requires the analysis of about 1.25×10^{11} models. It is nearly infeasible to exhaustively search a space that large, much less the space that

results from searching for interactions between three or more variables out of 500,000 or 1M SNPs. Since exhaustive approaches are intractable, alternative strategies must be employed.

1.2. *PLATO*

PLATO is a computational framework that analyzes SNPs and other independent variables using a variety of filters in an effort to identify a subset of interesting SNPs from a much larger set. A filter in this case is defined as an analytical method or knowledge-based approach which mediates a reduction in the number of SNPs to a smaller subset. PLATO allows the flexibility of applying filters in series, parallel, or individually and also allows the specification of filters for different disease models (additive, dominant, etc). Furthermore, PLATO is extensible, allowing users to easily implement their own analytical methods as filters using a modular C++ library. By narrowing down the number of SNPs using various filters, looking for interactions between the remaining variables may be feasible.

An important consideration when applying multiple analytical filters to a dataset is the potential for redundancy among the filters. It is well known that many analytical methods are similar and follow the same underlying principles. Still, in many studies, several similar methods are often used and the results compared. Within PLATO, many of the filters are highly correlated; however, the different filters are options for analysis to accommodate user preferences. Since some of the filters are correlated, it is not necessary to analyze datasets with all of them. By grouping filters into classes according to their tendency to identify overlapping subsets of putatively important SNPs, and subsequently running filters from these distinct classes, it may be possible to remove the most SNPs with the fewest number of filters, and subsequently reduce computation time. It is also possible that by running multiple distinct filters, "noise" SNPs can be removed and the truly significant effects can be found by singling out SNPs that repeatedly appear highly ranked across multiple filters.

To determine which of the PLATO filters yield unique results, a simulation study was performed. Simulations, where the true location and size of the genetic effect are known, prove indispensible for evaluating new analytical techniques. Genomic data with a known effect was simulated, specifying disease prevalence and a disease variant. The resulting data was then analyzed using all twenty-four PLATO filters individually. A kappa statistic was used as a measure of comparison to provide a mechanism for grouping filters into subsets that yield similar results. One filter from each resulting group was chosen as a representative filter for the group based on ease of use and interpretation. These filter sets were then further subset into filter classes by their tendency to rank embedded genetic effects similarly. Once a set of filter classes had been determined, we implemented a MAX statistic in an additional simulation. Here, one filter from each of the four filter classes was performed on the simulated data for each SNP in the dataset, taking the lowest p-value among the four tests for each SNP. Permutation was then performed on the entire analysis procedure to create an empirical null distribution and the results were compared with those found from running the four filters individually. The PLATO approach utilizing the MAX statistic (PLATO_MAX) out-performed all of the individual filters alone and demonstrates promise for future applications to multiple types of analyses, in particular the search for epistasis.

2. Methods

2.1. *Data Simulation*

The genomeSIMLA software [9,10] was used to conduct the data simulations. Simulation was performed by first generating a population of 100,000 chromosomes containing 1000 bi-allelic polymorphisms. For each chromosome, all polymorphisms with exception of the disease polymorphism(s) were initialized randomly with respect to allele frequency within a range of minor allele frequency between 10% and 50%. We conducted a two-stage simulation study. In the initial phase of the simulation study, the goal was to determine the redundancy among the PLATO filters. For these simulations, the disease minor allele frequency was fixed at 25%. In the second phase simulations, the goal was to evaluate the approach whereby the correlated filters were clustered into filter classes and the MAX statistic was evaluated. Here, the disease polymorphisms were allowed to vary freely in allele

frequency. Once the population of chromosomes was initialized, a penetrance function describing the size of the disease effect and the location of the disease locus was applied and random sampling theory was utilized in order to choose datasets of 1000 cases and 1000 controls. In all simulations for the estimation of power, 100 datasets were used; however, to test the Type 1 error rate of the PLATO_MAX approach, 1000 datasets were simulated.

A number of different disease models were simulated for the different elements of this study. Table 1 lists the different genetic models simulated. First, to determine the agreement between filters, single-locus additive, dominant, and recessive genetic effects with an odds ratio of 1.2, 1.5, 1.8, and 2.0 were simulated independently. In addition, a null model with no genetic effect was simulated separately. The simulated data used to further subset these filters into filter classes included six genetic effects: 2 each of additive, dominant, and recessive effects with one effect of each pair having an odds ratio of 1.2 and the other an odds ratio of 1.5. Finally, the data simulated to test the PLATO_MAX approach was evaluated using three effects, each exhibiting an odds ratio of 1.5 under additive, dominant, and recessive models.

Table 1. Simulation Design

	Model type (effect size)			
Experiment	Additive	Dominant	Recessive	Null
Agreement between filters (Kappa comparison)	OR = 1,2, 1.5, 1.8, 2.0	OR = 1,2, 1.5, 1.8, 2.0	OR = 1,2, 1.5, 1.8, 2.0	X
Creation of filter classes	OR = 1.2 and 1.5	OR = 1.2 and 1.5	OR = 1.2 and 1.5	N/A
PLATO_MAX Power analysis	OR = 1.5	OR = 1.5	OR = 1.5	N/A
Type I error analysis (null data)	N/A	N/A	N/A	X

2.2. *PLATO Filters*

The datasets generated were analyzed individually using each of 24 different filters making up the comprehensive set of filters currently available for PLATO (Table 2). There are several of these filters that are subsets of filters with different data encodings. For example, the LIKELIHOODRATIO (G) filter is a LIKELIHOODRATIO filter that uses a genotypic data encoding while LIKELIHOODRATIO (A) is a LIKELIHOODRATIO filter that uses an allelic data encoding. Each filter type is summarized below, including how it functions as well as the meaning of different data encodings. The contingency table analytical methods utilized by the filters are further illustrated in Figure 1.

Table 2.Filters implemented in current PLATO analyses. A-Allelic; G-Genotypic; ADD-Additive; D-Dominant; R-Recessive.

ARMITAGE (ADD)	MDR
ARMITAGE (G)	NMI (A)
CHISQUARE (A)	NMI (ADD)
CHISQUARE (G)	NMI (D)
LIKELIHOODRATIO (A)	NMI (G)
LIKELIHOODRATIO (ADD)	NMI (R)
LIKELIHOODRATIO (D)	ODDSRATIO
LIKELIHOODRATIO (G)	UNCERTAINTYCOEFF (A)
LIKELIHOODRATIO (R)	UNCERTAINTYCOEFF (ADD)
LOGISTICREGRESS (ADD)	UNCERTAINTYCOEFF (D)
LOGISTICREGRESS (D)	UNCERTAINTYCOEFF (G)
LOGISTICREGRESS (R)	UNCERTAINTYCOEFF (R)

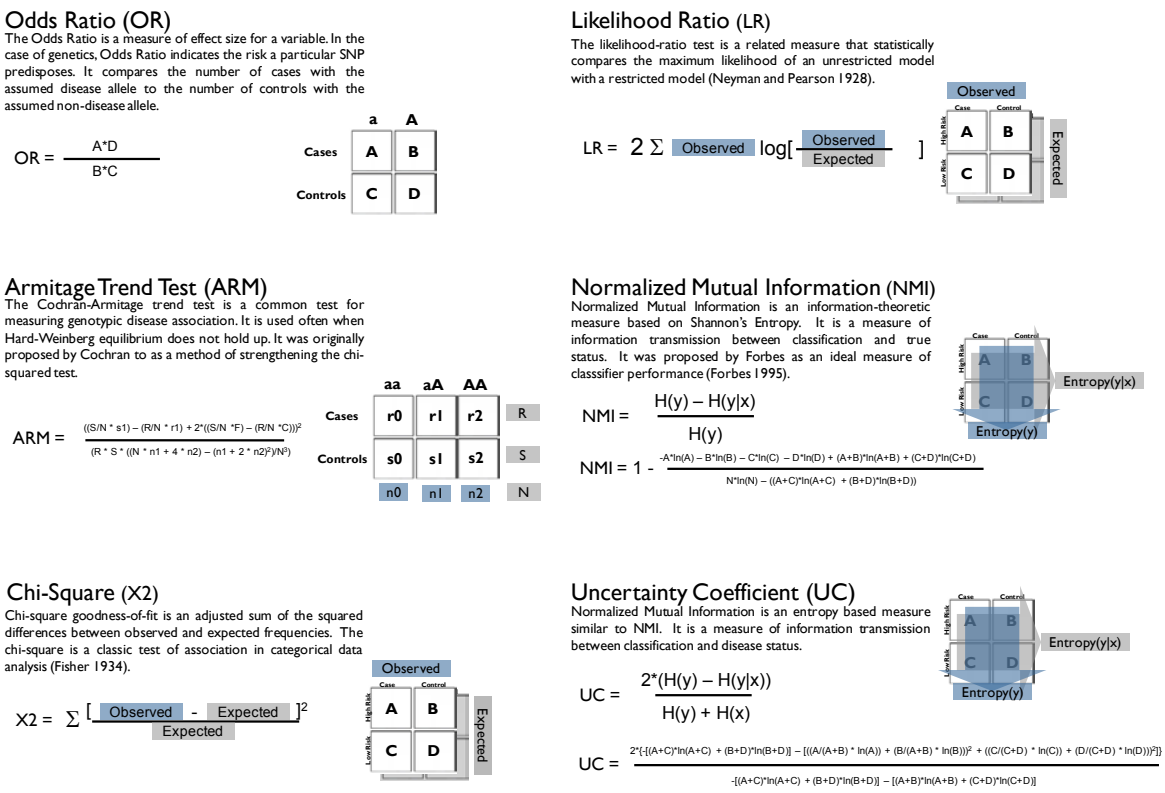


Figure 1. Six contingency table based filters used in PLATO. Adapted from (Bush, 2008)[11].

The ODDSRATIO filter utilizes a 2x2 table with the minor and major allele types as the columns and case and control status as the rows. To calculate the statistic, the product of cases with the minor allele (A) and controls with the major allele (D) is divided by the product of the cases with the major allele (B) over the controls with the minor allele (C). The ARMITAGE filter uses a Cochran-Armitage trend test to find the probability that a particular genotype is disease-associated by fitting the case-control distribution to a linear predictor equation of the form $p_i = a + Bx_i$ where *i* is the genotype being tested and *B* is the effect being attributed to the genotype [12-15]. Statistically speaking, testing disease association is looking for rejection of the null hypothesis that *B*=0. The CHISQUARE filter uses a chi-square [12] test to look for differences between observed and expected numbers of cases and controls for each genotype. LIKELIHOODRATIO filters use an analytical method that is very similar to the CHISQUARE filters. The difference between these methods is that in calculating the statistic, a log ratio of the difference between observed and expected is used as opposed to the squared deviation from expected [12]. The NMI and UNCERTAINTYCOEFFICIENT filters are quite similar, both functioning on the entropy in the data [16]. They examine the amount of information any particular genotype provides about the disease status. The main difference is that NMI (Normalized Mutual Information) is a normalized measure, as reflected in the name [12,17].

The LOGISTICREGRESS filters are one of the few types of filters - along with the Multifactor Dimensionality Reduction (MDR) filter - which do not use a contingency table measure to calculate the statistic used for comparison. LOGISTICREGRESS refers to logistic regression analysis. Logistic regression is a standard method used by epidemiologists when looking for disease association with both genetic and environmental factors [18]. Logistic regression uses a logistic equation to fit the pattern of cases and controls with respect to genotype and then determines if the genotype classes are predictive of disease. This equation is of the form $p(x)=\exp(a + Bx)/(1 + \exp(a + Bx))$, where $p(x)$ is the probability of getting the disease and *B* is the coefficient describing the effect of the genotype *x* [12]. MDR is an analytical method initially developed to analyze interactions between variables such as SNPs involved in disease susceptibility, although it can also identify single-locus effects [19]. The underlying method

of the MDR filter takes a specified number of polymorphisms and looks at the intersection of genotypes to determine if, for a particular single- or multi-locus genotype, there are more cases than controls with that genotype combination (Figure 2). MDR utilizes a cross-validation measure to divide the data into N equal-sized partitions, looking for high risk genotypes in N-1 partitions –the training set– and then examining the predictive value of those high risk genotypes in the remaining partition of the data – the testing set. The process is then repeated N times until all of the partitions have been used as the testing set. The result is two measures of accuracy for each model MDR evaluates, the classification accuracy and the prediction accuracy. The classification accuracy describes the number of cases and controls the particular model classifies correctly in the training set while the prediction accuracy describes the same measure in the testing set. In this PLATO study, MDR was run with 10-fold cross validation analyzing single-locus models only.

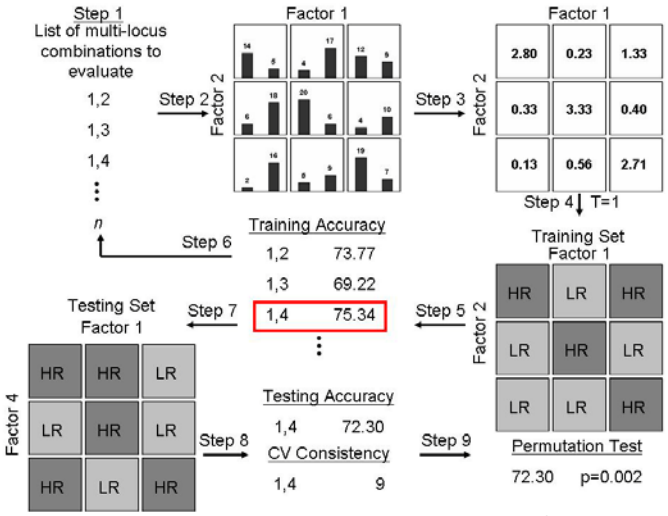


Figure 2. The MDR method. MDR partitions data into N parts and then uses N-1 of those to do the association test and the other part to look at the predictive accuracy of the models found. Adapted from (Ritchie and Motsinger, 2005)[20].

Most of the filters implemented in PLATO utilize multiple different data encodings, as shown in Table 1. There are 5 different data encodings: additive, dominant, recessive, allelic, and genotypic (Figure 3). The additive encoding assumes that the addition of each disease allele results in increased disease risk. Dominant and recessive encodings are very similar, the only difference being where the disease is assumed to reside. In both cases, a 2x2 table is made in which the cases and controls for the dominant homozygote and heterozygote from the 3x2 genotypic table are condensed into one column and the cases and controls for the recessive homozygote reside in the other column. The genotypic encoding is very similar to the additive encoding, with each genotype possessing one column. The only difference is that the genotypic encoding is not necessarily ordered. Where the additive encoding assumes an order to the genotypes in the model, the genotypic encoding does not necessarily possess order in the columns. The allelic encoding simply makes a 2x2 table with the cases and controls that have a major allele and minor allele. Having multiple encodings such as these allows the user to bias a test for a specific disease model which might be present.

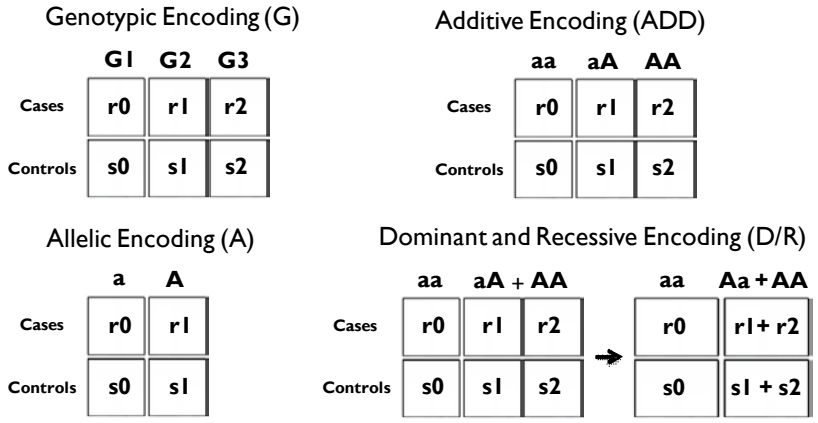


Figure 3. The five different data encodings available in using the PLATO filters are depicted. The r0, r1, r2 values correspond to the number of cases with the particular genotypes while s0, s1 and s2 refer to the controls with those genotypes.

2.3. *The Kappa Statistic*

The kappa statistic has been suggested as a good measure to determine agreement between classifiers [21]. It is a way of examining how well two ranking systems – in this case, filters – classify data in the same way. The idea is to build a 1000x1000 matrix corresponding to the rankings for the 1000 loci by each filter. In the matrix, tallies are placed according to the rankings for each filter (see example in Table 3). If two filters agree on a ranking, the tally will lie directly on the diagonal. The kappa statistic (Eq 1.) looks at the degree to which these tallies group around the diagonal and awards a score of 1 for a perfect agreement at all rankings for two filters. In our method, a weighting measure is used to score tallies closer to the diagonal higher than those that occur further away. Based on suggestions from Landis and Koch [22], a kappa score of 0.60 was used as the cutoff to group filters with similar results. Some of the filters run in this comparison were previously known to have correlated results but were included as a proof of concept for using the kappa statistic.

$$\kappa_w = \frac{x_{++}\sum_{cells} w_{ij}x_{ij} - \sum_{cells} w_{ij}x_{i+}x_{+j}}{x_{++}^2 - \sum_{cells} w_{ij}x_{i+}x_{+j}} \qquad \text{Eq (1)}$$

Table 3. An example of a kappa statistic matrix. First, the rankings for the two filters are aligned to create a matrix which is then populated by tallies for each agreement in rankings in each filter. The kappa statistic then measures the degree to which the tallies fall on the diagonal.

		Filter 1 Rankings					
	Rank	1	2	3	4	5	Sum
Filter 2 Rankings	1	6	.	3	1	.	10
	2	2	8	.	.	.	10
	3	1	1	5	.	3	10
	4	1	.	.	8	1	10
	5	.	1	2	1	6	10
	Sum	10	10	10	10	10	100

2.4. *Implementing the PLATO_MAX approach*

The MAX statistic is a measure utilizing multiple data encodings to maximize the power for finding a genetic effect. Sladek et al.[1] originally utilized this statistic with a combination of the additive, dominant, and recessive encodings in logistic regression. We have extended the statistic to include the genotypic encoding of the chi-square

test, which is shown in our results to be uncorrelated with logistic regression. To implement the PLATO_MAX approach[1] and test its efficacy, a simulation study was performed. First, 100 datasets with 1000 cases and 1000 controls were simulated to find the power of the method. Three genetic effects with an odds ratio of 1.5 – one additive model, one recessive model, and one dominant model – were embedded in these datasets. To find the MAX statistic for each SNP, four filters – LOGISTICREGRESS (ADD/D/R) and CHISQUARE (G) – were run and the minimum p-value between the four was kept as the best solution. These four filters represented one filter from each of the four filter classes identified (as described in the results below). We selected one filter per class based on ease of use and interpretation. In order to deal with multiple testing issues, a set of 1000 permutations was performed, building a null distribution for each SNP. Here, the disease status was randomized to create 1000 null datasets where the genotype matrix was held constant but the association between genotype and phenotype was removed. The full PLATO_MAX analysis was performed on each null dataset and the lowest p-value was obtained from each dataset and collected in the empirical null distribution. The original lowest p-value was then compared to the permutation null distribution to find a corrected p-value. The power was calculated for each of the three effects at α=0.01 and 0.05 levels as the number of times out of the 100 datasets that the SNP in question was found to be significant after permutation testing. The false positive rate was calculated as the average number of incorrect loci found to be significant for each dataset divided by the number of SNPs in the dataset. We also investigated the Type 1 error rate of the PLATO_MAX approach by simulating 1000 datasets with no genetic effect. The PLATO_MAX approach was then run with permutation and the number of times which SNPs were found to be significant with the null model was examined.

3. Results

3.1. *Kappa Comparisons*

The kappa score was used to do pair-wise comparisons between all 24 filters that are used in the current version of PLATO. This created a set of 276 comparisons which were repeated for all 13 models tested. To do the comparisons, the raw results from each filter were first sorted into a list of rankings that were suitable for making an ordered matrix (Table 3). For each filter comparison, the 1000 rankings were lined up so that one filter's rankings made up the columns and the other's made up the rows. Then, tallies were placed in the matrix corresponding to instances in which the rankings from the two filters agreed. The kappa statistic weighs the degree to which these tallies fall on the diagonal, as perfect agreement will be demonstrated by all tallies falling on the diagonal. A kappa statistic score of 1 is given in the case of perfect agreement between two filters. The score of 0.60 was used as significant based on literature about the statistic[22].

Table 4. The list of filter groups that resulted from kappa statistic comparisons.

Group 1	**Group 2**	**Group 3**	**Group 4**
LIKELIHOOD (ADD)	CHISQUARE (A)	ARMITAGE (A)	LIKELIHOODRATIO (D)
LIKELIHOOD (G)	LIKELIHOODRATIO (A)	ARMITAGE (G)	NMI (D)
NMI (ADD)	NMI (A)	LOGISTICREGRESS (ADD)	
NMI (G)			
Group 5	**Group 6**	**Group 7**	**Group 8**
LIKELIHOODRATIO (R)	UNCERTAINTYCOEFF (ADD)	CHISQUARE (G)	LOGISTICREGRESS (D)
NMI (R)	UNCERTAINTYCOEFF (G)		
Group 9	**Group 10**	**Group 11**	**Group 12**
LOGISTICREGRESS (R)	MDR	ODDSRATIO	UNCERTAINTYCOEFF (A)
Group 13	**Group 14**		
UNCERTAINTYCOEFF (D)	UNCERTAINTYCOEFF (R)		

Using the cutoff stated above, the filters were grouped into sets that provided non-redundant results. By grouping the filters for all 13 models, it became apparent that the same groups appeared regardless of the type or

size of the effect simulated, including a null model. The result of this experiment is a set of 14 groups in which all filters within each group had a kappa score of at least 0.60 with each other (Table 4). Once we arrived at these 14 groups, we chose one filter from each group as the representative filter based on the number of assumptions the method made and/or the commonality of its use. We then ran these 14 filters on 10 datasets simulated with 6 genetic effects to determine the remaining correlation between filter rankings present even after kappa statistic comparison. On the basis of these results, we grouped these 14 filters into four filter classes with correlated findings (Table 5). For future analyses, we propose using a single analytical filter from each filter class. This filter is chosen based on interpretability and ease of implementation.

Table 5. Filter Classes found to display correlation in analysis results. Highlighted in bold italics is the selected filter for the PLATO_MAX experiment.

Filter Class 1	Filter Class 2	Filter Class 3	Filter Class 4
CHISQUARE (A)	***CHISQUARE (G)***	LIKELIHOODRATIO (D)	LIKELIHOODRATIO (R)
LOGISTICREGRESS (ADD)	LIKELIHOODRATIO (ADD)	***LOGISTICREGRESS (D)***	***LOGISTICREGRESS (R)***
UNCERTAINTYCOEFF (A)	UNCERTAINTYCOEFF (ADD)	UNCERTAINTYCOEFF (D)	UNCERTAINTYCOEFF (R)
ODDSRATIO	MDR		

3.2. *The PLATO_MAX Approach*

The PLATO_MAX approach was implemented to examine its power to identify multiple types of genetic effects while controlling the false positive and Type 1 error rates of the method. The power of the PLATO_MAX approach (using the four filter class filters) is compared to using each of the four filters individually for each effect in Table 6. In addition, the false positive and Type 1 error rates are given in Table 6.

Table 6. Power, False positive and Type 1 error rate of the PLATO_MAX approach compared to each individual filters.

	MAX	**LOGISTIC (ADD)**	**LOGISTIC (D)**	**LOGISTIC (R)**	**CHISQUARE (G)**
Power (0.05)					
Additive	75	83	78	56	71
Dominant	97	96	97	23	97
Recessive	45	28	7	57	47
AVG POWER	72.3	69.0	60.7	45.3	71.7
False Positive	0.04795	0.05795	0.06039	0.06735	0.0537
Type 1 Error	0.054473				
Power (0.01)					
Additive	52	63	59	34	49
Dominant	92	85	97	11	91
Recessive	24	13	2	37	23
AVG POWER	56	53.7	52.7	27.3	54.3
False Positive	0.00954	0.01458	0.01482	0.01739	0.01308
Type I error	0.01254				

Here the false positive rate is the average number of incorrect loci found to be significant for each dataset divided by the number of SNPs in the dataset where an actual genetic model was simulated. On the contrary, the Type I error rate is the average number of incorrect loci found to be significant for each dataset divided by the number of SNPs in the dataset where no genetic effect was simulated. The PLATO_MAX approach had power of 75%, 97% and 45% to find the additive, dominant, and recessive effects respectively at an alpha level of 0.05 and

power of 52%, 92%, and 24% at the 0.01 level. The average power of the PLATO_MAX approach over the three effects at the 0.05 level was 72.3% as opposed to the average power of the four individual filters which was 69.0% for LOGISTICREGRESS (ADD), 60.7% for LOGISTICREGRESS (D), 45.3% for LOGISTICREGRESS (R), and 71.7% for CHISQUARE (G). The false positive rate of the PLATO_MAX was lower than the individual filters at 0.04795 and 0.00954 for an alpha of 0.05 and 0.01 respectively. Finally, the Type 1 error rate of the method was well controlled at 0.054473 at an alpha of 0.05 and 0.012541 at an alpha of 0.01.

4. Discussion

GWAS analyses have thus far been fairly straightforward single locus tests of association such as logistic regression, chi-square tests, or Cochran-Armitage trend tests. These tests have been successful in many situations. Clearly, the optimal test is highly dependent on the type of effect being detected. Since we do not know *a priori* what type of effect we are looking for, some groups, such as Sladek et al.[1] have proposed using multiple analyses simultaneously and taking the maximum statistic as the final solution. Using multiple analysis approaches (as filters) and employing a maximum statistic allows one to test for many known types of effects and have power to detect them while controlling the Type I error rate.

The motivation for PLATO is twofold. First, the fact that any *single* underlying analytical scheme will reveal only *some* important results and that multiple filters will reveal different subsets of important results. However, once results are obtained these results can be viewed in light of the results from other filters to best understand the full meaning of the genetic data. The potential to use multiple filters forces no *a priori* assumptions about the mode of action of the genetic components of a phenotype allowing the most general possible analysis and interpretation. This is critical as it is rare that we know what type of effect we are attempting to detect in disease gene association studies. Thereby the ability to evaluate the association in the context of many different models and select the optimum solution for the dataset at hand, while controlling Type I error rate is a great success. Second, it is hypothesized that the genetic architecture of complex disease will include interactions between many genes as well as the environment. In GWAS scale datasets, searching for interactions is a computational challenge; thus filtering the full set of GWAS SNPs to a smaller subset will be critical in the quest for detecting interactions. PLATO accomplishes both of these goals.

There are a large number of possible filters that one can envision for the PLATO framework. Currently, PLATO has the following tests implemented: Cochran-Armitage trend test, chi-square, likelihood ratio, logistic regression, MDR, normalized mutual information, odds ratio, and uncertainty coefficient as well as a thorough quality control filter including sample and SNP efficiency, HWE, allele frequency, rates of homozygosity, concordance checks, gender errors, and Mendelian errors. In addition, PLATO currently has the following filters under development: the Biofilter[23], data transformations, conditional logistic regression, MDR-PDT, generalized MDR, Cochran-Mantel-Haenszel analysis, linkage disequilibrium (r^2), linear regression, and TDT.

PLINK is another currently available software package for GWAS data[24]. PLATO differs from PLINK in two significant ways. First, PLINK is primarily for performing one test of association for each single SNP across the genome; whereas PLATO performs multiple single locus tests and uses a MAX statistic with permutation testing to determine statistical significance. Second, while PLINK has a few regression-based tests for interaction, that is not the focus; whereas, the primary goal of PLATO is to provide a mechanism for searching for complex gene-gene and gene-environment interactions in GWAS data. With multiple interaction filters integrated with tests for main effects as well as biological knowledge, PLATO will provide a powerful framework to elucidate the genetic architecture of complex disease.

This study examined the redundancy among filters currently available for PLATO, the creation of filter classes based on these correlations, and the utility of the PLATO_MAX approach in the context of one filter from each class. We simulated case-control data for a number of effects and then compared results from each filter after running them on this data. The kappa statistic provided a means of comparison between these results, allowing for the grouping of filters into sets based on similar results. From 276 filter-filter comparisons, 14 groups of filters with

kappa statistic scores greater than 0.60 were obtained. As expected based on the numerical formulas, these 14 groups were then filtered down through elimination of some filters and grouping of correlated entities to form 4 filter classes. The primary motivation for this research was finding an effective way to filter GWAS data to determine an interesting subset of SNPs and therefore reduce the number of model comparisons during interaction analysis. This was accomplished through selection of an informative group of filters to achieve reduced computational time during a PLATO run. In addition to reducing the number of filters, we have implemented a useful analysis tool in the form of the MAX statistic. Although the PLATO_MAX approach only offers a small power gain for individual genetic effects, it has shown that it has higher power to detect all types of genetic effects than the individual filters composing it. The PLATO_MAX approach also has a lower false positive rate than any of the other filters alone.

The current study offers multiple avenues for future exploration. Now that a set of filter groups has been proposed, it must be determined which filter from each group provides the most accurate results. While it is possible that the best filter from each group could vary for different effects, this will supply a default PLATO filter set to achieve the largest degree of filtering with the smallest computational obligation in most cases. In addition, we can use this new default filter set to test the idea that looking for an intersection between results from multiple filters can filter out background noise. By running several filters that each provide different results and then selecting for SNPs which receive high scores in all filters, it should be possible to sift out the uninformative background noise of SNPs that are significant in one filter only. Power and type I error studies must be performed to test this notion.

In addition to realizing an increase in power for single-locus genetic analysis, this exercise in implementing the MAX statistic has demonstrated an important point which was introduced by the computational optimization field: "No Free Lunch" (NFL)[25]. The NFL theory states that no one method run alone is best in all situations. Although we have demonstrated this theory in the search for single-locus genetic models, it is likely to be an even more important consideration when analyzing epistatic models. This concept is supported also by previous research in the field. Upon testing a number of interaction-searching analysis methods, it was found that the performance of each was dependent on the context of the interaction or genetic effect being searched for[26]. When a single-locus effect was presented, the methods that condition on main effects out-performed those which look specifically for epistasis. On the other hand, when two-locus and three-locus models were imparted to these methods, different interaction-searching methods surpassed both those conditioning upon main effects as well as each other depending on the particular context of the multi-locus effect. In addition, when MDR and FITF were applied to look for gene-environment interactions involved in the etiology of pancreatic cancer, the researchers found that a combination of methods was necessary to mine the data effectively and identify the important multi-locus models[27]. In the future we will extend the PLATO_MAX approach to include methods designed for interaction searching.

PLATO is a very flexible analytical method with promise as a major component of association studies. With its ability to run filters individually, in series or in parallel as well as the opportunity for users to implement their own filters, PLATO can be easily customized for any study. Future work will likely introduce a study design that takes advantage of this customization to use PLATO as both an analytical method and a prior to performing interaction analysis.

5. Acknowledgements

This work was funded by the National Institutes of Health (NIH) Pharmacogenetics Research Network (PGRN) Pharmacogenomics of Arrhythmia Therapy U01 (HL65962), R01 NS032830, U01 HG004608, and LM10040 as well as the Training Program on Genetic Variation and Human Phenotypes grant (5T32GM080178). The authors thank William S. Bush for insightful commentary given during the preparation of this manuscript. The Vanderbilt University Center for Human Genetics Research, Computational Genomics Core provided computational support for this work

6. References

(1) Sladek R, Rocheleau G, Rung J et al. A genome-wide association study identifies novel risk loci for type 2 diabetes. *Nature.* 2007;445:881-885.

(2) Hirschhorn JN, Altshuler D. Once and again-issues surrounding replication in genetic association studies. *J Clin Endocrinol Metab.* 2002;87:4438-4441.

(3) Moore JH. The ubiquitous nature of epistasis in determining susceptibility to common human diseases. *Hum Hered.* 2003;56:73-82.

(4) Culverhouse R, Klein T, Shannon W. Detecting epistatic interactions contributing to quantitative traits. *Genet Epidemiol.* 2004;27:141-152.

(5) Williams SM, Addy JH, Phillips JA, III et al. Combinations of variations in multiple genes are associated with hypertension. *Hypertension.* 2000;36:2-6.

(6) Carrasquillo MM, McCallion AS, Puffenberger EG, Kashuk CS, Nouri N, Chakravarti A. Genome-wide association study and mouse model identify interaction between RET and EDNRB pathways in Hirschsprung disease. *Nat Genet.* 2002;32:237-244.

(7) Dipple KM, McCabe ER. Phenotypes of patients with "simple" Mendelian disorders are complex traits: thresholds, modifiers, and systems dynamics. *Am J Hum Genet.* 2000;66:1729-1735.

(8) Dipple KM, McCabe ER. Modifier genes convert "simple" Mendelian disorders to complex traits. *Mol Genet Metab.* 2000;71:43-50.

(9) Edwards TL, Bush WS, Turner SD et al. Generating Linkage Disequilibrium Patterns in Data Simulations Using genomeSIMLA. *Lecture Notes in Computer Science.* 2008;4793:24-35.

(10) Dudek S, Motsinger AA, Velez D, Williams SM, Ritchie MD. Data simulation software for whole-genome association and other studies in human genetics. *Pac Symp Biocomput.* 2006;11:499-510.

(11) Bush WS, Edwards TL, Dudek SM, McKinney BA, Ritchie MD. Alternative contingency table measures improve the power and detection of multifactor dimensionality reduction. *BMC Bioinformatics.* 2008;9:238.

(12) Agresti A. *Categorical Data Analysis*. New York: John Wiley & Sons; 1990.

(13) Armitage P. Tests for linear trends in proportions and frequencies. *Biometrics.* 1955;11:375-386.

(14) Cochran WG. Some methods for strengthening the common chi-squared tests. *Biometrics.* 1954;10:417-451.

(15) Freidlin B, Zheng G, Li Z, Gastwirth JL. Trend tests for case-control studies of genetic markers: power, sample size and robustness. *Hum Hered.* 2002;53:146-152.

(16) Shannon CE. A Mathematical Theory of Communication. *The Bell System Technical Journal.* 1948;27:379-423.

(17) Forbes AD. Classification-algorithm evaluation: five performance measures based on confusion matrices. *J Clin Monit.* 1995;11:189-206.

(18) Jewell, N. P. Statistics for Epidemiology. 2004. Boca Raton, FL, CRC Press LLC.

(19) Ritchie MD, Hahn LW, Roodi N et al. Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *Am J Hum Genet.* 2001;69:138-147.

(20) Ritchie M, Motsinger AA. Multifactor dimensionality reduction for detecting gene-gene and gene-environment interactions in pharmacogenomics studies. *Pharmacogenomics.* 2005;6:823-834.

(21) Wickens, T. D. Multiway Contingency Tables Analysis for the Social Sciences. 1989. Hillsdale, NJ, Lawrence Erlbaum Associates, Inc.

(22) Landis JR, Koch GG. The measurement of observer agreement for categorical data. *Biometrics.* 1977;33:159-174.

(23) Bush WS, Dudek SM, Ritchie MD. Biofilter: a knowledge-integration system for the multi-locus analysis of genome-wide association studies. *Pac Symp Biocomput.* 2009;368-379.

(24) Purcell S, Neale B, Todd-Brown K et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. *Am J Hum Genet.* 2007;81:559-575.

(25) Wolpert DH, Macready WG. No Free Lunch Theorems for Optimization. *Transactions on Evolutionary Computation.* 1997;1:67-82.

(26) Motsinger-Reif AA, Reif DM, Fanelli TJ, Ritchie MD. A comparison of analytical methods for genetic association studies. *Genetic Epidemiology.* 2008;32:767-778.

(27) Duell EJ, Bracci PM, Moore JH, Burk RD, Kelsey KT, Holly EA. Detecting pathway-based gene-gene and gene-environment interactions in pancreatic cancer. *Cancer Epidemiology, Biomarkers, and Prevention.* 2008;17:1470-1479.

ENABLING PERSONAL GENOMICS WITH AN EXPLICIT TEST OF EPISTASIS

CASEY S. GREENE, DANIEL S. HIMMELSTEIN
Department of Genetics, Dartmouth Medical School, Lebanon, NH 03756, USA

HEATHER H. NELSON
Division of Epidemiology and Community Health, University of Minnesota School of Public Health, Minneapolis, MN, USA

KARL T. KELSEY
Department of Community Health, Brown University, Providence, RI, USA

SCOTT M. WILLIAMS
Department of Molecular Physiology and Biophysics, Vanderbilt University, Nashville, TN, USA

ANGELINE S. ANDREW, MARGARET R. KARAGAS
Department of Community and Family Medicine, Dartmouth Medical School, Lebanon, NH 03756, USA

JASON H. MOORE
Departments of Genetic and Community and Family Medicine, Dartmouth Medical School, Lebanon, NH 03756, USA

One goal of personal genomics is to use information about genomic variation to predict who is at risk for various common diseases. Technological advances in genotyping have spawned several personal genetic testing services that market genotyping services directly to the consumer. An important goal of consumer genetic testing is to provide health information along with the genotyping results. This has the potential to integrate detailed personal genetic and genomic information into healthcare decision making. Despite the potential importance of these advances, there are some important limitations. One concern is that much of the literature that is used to formulate personal genetics reports is based on genetic association studies that consider each genetic variant independently of the others. It is our working hypothesis that the true value of personal genomics will only be realized when the complexity of the genotype-to-phenotype mapping relationship is embraced, rather than ignored. We focus here on complexity in genetic architecture due to epistasis or nonlinear gene-gene interaction. We have previously developed a multifactor dimensionality reduction (MDR) algorithm and software package for detecting nonlinear interactions in genetic association studies. In most prior MDR analyses, the permutation testing strategy used to assess statistical significance was unable to differentiate MDR models that captured only interaction effects from those that also detected independent main effects. Statistical interpretation of MDR models required post-hoc analysis using entropy-based measures of interaction information. We introduce here a novel permutation test that allows the effects of nonlinear interactions between multiple genetic variants to be specifically tested in a manner that is not confounded by linear additive effects. We show using simulated nonlinear interactions that the power using the explicit test of epistasis is no different than a standard permutation test. We also show that the test has the appropriate size or type I error rate of approximately 0.05. We then apply MDR with the new explicit test of epistasis to a large genetic study of bladder cancer and show that a previously reported nonlinear interaction between is indeed significant, even after considering the strong additive effect of smoking in the model. Finally, we evaluated the power of the explicit test of epistasis to detect the nonlinear interaction between two XPD gene polymorphisms by simulating data from the MDR model of bladder cancer susceptibility. The results of this study provide for the first time a simple method for explicitly testing epistasis or gene-gene interaction effects in genetic association studies. Although we demonstrated the method with MDR, an important advantage is that it can be combined with any modeling approach. The explicit test of epistasis brings us a step closer to the type of routine gene-gene interaction analysis that is needed if we are to enable personal genomics.

1. Introduction

1.1. *Personal Genomics*

The era of commercial genetic testing and personal genomics was ushered in with help from the discovery and characterization of mutations in *BRCA1* and *BRCA2* that account for between 20% and 40% of all cases of familial breast cancer [1]. Unfortunately, the remaining 60% to 80% of familial breast cancer remains unexplained and the elusive *BRCA3* gene has not yet been identified despite significant efforts using the full spectrum of genetic and genomic tools available [2]. Failure to find the putative *BRCA3* gene is somewhat surprising given the familial nature and high heritability of this type of breast cancer. The current strategy for revealing genetic architecture is to carry out a genome-wide association study (GWAS) with a million or more single nucleotide polymorphisms (SNPs) that capture much of the common single nucleotide variation in the human genome by tagging blocks of variants that are in linkage disequilibrium [3,4]. These SNPs are then individually tested for association with a specific disease state. The GWAS approach is based on the hypothesis that scanning the entire genome for single SNP associations in an unbiased manner that ignores current

understanding about disease etiology will reveal much of the currently unexplained genetic architecture of a particular disease.

Despite the excitement surrounding the GWAS approach, and the time and financial resources already committed, the results have generally been underwhelming. Consider, for example, the application of GWAS to identification of cancer susceptibility genes. A recent review of these studies shows that a number of new susceptibility loci have been identified for several types of cancer, including breast, prostate, colorectal, lung and skin [5]. The identification of new associations is certainly important. However, as Easton and Eeles [5] note, the increase in risk for the susceptibility alleles at each of these loci is generally 1.3-fold or less. For familial breast cancer, Easton et al. [6] reported five significant, replicated associations that were identified by GWAS in a three-stage study design. Four of these variants were in known genes and one was located in a hypothetical gene. Assuming a multiplicative model, these five loci combine to explain only 3.6% of the excess familial risk of breast cancer and, as suggested by Ripperger et al. [2] were not deemed to be suitable for genetic testing due to their small effect sizes [6]. In a recent follow up study with two additional stages of testing and replication two additional susceptibility loci were identified with odds ratios of 1.11 and 0.95, respectively, each accounting for much less than 1% of the familial risk of breast cancer [7]. When combined with the previously known genetic risk factors for familial breast cancer, the estimated fraction of risk explained is approximately 5.9%. This is in stark contrast to *BRCA1* and *BRCA2* mutations that account for between 20% and 40% of familial breast cancer. While the application of GWAS to familial breast cancer has generated new knowledge, it has not resulted in new genetic tests that can be used to predict and prevent familial breast cancer. These results are particularly discouraging for more common diseases such as sporadic breast cancer that are likely to have a much more complex genetic architecture. As Clark et al. [8] predicted, our success with GWAS depends critically on the assumptions we make about disease complexity. It is the goal of this study to develop a new hypothesis testing methodology that can be used to directly confront the challenge of detecting and characterizing epistasis or nonlinear gene-gene interaction that accounts for a portion of the complex etiology of common diseases.

1.2. *Genetic Architecture of Common Diseases*

When designing and executing a genetic association study of disease susceptibility it is very important to consider the assumptions that are being made about the genetic architecture of the disease [8]. The questions that we ask, the hypotheses that we formulate, the analytical tools selected for data analysis and the inferences we make from the results are all limited by the assumptions we make about genetic architecture. Weiss [9] has defined genetic architecture as 1) the set of genes and DNA sequence involved in the disease, 2) their variation in the population and 3) their specific effects on the phenotype. It was initially thought that much of the genetic risk of familial breast cancer could be explained by three genes (*BRCA1*, *BRAC2* and the hypothetical *BRCA3*). However, it is now clear that the remaining 60% to 80% of risk is likely to be explained by many genes each with multiple variations that have very small effects. It also likely that each variant contributes to risk of sporadic breast cancer through nonlinear interactions with other variants in the genome and with multiple environmental factors such as diet and smoking. We focus here on epistasis or gene-gene interaction that is expected to be a ubiquitous component of the genetic architecture of common diseases.

William Bateson coined the word epistasis in the early 1900s to explain deviations from Mendelian inheritance [10]. The term literally means “standing upon”, and Bateson used it to describe characters that were layered on top of other characters thereby masking their expression. Since Bateson there have been many different and evolving definitions of epistasis or gene-gene interaction [e.g. 11-17]. For example, Fisher [18] defined epistasis in a statistical manner as an explanation for deviation from additivity in a linear model. This non-additivity of genetic effects measured mathematically is different than the more biological definition of epistasis from Bateson. We have previously made the distinction between Bateson's biological epistasis and Fishers statistical epistasis [16]. This distinction is important to keep in mind when thinking about the genetic architecture of common human diseases because biological epistasis happens at the cellular level in an individual while statistical epistasis is a pattern of genotype to phenotype relationships that results from genetic variation in a human population. This distinction becomes important when attempting to draw a biological conclusion from a statistical model that describes a genetic association. Moore and Williams [16] and Phillips [13] have discussed the idea that more modern definitions of epistasis may be needed in light of our new knowledge about gene networks and biological systems. However, the classic definitions provided by Bateson [10] and Fisher [18] still provide a good starting point for thinking about gene-gene interactions.

1.3. *A Multifactor Dimensionality Reduction Approach to Detecting Epistasis*

As discussed above, one of the early definitions of epistasis was deviation from additivity in a linear model [18]. The linear model plays a very important role in modern genetic epidemiology because it has a solid theoretical foundation, is easy to implement using a wide-range of different software packages, and it is easy to interpret.

Despite these good reasons to use linear models [14,15], they do have limitations for explaining genetic models of disease because they have limited ability to detect nonlinear patterns of interaction [19]. Here, a nonlinear interaction is defined as a synergistic or nonadditive effect of multiple genetic variants that is greater than the independent effects of the variants considered alone. It is well documented that linear models have greater power to detect main effects than interactions [20-22]. The limitations of the linear model and other parametric statistical approaches have motivated the development of computational approaches such as those from machine learning and data mining that make fewer assumptions about the functional form of the model and the effects being modeled [23-25]. Several recent reviews highlight the need for new methods [26] and discuss and compare different strategies for detecting statistical epistasis [15,27].

As reviewed recently by Cordell [15], multifactor dimensionality reduction or MDR has emerged as one important new and novel method for detecting and characterizing patterns of statistical epistasis in genetic association studies that complements the linear modeling paradigm. Multifactor dimensionality reduction (MDR) was developed as a nonparametric (i.e. no parameters are estimated) and genetic model-free (i.e. no genetic model is assumed) data mining and machine learning strategy for identifying combinations of discrete genetics and environmental factors that are predictive of a discrete clinical endpoint [28-34]. Unlike most other methods, MDR was designed to detect interactions in the absence of detectable main effects and thus complements other statistical approaches such as logistic regression and other machine learning methods such as random forests and neural networks. At the heart of the MDR approach is a feature or attribute construction algorithm that creates a new variable or attribute by pooling genotypes from multiple SNPs (see Figure 1). The general process of defining a new attribute as a function of two or more other attributes is referred to as constructive induction, or attribute construction, and was first described by Michalski [35]. Constructive, induction using the MDR kernel, is accomplished in the following way. Given a threshold *T*, a multilocus genotype combination is considered high-risk if the ratio of cases (subjects with disease) to controls (healthy subjects) exceeds *T*, otherwise it is considered low-risk. Genotype combinations considered to be high-risk are labeled G_1 while those considered low-risk are labeled G_0. This process constructs a new one-dimensional attribute with values of G_0 and G_1. It is this new single variable that is assessed, using any classification method. The MDR method is based on the idea that changing the representation space of the data will make it easier for methods such as logistic regression, classification trees, or a naive Bayes classifier to detect attribute dependencies. As such, MDR complements any classification methods such as those reviewed by Hastie et al. [24]. Cross-validation is used to prevent overfitting while permutation testing is used to assess statistical significance and to control for false-positives due to multiple testing. This method has been confirmed in numerous simulation studies and a user-friendly open-source MDR software package written in Java is freely available from www.epistasis.org.

Although MDR is a powerful method for detecting nonlinear interactions in the absence of independent main effects it, like other machine learning methods, does not explicitly disentangle these two types of genetic effects. In other words, a statistically significant MDR model could capture interactions, main effects or both interactions and main effects. It may not be immediately apparent to the user which types of effects are represented in a high-order MDR model. This has been previously addressed through post-hoc analysis methods that use entropy-based measures of interaction information to identify evidence of nonlinear interactions [33]. These information theoretic approaches work well but do not reveal directly which genetic effects made a meaningful contribution to the statistical significance. We propose here a new explicit test of epistasis that can be used in conjunction with MDR or any other method to directly test for nonlinear gene-gene interaction while holding the independent main effects constant.

1.4. *Redefining the Null Hypothesis in Genetic Association Studies*

The present study is motivated by the need to greatly improve our knowledge of biological and statistical epistasis and its role in human health and disease. We know very little about the role of epistasis in human biology and public health because the focus for so long as has been on the effects of single genes and single genetic variants in biological and clinical endpoints. Given the ubiquity of complexity in genetic architecture, with epistasis as a central component, we propose a rephrasing of our research questions. Instead of asking which single SNP is associated with disease, we propose asking which combination of SNPs is associated with disease. Rephrasing the question in this manner necessitates a redefinition of the null hypotheses that needs to be tested using statistical and computational methods. Given the reality of complexity, and this specific research question, we propose the following logical set of hypotheses as a starting point for retooling our analytical approach to this problem. First, we propose testing the null hypothesis that the associations in the data are only linear and additive using methods such as MDR and the explicit test of epistasis that were designed specifically for this purpose. Once there is significant evidence for rejecting the null hypothesis of linearity, it is then a logical next step to test the universal null hypothesis of no association using linear statistical methods such as logistic regression that are powered to model the independent and additive main effects. Rejection of the universal null in addition to the linear null provides a set of results generated in a systematic manner that

addresses complexity that can then be interpreted biologically using experimental methods or that can be interpreted statistically using approaches such as parsimony. Is the evidence generated by testing the linear null more compelling than the evidence generated by testing the universal null? Answering this question will help further our understanding of genetic architecture. We propose here a new 'explicit test of epistasis' that allows us to directly test the linear null hypothesis using MDR or any other method.

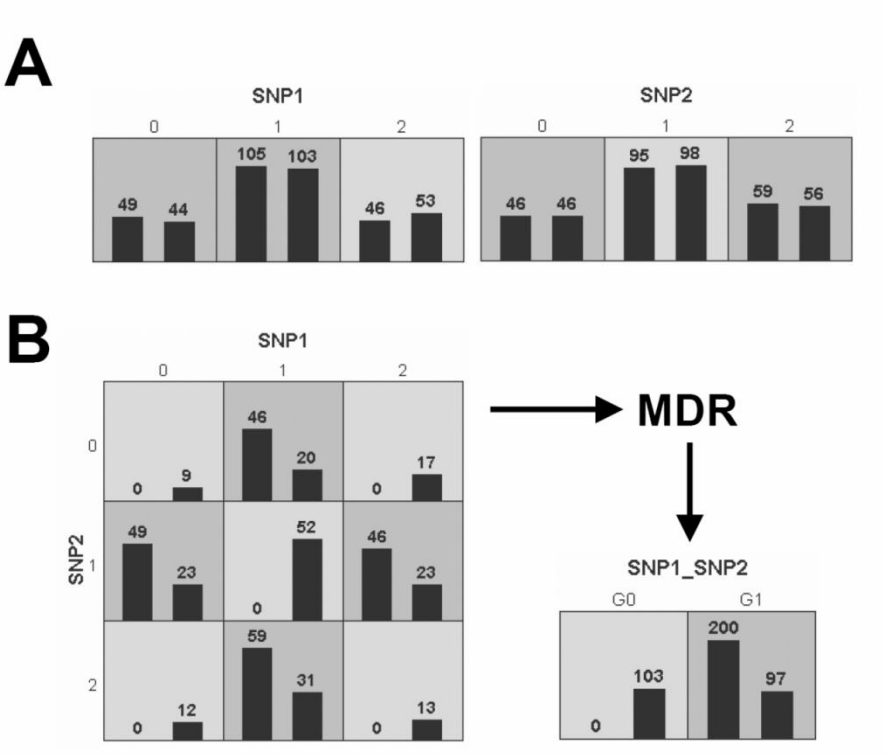


Figure 1. MDR attribute construction. A) Illustrates distribution of cases (left bars) and controls (right bars) for each of the three genotypes of SNP1 and SNP2. The dark-shaded cells have been labeled 'high-risk' using a threshold of $T = 1$. The light-shaded cells have been labeled 'low-risk'. B) Illustrates the distribution of cases and controls when the two functional SNPs are considered jointly. A new single attribute is constructed by pooling the "high-risk" genotype combinations into one group (G1) and the low-risk" into another group (G0).

2. Methods

2.1. *An Explicit Test of Epistasis*

The goal of our proposed explicit test of epistasis is provide a hypothesis testing framework that will allow us to directly test the null hypothesis that the only genetic effects in the data are linear and additive. As described in detail by Pattin et al. [36], the current hypothesis testing framework for MDR is based on a permutation test that randomizes the class (i.e. case and control) labels so that the only genetic associations in the permuted data are there by chance (see Figure 1A and 1B). Permutation testing is used because it doesn't assume we know the null distribution of the test statistic (e.g. testing accuracy) and it controls for false-positives due to multiple testing. However, the current permutation testing framework provides a global p-value for an MDR model that might have main effects, gene-gene interactions, or a combination of both. Significance tells us nothing about the nature of the MDR model and only reflects the fact that the model predicts class better than chance.

We propose here an explicit test of interaction that has all the same advantages of the permutation testing framework but that is able to provide a p-value that reflects only the nonlinear interaction or epistasis component of the model. To accomplish this, we first sort the data rows (i.e. the subjects) by class into cases and controls (see Figure 1C). We then randomize each column (i.e. the SNPs) within each class. This removes any relationship between genotypes within class but preserves the overall genotype frequency difference between the classes. This new type of permutation randomizes any interaction effects while keeping the independent main effects as defined by class differences in genotype frequency. This allows us to generate permuted datasets under the null hypothesis that the only genetic associations in the data are linear or additive in nature and that any nonlinear interaction effects are only there by chance. This yields an explicit test of epistasis when combined with a method such as MDR that is capable of modeling nonlinear interactions.

We have included the explicit test of interaction in the MDR permutation testing (MDRpt) module that is open-source and freely available from www.epistasis.org.

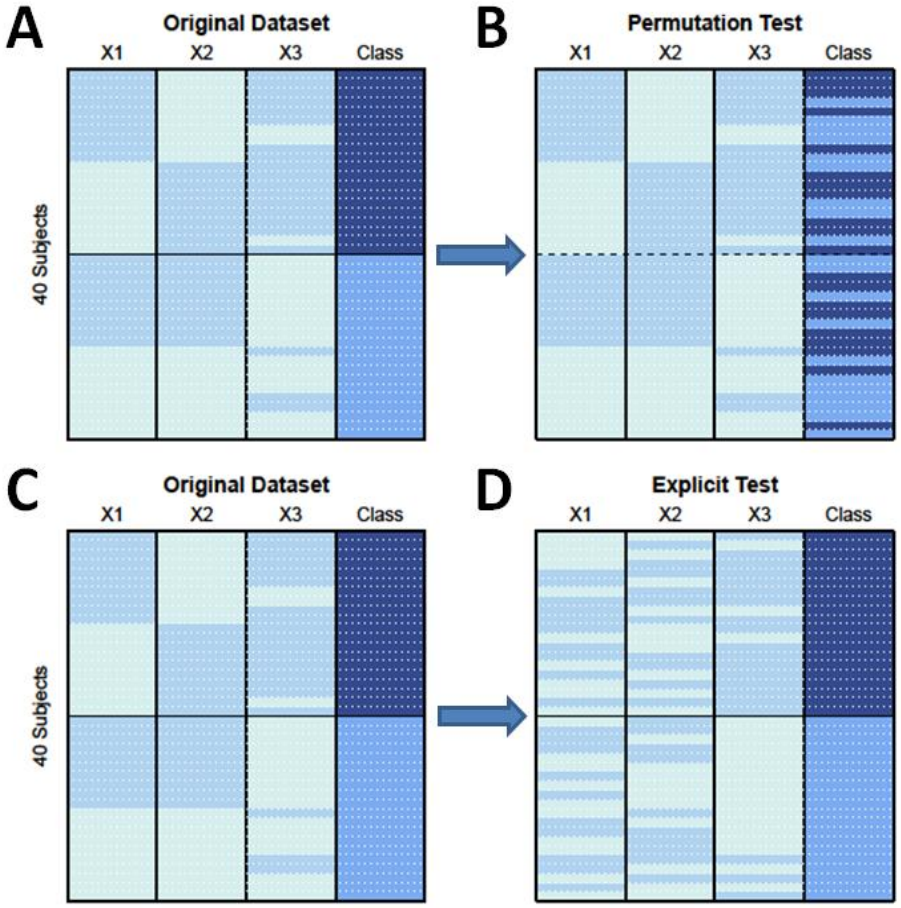


Figure 2. Overview of the explicit test of epistasis. Shown on the left (panels A and C) is a hypothetical dataset with three attributes (e.g. SNPs) coded X1, X2 and X3 and class (i.e. case-control status). Each row of the dataset is one of 40 subjects with hypothetical binary genotypes colored in light shades of blue and case-control status coded darker shades of blue. In this simple example, X1 and X2 effect disease risk through a nonlinear interaction while X3 has an independent main effects that is reflected by a frequency difference in genotypes between cases and controls. Panel B shows the process of randomizing class labels in a standard permutation test. Panel D shows the same data randomized for the explicit test of interaction. Here, the columns are randomized within each class. Note that the genotype frequencies within each class remain fixed. This preserves the independent main effects while randomizing any nonlinear interactions.

2.2. *Multifactor Dimensionality Reduction Analysis*

As described above, the goal of MDR is to change the representation space of the data using constructive induction to make nonlinear interactions easier to detect. This is accomplished by combining two or more variables or attributes into a single attribute that can be modeled using a discrete data classifier. Here, we used a simple probabilistic classifier that is similar to naïve Bayes [31] to model the relationship between variables constructed using MDR and case-control status. Naïve Bayes classifiers were assessed using balanced accuracy as recommended by [37]. For each dataset we evaluated all possible pairwise combinations of SNPs using MDR. The model with the maximum training accuracy as assessed with ten-fold cross validation was selected as the best model. The testing accuracy (i.e. predictive ability) of the single best MDR model was then assessed using the cross-validation hold-out data. We used the open-source MDR software package that is freely available from www.epistasis.org. A tutorial on MDR can be found in the November and December 2006 postings at compgen.blogspot.com.

2.3. *Evaluation of Power and Type I Error Using Simulated Data*

The goal of the simulation study was to generate artificial datasets that could be used to evaluate the power of the MDR within the explicit test of epistasis framework to detect nonlinear gene-gene interactions. We developed a total of 35 different penetrance functions that define a probabilistic relationship between genotype and phenotype where susceptibility to disease is dependent on genotypes from two loci in the absence of any marginal effects. The models were distributed evenly across seven broad-sense heritabilities (0.01, 0.025, 0.05,

0.1, 0.2, 0.3, and 0.4) with minor allele frequencies of 0.4. A total of five models for each of the seven heritabilities were generated for a total of 35 models. More information about the mathematics of penetrance functions and heritability can be found in Culverhouse et al. [38]. A heritability of 0.01 is a very small genetic effect size while 0.4 is a very large genetic effect size. The details of the 35 penetrance functions used here have been previously described in detail by Velez et al. [37]. Genotype frequencies for all 35 epistasis models were consistent with Hardy-Weinberg proportions. One hundred data sets were generated for each model with three sample sizes (400, 800, and 1600 total individuals) with case-control proportions of 1:1. Each pair of functional polymorphisms was embedded within a set of 20 independent single-nucleotide polymorphisms (SNPs). A total of 7,000 artificial datasets were generated and analyzed. For evaluating the type I error of the explicit test of epistasis, null data sets with no functional SNPs were generated by permuting the case-control labels of the data sets described above. All simulated data are available upon request.

The power of MDR using the explicit test of epistasis test was estimated as the percentage of times MDR correctly identified the two functional SNPs in the best model out of each set of 100 datasets for which the result was statistically significant at the 0.05 level (i.e. the testing accuracy was equal to or higher than the top 5% highest testing accuracies in the permuted data). Type I error was estimated as the proportion of times that the permutation test indicated a statistically significant MDR model in data consistent with the null hypothesis of no association.

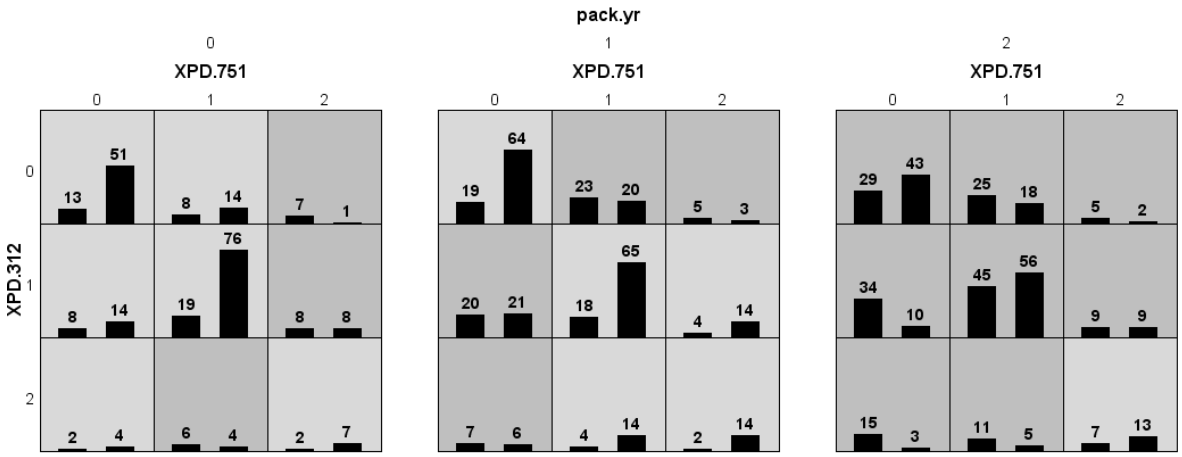


Figure 3. Distribution of cases (left bars) and controls (right bars) for each XPD genotype (coded 0, 1, 2) and for pack years of smoking (pack.yr) in the bladder cancer example. Dark shaded cells indicate high-risk for disease while light-shaded cells indicate low-risk. The p-value from a standard permutation test for this model was <0.001. Note that it is difficult to tell which attribute has a main effect and which are interacting and how these different effects contribute to the statistical significance.

2.4. *Application to Bladder Cancer*

We demonstrated use of the explicit test of epistasis with real data by applying it to a genetic epidemiology study that examined the relationship between DNA repair gene SNPs, smoking, and bladder cancer susceptibility that was previously analyzed using MDR and a 1000-fold permutation test [39]. The study analyzed 355 bladder cancer cases and 559 controls ascertained from the state of New Hampshire. This study focused specifically on genes that play an important role in the repair of DNA sequences that have been damaged by chemical compounds (e.g. carcinogens). Seven SNPs were measured including two from the *X-ray repair cross-complementing group 1* gene (*XRCC1*), one from the *XRCC3* gene, two from the *xeroderma pigmentosum group D (XPD)* gene, one from the *nucleotide excision repair* gene (*XPC*), and one from the *AP endonuclease 1* gene (*APE1*). Each of these genes plays an important role in DNA repair. Smoking is a known risk factor for bladder cancer and was included in the analysis along with gender and age for a total of 10 attributes. Age was discretized to > or ≤ 50 years.

A parametric linear statistical analysis of each attribute individually revealed a significant independent main effect of smoking as expected ($P < 0.05$). However, none of the measured SNPs were significant predictors of bladder cancer individually ($P > 0.05$). Andrew et al. [39] used MDR to exhaustively evaluate all possible two-, three-, and four-way interactions among the genetic and environmental attributes. For each combination of attributes a single constructed attribute was evaluated using a naïve Bayes classifier. Training and testing accuracy were estimated using 10-fold cross-validation. A best model was selected that maximized the testing accuracy. The best model had a testing accuracy of approximately 0.63 and included two SNPs from the *XPD* gene and smoking. The distribution of cases and controls with each genotype/smoking combination is illustrated above in Figure 3. They statistically evaluated this model with a 1000-fold permutation test and determined these results to be highly significant ($p<0.001$). Post-hoc analysis of the MDR model using entropy-based

measures of interaction information revealed that the two *XPD* polymorphisms had evidence of nonlinear interaction or synergy in the near complete absence of main effects. Interestingly, the joint effect of the two *XPD* SNPs was larger than the independent from the effect of smoking. As such, these data provide an ideal test case for the proposed explicit test of interaction. Is the nonlinear interaction between the two *XPD* SNPs statistically significant after holding the effects of smoking constant in the new permutation test or was the significance only due to the large effect of smoking? To answer this question we applied MDR with the explicit test of interaction to the bladder cancer data and determined the statistical significance of the model comprised of the two SNPs from the *XPD* gene and smoking.

To assess the power of the explicit test of epistasis to detect the joint effect of the two *XPD* SNPs in the bladder cancer we simulated 100 datasets using three different MDR models from the bladder cancer data analysis described above. First, we simulated 100 datasets using the MDR model containing the two *XPD* SNPs. Second, we simulated 100 datasets using the MDR model containing just smoking. Third, we simulated 100 datasets using the MDR model containing the two *XPD* SNPs with smoking. The total number of simulated attributes was the same as the original data. We applied MDR along with the explicit test of interaction to each simulated dataset and recorded the power to detect an interaction. We expect the results of this study to provide realistic power estimates for real data with a detectable interaction and a strong independent main effect.

3. Results

3.1. *The Power and Type I Error of the Explicit Test of Epistasis*

Table 1 summarizes the power and the type I error (in parentheses) of the explicit test of epistasis to detect nonlinear interactions in the simulated data using MDR models. The power exceeded 0.80 for all sample sizes for data with moderate to large genetic effect sizes (heritability > 0.025). Power also exceeded 0.80 at sample sizes of 1600 and 800 for the small genetic effect sizes of 0.01 and 0.025, respectively. It is important to note that these power estimates are extremely close (± 0 to 0.01) to those estimated using a standard permutation test by Pattin et al. [36]. These results demonstrate that the new explicit test of interaction does not lose power to detect nonlinear interactions as compared to a standard permutation test.

Also shown in Table 1 in parentheses are the estimates of the false-positive rate or type I error. Note that in each case the type I error rate was approximately 0.05 suggesting that the explicit test of interaction is an appropriately sized test. As with power, this is not different than has been previously reported for standard permutation tests with MDR [36]. This is important given that MDR is a machine learning algorithm that looks at the data in a combinatorial manner.

Table 1. Summary of the power and type I error (parentheses) of the explicit test of interactions when combined with MDR.

	Heritability						
Sample Size	0.01	0.025	0.05	0.10	0.20	0.30	0.40
400	0.22 (0.06)	0.65 (0.06)	0.87 (0.04)	0.95 (0.05)	1.00 (0.05)	1.00 (0.05)	1.00 (0.05)
800	0.51 (0.04)	0.88 (0.07)	0.98 (0.04)	1.00 (0.06)	1.00 (0.06)	1.00 (0.04)	1.00 (0.07)
1600	0.87 (0.05)	1.00 (0.05)	1.00 (0.05)	1.00 (0.04)	1.00 (0.05)	1.00 (0.05)	1.00 (0.04)

3.2. *Application to Bladder Cancer*

As described above, the bladder cancer study of Andrew et al. [39] makes an ideal test case for the new explicit test of interaction because a statistically significant MDR model was detected that consisted of two interacting SNPs and smoking that appeared to have an independent main effect. This model was determined to be significant at the 0.001 level using a standard permutation test and, at the time, it wasn't clear the degree to which the significance was due to the main effect of smoking, the nonlinear gene-gene interaction, or both. We applied MDR with the explicit test of interaction and found the same best model with the p-value of 0.005. This is a highly significant result that confirms the important role of a nonlinear interaction between the two *XPD* polymorphisms. This synergistic interaction was still highly significant even after controlling for the contribution made by smoking, a known risk factor for bladder cancer.

Figure 4 below illustrates the distribution of testing accuracies for best MDR models from the standard permutation test and the explicit test of interaction. First, note that the center for the permutation distribution is approximately 0.50. This is the result that is expected if a fair coin were used to predict who is a case and who is a control. Now note that the distribution for the explicit test of epistasis is shifted to the right. This shift is due to the factors with independent main effects in the data such as smoking that are fixed during the randomization process used by the explicit test of epistasis.

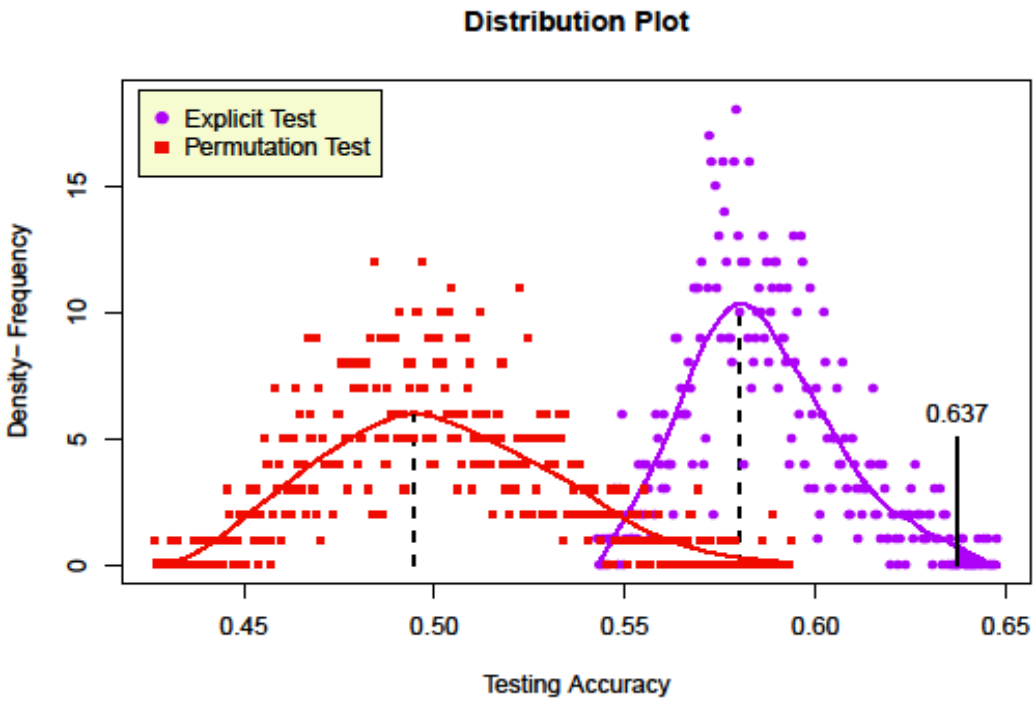


Figure 4. Distribution of testing accuracies from best MDR models obtained from 1000 datasets randomized using a standard permutation test (red squares) and the explicit test of epistasis (purple circles). Note that the permutation distribution is centered (dashed line) at approximately 0.50, as expected. However, the center null distribution derived from the explicit test of epistasis is shifted to the right. This new center is consistent with the fixed main effects in the data. The testing accuracy for the best MDR model from the bladder cancer data is shown on the right (solid line). The area to the right of 0.637 is shaded purple and is equivalent to the p-value of 0.005.

What was the power to detect this effect? As described above, we evaluated power by simulating data from MDR models of the two *XPD* SNPs, just smoking and the two *XPD* SNPs with smoking. We found that the power to detect just the interaction was 1.00 while the power to detect the interaction with the effect of smoking in the model was 0.94. This reduction in power is not surprising given the increase in MDR model size from two factors (two dimensions) to three factors (three dimensions). As expected, the power to detect an interaction for the model with just smoking was 0.06. This is approximately equal to the type I error rate of 0.05 since there was no interaction to find. These findings confirm the results from the earlier simulation study. The power results are not relevant to the actual bladder cancer data analysis since a highly significant model was detected with a p-value of 0.005. However, they do help to reveal the operating characteristics of the explicit test of interaction.

4. Discussion

Epistasis or gene-gene interaction is expected to be a ubiquitous component of the genetic architecture of common human diseases [40]. As such, it has very important implications for the success of personal genomics which is currently based almost entirely on results from genetic association studies that only consider one SNP or one gene at a time. Moore and Williams [41] have suggested that personal genomics will not reach its full potential to impact human health until the full complexity of the genotype-to-phenotype relationship is addressed in all genetic studies. What can be done to improve the usefulness of personal genomics? Moore and Williams [41] offer the following five recommendations:

1. We need to greatly improve our understanding of biological and statistical epistasis and their roles in human health and disease.
2. We need powerful analytical tools that are designed to address the complexity of genetic architecture due to epistasis and other phenomena.
3. We need better experimental methods for confirming statistical models of epistasis in animal models or in human cell culture.
4. We need to remember the principles of classical genetics as we immerse ourselves in the excitement of cutting-edge genotyping technology and emerging methods for rapidly sequencing an entire genome.

5. We need to continue to integrate systems biology into human genetics in a meaningful manner.

The goal of the present study was to develop a hypothesis testing framework and methodology that can be used with methods such as MDR that were designed specifically for detecting and characterizing nonlinear or nonadditive gene-gene interactions in genetic association studies. As such, this study is consistent with the first two recommendations listed above. We have introduced an explicit test of epistasis that can be used to test the null hypothesis that the only genotype-to-phenotype relationships in the data are linear and additive. This is important because until now methods such as MDR could only perform a universal test of the null hypothesis of no association [36]. Inferences about nonadditive interactions were made from post-hoc analyses using methods based on information theory [33]. We demonstrated using simulated data that this approach retains the power of a standard permutation test to detect epistasis across a range of effects sizes and sample sizes. Further, we demonstrated that this new approach has a reasonable type I error rate of approximately 0.05. Finally, we applied this new approach to a large genetic study of bladder cancer and were able to confirm a previously reported nonadditive gene-gene interaction in the presence of the large independent effect of smoking [39].

In addition to introducing a new method for epistasis analysis, we have also introduced a new hypothesis testing framework that redefines the null hypothesis of no genetic association into component parts that are more consistent with the assumption that the genetic architecture of common diseases is complex (see Section 1.4). It is important to note that idea of testing the null hypothesis of linearity using nonlinear statistical methods is not new. For example, Theiler et al. [42] introduced the method of surrogate data in the context of time series analysis as way to test for nonlinear patterns with the confounding of linear patterns. With the method of surrogate data, a discrete Fourier transform of a time series is taken, the phases are randomized and a new time series generated using an inverse discrete Fourier transform. The resulting phase-randomized time series has the same linear patterns as the original time series with all other patterns randomized. This procedure makes it possible to test the null hypothesis of linearity using any statistic that is capable of measure nonlinear patterns. As reviewed by Moore [43], the method of surrogate data is a type of permutation and thus has many similarities to the explicit test of interaction introduced here.

The advantages of the explicit test of epistasis include its simplicity and its flexibility. First, the explicit test of interaction is simply a modified permutation test that randomizes the attribute columns within each class. Thus, it can be easily implemented in a Perl or Python script or in a data analysis package such as R. We have also provided the method in the open-source MDR permutation testing module. Second, the approach is very flexible in that it can be generally applied to any method that is designed for detecting nonlinear gene-gene interactions. Thus, it could be combined with other machine learning methods such as decision trees, neural networks or support vector machines. The only disadvantage of the approach is that permutation testing can add a significant amount of computational time. This will be important for application of these methods to GWAS. Approaches such as the extreme value distribution (EVD) that can reduce the number of permutations that need to be performed are likely to help address this problem [36].

We recommend several future studies with the explicit test of epistasis. First, it will be interesting to use the explicit test of epistasis to compare the power of different methods for detecting gene-gene interactions in the presence of independent main effects. This will be important because some methods may be confounded by any linear additive patterns in the data. Second, it will be important to demonstrate that the EVD approach described by Pattin et al. [36] could be combined with the explicit test of epistasis without violating the distributional assumptions of the EVD. This will be important in the context of GWAS where computational efficiency is extremely important. Finally, it will be very important to implement the explicit test of interactions with other real datasets where both interactions and independent main effects are present. Reanalysis of published epistasis results to confirm nonlinear interactions will be helpful for determining statistical significance. We anticipate the explicit test of epistasis will play an important role in the detection, characterization and interpretation of nonlinear gene-gene interactions in genetic association studies. As such, it will play an important role in improving the impact of personal genomics and other healthcare endeavors that depend critically on published genetic association results that reflect the underlying genetic architecture of the disease in question.

Acknowledgments

This work was supported by NIH grants LM009012, LM010098, HD047447, AI59694 and ES007373. We would like to thank the anonymous reviewers for their very helpful comments.

References

1.Narod SA, Foulkes WD (2004) Nat Rev Cancer 4:665-76.
2.Ripperger T, Gadzicki D, Meindl A, Schlegelberger B (2009) Eur J Hum Genet 17:722-31.

3.Hirschhorn JN, Daly MJ (2005) Nat Rev Genet 6:95-108.
4.Wang WY, Barratt BJ, Clayton DG, Todd JA (2005) Nat Rev Genet 6:109-18.
5.Easton DF, Eeles RA (2008) Hum Mol Genet 17:R109-15.
6.Easton DF, Pooley KA, Dunning AM, Pharoah PD, Thompson D, Ballinger DG, et al. (2007) Nature 447:1087-93.
7.Ahmed S, Thomas G, Ghoussaini M, Healey CS, Humphreys MK, Platte R, et al. (2009) Nat Genet 41:585-90.
8.Clark AG, Boerwinkle E, Hixson J, Sing CF (2004) Genome Res 15:1463-7.
9.Weiss KM (1993) Genetic variation and human disease. Cambridge University Press, New York.
10.Bateson W (1909) Mendel's Principles of Heredity. Cambridge University Press, Cambridge.
11.Hollander WF (1955) J Hered 46:222-225.
12.Phillips PC (1998) Genetics 149:1167-71.
13.Phillips PC (2008) Nat Rev Genet 9:855-67.
14.Cordell HJ (2002) Hum Mol Genet 11:2463-8.
15.Cordell HJ (2009) Nat Rev Genet, in press.
16.Moore JH, Williams SW (2005) BioEssays 27:637-46.
17.Tyler AL, Asselbergs FW, Williams SM, Moore JH (2009) Bioessays 31:220-7.
18.Fisher RA (1918) Trans R Soc Edinb 52:399-433.
19.Moore JH, Williams SW (2002) Ann Med 34:88-95.
20.Lewontin RC (1974) Am J Hum Genet 26:400-411.
21.Lewontin RC (2006) Int J Epidemiol 35:536-7.
22.Wahlsten D (1990) Behav Brain Sci 13:109-161.
23.Mitchell T (1997) Machine Learning. McGraw-Hill, New York.
24.Hastie T, Tibshirani R, Friedman J (2001) The Elements of Statistical Learning : Data Mining, Inference, and Prediction. Springer-Verlag, New York.
25.McKinney BA, Reif DM, Ritchie MD, Moore JH (2006) Appl Bioinformatics 5:77-88.
26.Thornton-Wells TA, Moore JH, Haines JL. (2004) Trends Genet 20:640-7.
27.Motsinger AA, Ritchie MD, Reif DM (2007) Pharmacogenomics 8:1229-41.
28.Ritchie MD, Hahn LW, Roodi N, Bailey LR, Dupont WD, Parl FF, Moore JH (2001) Am J Hum Genet 69:138-147.
29.Ritchie MD, Hahn LW, Moore JH (2003) Genet Epidemiol 24:150-157.
30.Hahn LW, Ritchie MD, Moore JH (2003) Bioinformatics 19:376-82.
31.Hahn LW, Moore JH (2004) In Silico Biol 4:183-94.
32.Moore JH (2004) Expert Rev Mol Diagn 4:795-803.
33.Moore JH, Gilbert JC, Tsai CT, Chiang FT, Holden T, Barney N, White, B. C. (2006) J Theor Biol 241, 252-261.
34.Moore JH (2007) In: Zhu, X., Davidson, I. eds. Knowledge Discovery and Data Mining: Challenges and Realities with Real World Data, IGI Global, pp. 17-30.
35.Michalski RS (1983) Artif Intell 20:111-161.
36.Pattin KA, White BC, Barney N, Gui J, Nelson HH, Kelsey KT, Andrew AS, Karagas MR, Moore JH (2009) Genet Epidemiol. 33(1):87-94.
37.Velez DR, White BC, Motsinger AA, Bush WS, Ritchie MD, Williams SM, Moore JH (2007) Genet Epidemiol 31:306-315.
38.Culverhouse R, Suarez BK, Lin J, Reich T (2002) Am J Hum Genet 70:461-71.
39.Andrew AS, Nelson HH, Kelsey KT, Moore JH, Meng AC, Casella DP, Tosteson TD, Schned AR, Karagas MR (2006) Carcinogenesis 27:1030-7.
40.Moore JH (2003) Hum Hered 56:73-82.
41.Moore JH, Williams SM (2009) Am J Hum Genet, in press.
42.Theiler J, Eubank S, Longtin A, Galdrikian B, Farmer JD (1992) Physica D 58:77-94.
43.Moore JH (1999) Phys Med Biol 44:L11-2.

LOSS OF POST-TRANSLATIONAL MODIFICATION SITES IN DISEASE

SHUYAN LI
School of Informatics and Computing, Indiana University
Bloomington, IN 47408, U.S.A.
College of Chemistry and Chemical Engineering, Lanzhou University
Lanzhou, Gansu 730000, China

LILIA M. IAKOUCHEVA
Laboratory of Statistical Genetics, The Rockefeller University
New York, NY 10065, U.S.A.

SEAN D. MOONEY
Buck Institute for Age Research
Novato, CA 94945, U.S.A.

PREDRAG RADIVOJAC
School of Informatics and Computing, Indiana University
Bloomington, IN 47408, U.S.A.

Understanding and predicting molecular cause of disease is one of the major challenges for biology and medicine. One particular area of interest continues to be computational analyses of disease-associated amino acid substitutions. To this end, various studies have been performed to identify molecular functions disrupted by disease-causing mutations. Here, we investigate the influence of disease-associated mutations on post-translational modifications. In particular, we study the loss of modification target sites as a consequence of disease mutation. We find that about 5% of disease-associated mutations may affect known modification sites, either partially (4%) of fully (1%), compared to about 2% of putatively neutral polymorphisms. Most of the fifteen post-translational modification types analyzed were found to be disrupted at levels higher than expected by chance. Molecular functions and physiochemical properties at sites of disease mutation were also compared to those of neutral polymorphisms involved in the process of post-translational modification site disruption. Disease-associated mutations in the neighborhood of post-translationally modified sites were found to be enriched in mutations that change polarity, charge, and hydrophobicity of the wild-type amino acids. Overall, these results further suggest that disruption of modification sites is an important but not the major cause of human genetic disease.

1. Introduction

1.1. *Function of post-translational modifications*

Protein post-translational modifications are reversible or irreversible chemical alterations of a protein after its translation. They include covalent additions of particular chemical groups (e.g. phosphoryl), lipids (e.g. palmitic acid), carbohydrates (e.g. glucose) or even entire proteins (e.g. ubiquitin) to amino acid side chains, as well as the enzymatic cleavage of peptide bonds [1]. With some exceptions (e.g. hydroxylation), protein post-translational modifications occur at side chains that can act as either strong (C, M, S, T, Y, K, H, R, D, E) or weak (N, Q) nucleophiles, while the remaining residues (P, G, L, I, V, A, W, F) are rarely involved in covalent modifications of their side chains. Post-translational modifications frequently affect protein function via changes in the protein structure and dynamics. Alternatively, modified residue may be a part of a binding region directly recognized by a partner. For example, phosphotyrosines are known to be directly targeted by the SH2 domains [2] and acetyllysines are targeted by bromodomains [3]. Similarly, proteolitic cleavage is typically a part of degradation pathways. Biologically, post-translational modifications are involved in a number of activities such as regulation of gene expression, activation/deactivation of enzymatic activity, protein stability or destruction, mediation of protein-protein interactions etc. [1]. Whatever the molecular context, the major role of post-translational modifications is to enable signaling and regulatory mechanisms that modulate protein's cellular function.

Correspondence: Predrag Radivojac, 901 E. 10th St., Bloomington IN, 47408, U.S.A.; Email: predrag@indiana.edu. Availability: http://mutdb.org

There are more than 200 documented types of post-translational modifications, many of which were discovered only recently [4]. More interestingly, a large fraction of them are catalyzed by modifying enzymes. It is estimated that about 5% of the genes in *Homo sapiens* are modifying enzymes [1]. There are 518 kinases in the human genome and more than 150 phosphatases [5]. Similarly, the human genome also codes for around 600 E3 ubiquitinating ligases and 80 deubiquitinases [6]. These modifying enzymes are ubiquitous in all kingdoms of life, especially in eukayotes. For example, there are 1019 kinase- and 300 phosphatase-coding genes in *Arabidopsis thaliana* and even the yeast genome codes for 119 kinases [7]. However, despite the increasing recognition of their importance, the commonness and full functional repertoire of post-translational modifications are still unknown. The focus of this study is on the phenotypic effects of the disruption of post-translationally modified sites by the single amino acid substitution events.

1.2. *Mutation of post-translational modification sites may lead to disease*

There are a number of cases in which mutations of the post-translational target sites were found to be directly involved in disease. One example is a loss of N-linked glycosylation in the prion protein (PRNP), where amino acid substitution T183A was shown to be involved in autosomal dominant spongiform encephalopathy [8]. This particular variant causes numerous clinical symptoms such as early-onset dementia, cerebral atrophy, and hypometabolism. Interestingly, a wild-type form of PRNP was also found to be protease-resistant in the presence of the mutant. N-linked glycosylation occurs on asparagine residues in NX[ST] motifs, thus the loss of the threonine in the consensus sequence prevents the attachment of a carbohydrate. Modifications of the NX[ST] motif have previously been implicated in intracellular accumulation of PRNP *in vitro* [9]. Another example is a loss of acetylation sites in androgen receptor (AR). Loss of AR acetylation has been implicated in Kennedy's disease, an inherited neurodegenerative disorder. Here, amino acid substitution K630A or both K632A and K633A have been shown to cause a significant slowdown of ligand-dependent nuclear translocation [10]. Furthermore, the non-acetylated mutants misfold and form aggregates with several other proteins, including ubiquitin ligase E3, thus affecting proteosomal degradation. And yet another example involves serine phosphorylation in the period circadian protein homolog 2 protein (PER2). Mutation of S662 is associated with the familial advanced sleep phase syndrome, an autosomal dominant disorder with early sleep onset (around 7:30pm) and early awakening (around 4:30am), but normal sleep duration [11]. Biochemical studies have shown that phosphorylation of S662 affects phosphorylation (by casein kinase CKIε) of several other residues in PER2, resulting in an overall hypophosphorylation of PER2. Interestingly, creation of a negative charge by S662D or an excess of CKIε restores the phosphorylation patterns of PER2. The current working hypothesis regarding PER2 is that phosphorylation of S662 likely creates a recognition site for CKIε and triggers a cascade of downstream effects. However, functional roles of phosphorylated PER2 are still largely unknown [11].

In addition to the individual examples, systematic studies implicating post-translational modifications in disease are now facilitated by the rapid growth of databases containing disease-associated mutations, human polymorphisms, and also post-translational modifications. One of the first such studies was carried out by Wang and Moult who analyzed protein structures and concluded that a large majority of human inherited disease mutations affect protein stability [12]. Only a small percentage of amino acid substitutions were estimated to affect post-translational modifications and binding sites in general; however, only N-linked glycosylation was investigated. In addition, Wang and Moult studied only protein structures, whereas several types of post-translational modifications were shown to be preferentially occurring in the disordered protein regions [13-15]. Vogt et al. looked into the gain of N-linked glycosylation sites and their involvement in disease predicting that a number of disease associated mutations introduce changes in glycosylation patterns by creating NX[ST] motifs [16, 17]. Lee et al. [18] and Yang et al. [19] matched experimentally determined modification sites with amino acid substitutions from different databases and found 47 and 64 substitutions to affect post-translational modifications. In our previous work, we studied modification of confidently predicted phosphorylation sites affected by the somatic mutations and found that both gain and loss of phosphorylation target sites may be an active mechanism in cancer [20]. This study was

recently extended to include confident predictions of methylation, ubiquitination, and O-linked glycosylation, implicating all three modifications in disease [15, 21, 22].

1.3. *Outline of the study*

In this study, we adopt a simple strategy and analyze a larger number of post-translational modifications in the context of disease-associated and putatively neutral amino acid substitutions. Experimentally verified sites of post-translational modifications were searched against amino acid substitution databases with the goal of investigating whether (and in what ways) changes of post-translational modifications are affected by inherited and somatic disease mutations. We found that disease-associated mutations are enriched in the fraction of directly disrupted modification sites, but also those found in their close proximity. In contrast, the putatively neutral polymorphisms occur less frequently in the neighborhoods of the modification sites. Furthermore, we found that the sites of post-translational modifications were enriched in amino acid substitutions that change physicochemical properties of the wild-type amino acids.

2. Materials and Methods

2.1. *Data sets*

The data sets of post-translational modifications were collected from several public databases and the literature. We mined Swiss-Prot [23], Human Protein References Database (HPRD) [24], Phospho.ELM [25], Protein Data Bank (PDB) [26], O-GlycBase [27], PhosphoSite [28], and PhosphoPOINT [19]. Only modification types containing 50 or more instances were of interest, resulting in 15 different post-translational modifications from a number of different species. In total, these data sets contained 78,975 unique sites (Table 1).

Table 1. Summary of the data sets of post-translational modifications. All modifications were extracted from Swiss-Prot and HPRD. Glycosylation sites were also extracted from O-GlycBase and PDB. Phosphorylation sites were additionally extracted from PDB, Phospho.ELM, PhosphoSite, and PhosphoPOINT.

Post-translational modification	Total sites	Total proteins	Human sites	Human proteins
Phosphorylation	62,269	17,116	30,838	8,428
N-linked glycosylation	4,971	2,257	2,181	906
O-linked glycosylation	2,853	367	295	93
Acetylation	2,600	1,896	1,024	677
Amidation	2,163	1,339	44	30
Hydroxylation	1,301	251	211	29
Proteolytic cleavage	1,285	531	1,285	531
Methylation	911	407	430	143
Pyrrolidone carboxylic acid	728	590	78	74
Ubiquitination	516	353	266	196
Carboxylation	447	122	88	15
SUMOylation	381	201	319	160
Palmitoylation	328	200	163	88
Sulfation	229	145	80	38
Myristoylation	156	153	61	58

The data set of the inherited amino acid substitutions in humans (Disease-I) was assembled from the Human Gene Mutation Database (HGMD) [29] and Swiss-Prot. The data set of somatic mutations in cancer (Disease-S) was also collected from Swiss-Prot and several recent cancer gene resequencing projects reviewed by Lee et al. [30]. The sites already present in the Disease-I data set were removed from Disease-S. Finally, the putatively neutral polymorphisms (Neutral) were downloaded from the Swiss-Prot database. All polymorphisms found in the disease sets were removed. We assumed that only a small fraction of neutral polymorphisms may be involved in disease, that is, that the large majority of them are either neutral or have minor phenotypic effects. The data sets of amino acid substitutions are summarized in Table 2. In total, the set contained 73,463 amino acid substitutions from 12,987 proteins.

Table 2. Summary of the data sets of amino acid substitutions. The number of sites includes the set of unique positions of amino acid substitutions in a particular set.

Data set	Number of substitutions	Number of sites	Number of proteins	Source
Putatively neutral (Neutral)	29,190	28,864	10,416	Swiss-Prot
Disease, inherited (Disease-I)	40,512	33,416	2,605	HGMD, Swiss-Prot
Disease, somatic (Disease-S)	3,761	3,191	2,336	Swiss-Prot, Literature

2.2. *Matching post-translational modifications with amino acid substitutions*

In order to investigate the relationships between post-translational modifications and amino acid substitutions, different scenarios were considered. First, a set of human post-translational modifications was created by: (1) including only those sites that were experimentally identified in human proteins, and (2) mapping of 25-residue long fragments from any other species (modification site ±12 amino acids around it) to the human proteins such that all 25 residues were identical to the corresponding residues in the human protein. Clearly, in the latter case, the correctness of such modification sites is not guaranteed; however, an exact 25-residue fragment match is expected to be a strong indication of functional similarity. This is often true for the modifications where only local interaction exists with the modifying enzyme (e.g. kinases), however, in some other cases with long-range interactions (e.g. E3 ligase binding in ubiquitination) the assumption may be less likely to hold. The fragment length of 25 was chosen based on the phosphorylation data for which there is evidence of physical kinase-substrate binding within about 7-12 residues of the modification site [31]. We refer to the experimentally verified human modification sites as *true sites*, while the ones obtained by the exact fragment matches are referred to as the *homology sites*.

Two types of matching between amino acid substitutions and post-translational modifications were considered: (1) matches where the substitutions occurred at a modification site and (2) matches where the substitution site was in the neighborhood of the modification site (i.e. between residues −3 and +3). This matching was based on an assumption that a mutation can affect the post-translational modification if it is in the vicinity of the target residue. One such situation occurs with mutation R16C, which diminishes phosphorylation of S19, in human PTP synthase and causes hyperphenylalaninemia [32-34]. The situation where a substitution site and the modification site are at the same position is referred to as the *direct match*. A substitution site that is no more than 3 residues away from the modification site is referred to as the *neighborhood match*. An example of a neighborhood match to a homology site is shown in Figure 1.

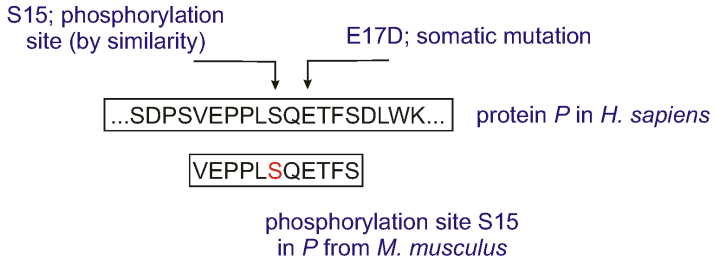


Figure 1. Stylized depiction of the matching between post-translational modification sites and amino acid substitutions. In this example, an 11-residue fragment corresponding to an experimentally verified phosphorylation site (serine, colored in red) from *M. musculus* is first matched to a homologous protein from *H. sapiens*. The corresponding residue S15 in the human protein is subsequently designated as phosphorylated (homology site). Finally, an observed amino acid substitution E17D, located within 3 residues from S15, is considered to impact the phosphorylation site (neighborhood match).

2.3. *Statistical analysis*

With all the matched sites of post-translational modifications and amino acid substitutions, two strategies were adopted to estimate statistical confidence of the observed trends. First, we used the t-test and the binomial test to estimate whether a certain group of amino acid substitutions (Neutral, Disease-I, Disease-S) is enriched or depleted in a particular modification. The hypergeometric test was used to estimate enrichment and depletion in functional terms for each of the categories. The amino acid property changes were studied for the mutations in the

neighborhood of modification sites (±3 residues). Three properties were investigated: side chain polarity (polar, non-polar), charge (positive, negative, neutral), and hydrophobicity (hydrophobic, hydrophilic), and the significance of those results was estimated using the t-test.

2.4. *Conservation index analysis*

Positional conservation was calculated using a commonly used conservation index AL2CO [35]. First, all 12,987 human proteins with mutations were searched against GenBank for the 500 best hits. These sequences were subsequently aligned using ClustalW program [36]. Then, the positional entropy by Henikoff and Henikoff [37] was calculated as the conservation index for all modification sites. The conservation index value was normalized to the 0-1 interval; the higher the value a position gets, the more conserved the position is.

3. Results

Using the two scenarios for obtaining post-translational modification sites (true and homologous sites) and using two strategies of matching them to the amino acid substitutions (direct and neighborhood matches), we analyzed amino acid substitutions in inherited disease, somatic disease, and neutral polymorphisms with respect to post-translational modifications.

3.1. *Disease-associated and neutral mutations affecting post-translational modifications*

The percentage of all amino acid substitutions that lie directly on or in the neighborhoods of modification sites in Disease-I, Disease-S, and Neutral data sets was investigated first. We found that direct and neighborhood mutations were in the vicinity of true and homology modification sites in 4.5% of cases in Disease-I, 3.1% of cases in Disease-S, and 2.1% of cases in Neutral data set (Figure 2). When only unique substitution sites were considered, these frequencies were 3.9%, 3.3%, and 2.1%, respectively (Figure 2). The most significant differences between the sets of inherited disease and neutral substitutions were detected in the cases of N-linked glycosylation (233 out of 306 in Disease-I; P = 1.6e–19), carboxylation (62/63; P = 2.4e–17), hydroxylation (68/72; P = 8.6e–16), acetylation (75/91; P = 4.9e–10), proteolytic cleavage (84/120; P = 1.9e–5), and O-linked glycosylation (32/41; P = 3.5e–4). Thus, the disease-associated mutations are more likely to affect post-translational modifications than the neutral substitutions (P = 5.6e–4).

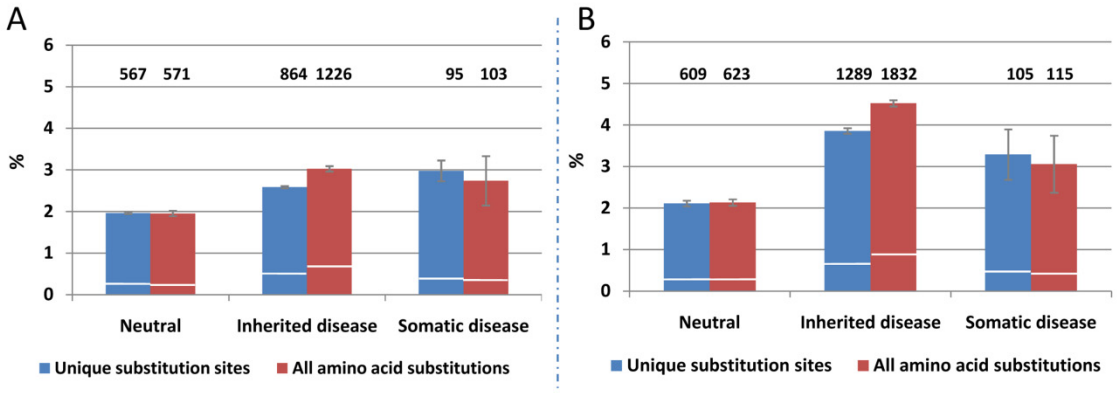


Figure 2. The fraction of disease-associated amino acid substitutions (inherited and somatic) that are assumed to affect post-translational modifications, compared to the fraction of neutral polymorphisms: (A) true modification sites and (B) homology modification sites. The white line in each bar separates direct matches from neighborhood matches. The error bars indicate 68% confidence intervals estimated by bootstrapping with 10,000 iterations.

Figure 3 shows the trends of enrichment and depletion of unique amino acid substitution sites that directly match experimentally verified (i.e. true) modification sites. The trend T was calculated as

$$T = \frac{f_{obs} - f_{exp}}{f_{obs} + f_{exp}}$$

where f_{obs} and $f_{exp} \neq 0$ are the observed and expected rates (relative frequencies) of substitutions that match modification sites, respectively. The trend is positive if $f_{obs} > f_{exp}$ and negative if $f_{exp} > f_{obs}$. $T = 1$ is the maximum value and involves a hypothetical situation with $f_{exp} = 0$ and $f_{obs} \neq 0$; $T = -1$ is the minimum value and indicates that $f_{obs} = 0$. Since inherited disease, somatic disease, and neutral polymorphisms contain 51%, 5%, and 44% of all substitution sites used in this study, f_{exp} was set to 0.51, 0.05, and 0.44 for the three data sets, respectively. The ratio of the three groups of amino acid substitutions also determines the null hypothesis that was used to calculate statistical significance of the observed enrichment or depletion. Trends similar to those observed in Figure 3 were present when the matching process was extended to homologous modification sites and neighborhood matches (Figure 4).

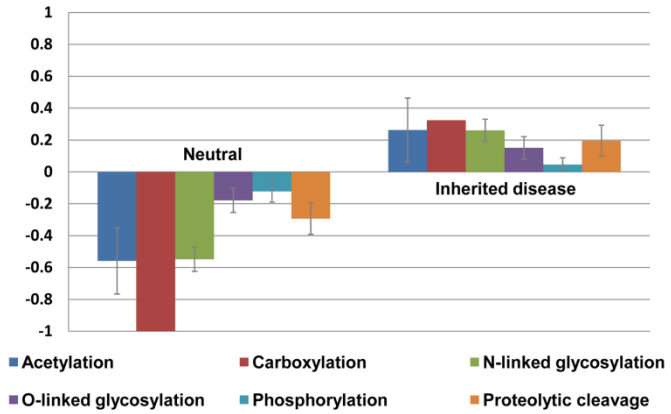


Figure 3. Trends that indicate whether a data set is enriched ($T > 0$) or depleted ($T < 0$) in direct matches between amino acid substitutions and true post-translational modifications. Only modifications with 5 matches or more are shown. The error bars indicate 68% confidence intervals estimated by bootstrapping with 10,000 iterations.

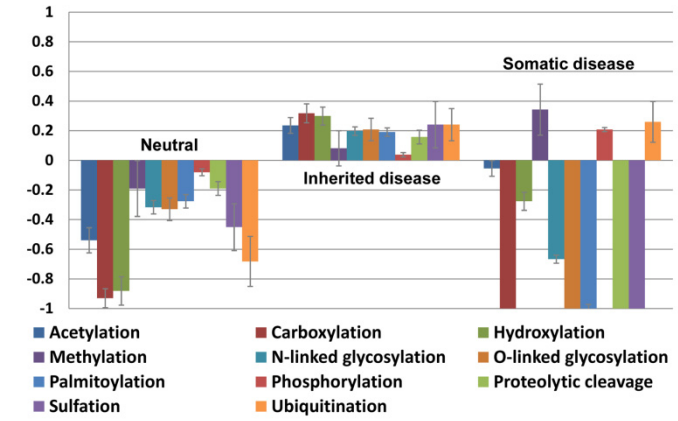


Figure 4. Trends that indicate whether a data set is enriched ($T > 0$) or depleted ($T < 0$) in direct and neighborhood matches between amino acid substitutions and true and homology post-translational modifications. Only modifications with 10 matches or more are shown. The error bars indicate 68% confidence intervals estimated by bootstrapping with 10,000 iterations.

Table 3 shows the observed numbers of amino acid substitutions matching post-translationally modified sites. The P-value for the positive trends T was calculated as

$$P = \sum_{i=k}^{K} \binom{K}{i} f_{exp}^{i} (1 - f_{exp})^{K-i}$$

where K is the total number of matches to any of the three substitution sets and k is the observed number of matches in a particular data set. The P-value for the negative trends was calculated by replacing k by 0 and K by k in the limits of the summation operator.

Table 3. The observed rates of matches between amino acid substitutions and true and homology post-translational modification sites. Example: for N-linked glycosylation, out of 31 mutations that directly matched N-linked glycosylation sites, 4 were observed in the neutral set (Neutral) and 27 in the inherited disease set (Disease-I). The expected number of matches were 13.6 and 15.8, respectively. Blue color indicates lower-than-expected observed frequencies; red indicates higher-than-expected observed frequencies. P-values were calculated using the binomial distribution. The Bonferroni-corrected value of 5.6e–4 (n = 90) was used to assign statistical significance for each group.

Post-translational modification	Direct match; true sites						Direct and neighborhood match; homology sites					
	Neutral		Disease-I		Disease-S		Neutral		Disease-I		Disease-S	
	f_{obs}	P	f_{obs}	P	f_{obs}	P	f_{obs}	P	f_{obs}	P	f_{obs}	P
Phosphorylation	47/136		76/136		13/136		457/1218	2.0e–6	670/1218		91/1218	5.3e–5
N-linked glycosylation	3/28	2.2e–4	27/31	2.7e–5	0/31		70/306	8.3e–15	233/306	1.6e–19	3/306	1.8e–4
O-linked glycosylation	4/13		9/13		0/13		10/45		35/45	2.1e–4	0/45	
Acetylation	1/8		7/8		0/8		12/91	2.4e–10	75/91	4.5e–10	4/91	
Amidation	1/1		0/1		0/1		2/4		2/4		0/4	
Hydroxylation	0/2		1/2		1/2		2/72	1.1e–15	68/72	6.6e–16	2/72	
Proteolytic cleavage	7/29		22/29		0/29		36/120		84/120	1.9e–5	0/120	
Methylation	0/4		4/4		0/4		6/20		12/20		2/20	
Pyrrolidone carboxylic acid	1/1		0/1		0/1		2/2		0/2		0/2	
Ubiquitination	0/0		0/0		0/0		2/24	1.7e–4	20/24		2/24	
Carboxylation	0/29	4.8e–8	29/29	3.4e–9	0/29		1/63	6.3e–15	62/63	2.4e–17	0/63	
SUMOylation	0/0		0/0		0/0		2/7		4/7		1/7	
Palmitoylation	0/0		0/0		0/0		4/16		12/16		0/16	
Sulfation	0/3		3/3		0/3		2/12		10/12		0/12	
Myristoylation	0/1		1/1		0/1		1/3		2/3		0/3	

Figure 3, Figure 4, and Table 3 indicate that the substitutions associated with inherited disease affect the sites of post-translational modifications with frequencies higher than expected by chance. In contrast, putatively neutral polymorphisms affect modification sites with lower-than-expected frequencies.

3.2. *Conservation index analysis*

It has been widely studied that disease-associated mutation sites are more evolutionarily conserved than human polymorphic sites [38]. Here, we analyzed the conservation of the post-translationally modified sites for which there are known amino acid substitutions either at the modification site itself or in its neighborhood. Not so surprisingly, we find that the modification sites directly matching disease-associated substitution sites are more conserved than those matching neutral polymorphic sites (Figure 5A). However, post-translational modifications lying in the neighborhood (±3 residues) of inherited disease mutations are also more conserved than the modification sites corresponding to the neutral polymorphisms, with a similar margin. On the contrary, the conservation of somatic mutations is significantly lower when the neighborhoods of modification sites were considered.

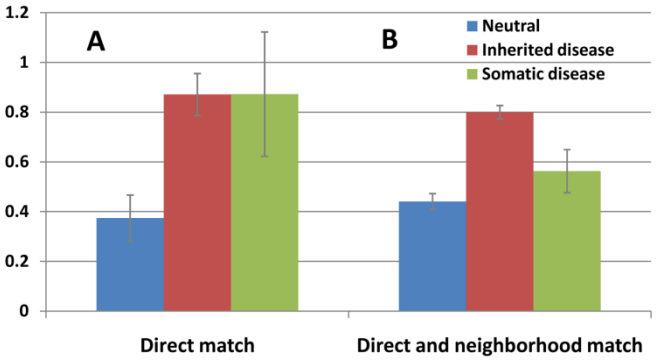


Figure 5. Sequence conservation index for post-translational modification sites matching disease-associated and neutral substitution sites: (A) direct match and (B) direct and neighborhood match. The conservation index was calculated as described in Section 2.4, with higher numbers indicating higher conservation. The error bars indicate 68% confidence intervals estimated by bootstrapping with 10,000 iterations.

3.3. *Gene function analysis*

To further study the impact of amino acid substitutions that occur in the vicinity of the modification sites, we analyzed gene functions of all types of post-translational modifications. The genes were first separated into the gene set containing disease mutations (inherited and somatic) and the gene set containing neutral polymorphisms (the two sets of genes were overlapping). Each of the sets was further split into the set of genes where amino acid substitutions impact modifications sites and the remaining genes. Then, the gene enrichment analysis of the two pairs of data sets was performed using the GOstat software [39]. It is important to understand that the set of disease-associated mutations impacting modification sites was compared to the remaining set of genes containing disease mutations in order to avoid biases that correspond to the disease genes [40]. In this way, we assume that it may be possible to identify molecular and cellular functions that were disrupted by the disease mutations or those that are regulated by post-translational modifications and related to minor phenotypic variations. The results of this analysis, using the Gene Ontology [41] category molecular function, are shown for the phosphorylation data set (Figure 6).

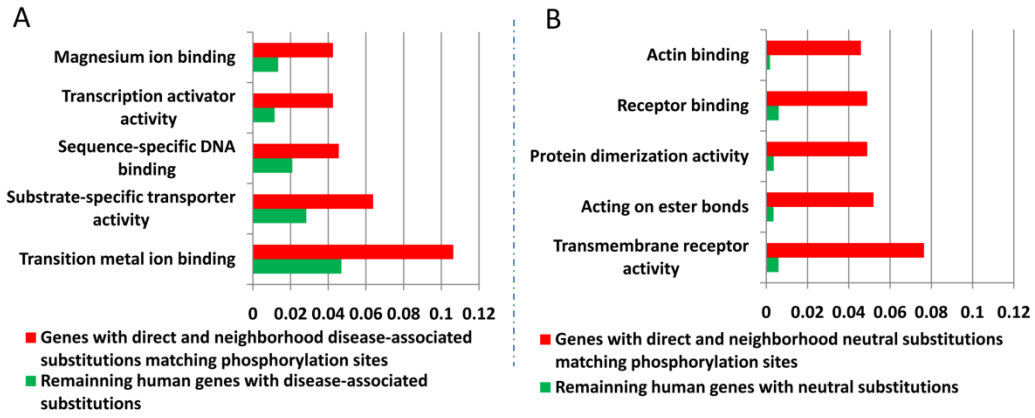


Figure 6. GO enrichment analysis between genes containing mutations in the vicinity of the predicted phosphorylation sites vs. the remaining genes from a selected data set. The analysis was done separately for genes containing disease-associated mutations and genes containing neutral polymorphisms. Top 5 molecular functions are shown, where the upper (red) bars indicate functions enriched in the set with phosphorylation site disruption, while the lower (green) bars indicate functions for the remaining set of genes from the group.

Interestingly, both sets of genes (with disease related mutations and with neutral polymorphisms) in the vicinity of phosphorylation sites have significant enrichment in different molecular functions. For example, kinase, transferase, and signal transduction activities are significantly enriched in the disease-associated set, whereas RNA binding, transcription factor and receptor activities are significantly enriched in the neutral substitutions set. However, both

sets are enriched in important molecular functions, thereby suggesting that both disease and neutral substitutions in the vicinity of phosphorylation sites, may have an impact on protein function.

3.4. *Amino acid properties of substitutions in the vicinity of modification sites*

Next, we analyzed physicochemical properties of the amino acid substitutions affecting post-translational modifications. Polarity, charge, and hydrophobicity were chosen for this analysis. These properties were studied for the amino acid substitutions occurring directly and in the neighborhood (±3 residues) of all types of modified sites. Figure 6 shows the enrichment and depletion of the observed changes in all three data sets. We observed that the inherited disease mutations are enriched in the change of all three properties for several modification types. On the other hand, neutral mutations are depleted in such changes. Somatic mutations do not show significant signals potentially due to the small size of the data set.

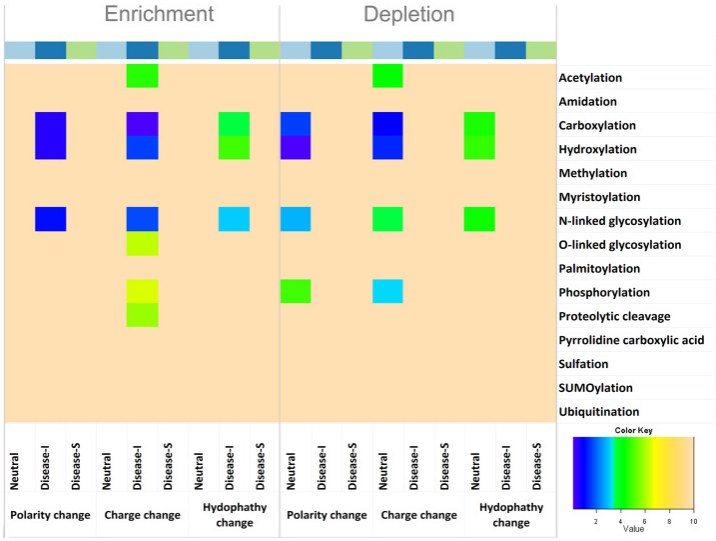


Figure 7. Enrichment and depletion of types of amino acid substitutions observed in the sets of disease-associated substitutions and putatively neutral polymorphisms. P-values were calculated using the binomial distribution. The plotted value is $10 + \log_{10}(P + 1 \cdot 10^{-10})$, where P is the P-value. The darker colors indicate values with lower P-values. All values below the Bonferroni-corrected value 3.7e–4 were set to 1.

4. Discussion

When personalized medicine is the next frontier for scientists, industry, and the general population, it is important to develop computational approaches that can lead to a better understanding of the etiology of disease. Integration of genetic and molecular information is a sensible step in this direction because it provides a structural and functional perspective to the human variation data.

In this study, we analyzed disease-associated and putatively neutral amino acid substitution data and found that about 4.5% of amino acid substitutions (3.9% of unique sites) may affect protein function through disruption of post-translational modifications. On the other hand, about 2.1% of neutral polymorphisms may be affecting post-translational modifications. These numbers further indicate that post-translational modifications are not the majority cause of human genetic disease. However, we have still found 238 post-translationally modified sites in human proteins whose mutation was causative of disease. In total, 1,289 modification sites were found to be in the close proximity to the inherited disease mutations and represent candidates for further experimental verification.

Given our data, there are several problems that could have lead to the ascertainment bias. For example, our data set of post-translational modifications was heavily skewed towards phosphorylation (79%), where mass spectrometry techniques have lead to a recent explosion in the number of identified sites. On the other hand, it may be argued that the modifications not identified using high-throughput methods may be more likely to be disease-

relevant. It is also unclear whether the sets of inherited disease data are representative since it may be expected that genetic-association studies are more successful in identifying markers of monogenic diseases or familiar forms of complex diseases. Finally, the set of neutral polymorphisms is probably contaminated with yet undiscovered disease mutations and has not been controlled for population biases.

We also analyzed the enrichment and depletion of amino acid substitutions for each post-translational modification and found that most follow similar trends when inherited disease is compared to the neutral polymorphisms. These trends held for both experimentally verified modification sites and those transferred by homology. In the case of somatic mutations, we observed some interesting cases as well. For most examples, we have not found matches between post-translational modifications and observed somatic mutations. However, in the cases of methylation, phosphorylation and ubiquitination, there was an increased trend of disruption of post-translational modifications. Previous work has already addressed disruption of confidently predicted phosphorylation sites in cancer [20]. Thus, the correspondence between actual sites and somatic mutations found in this study further supports this hypothesis.

While direct disruption of post-translational modifications is likely to have functional implications, the partial disruption of modified sites has a potential to lead to subtle phenotypic effects that may be more dependent on the variation present in other genes before causing organism-wide dysregulation. We believe that such changes are particularly fitting to the framework of complex disease and interaction between genetic and environmental factors.

Acknowledgments

This work was supported by the NIH grant R01LM009722-01 to SDM, NIH grant R21CA113711 to LMI, and NSF grant DBI-0644017 to PR.

References

1. Walsh, C.T., Posttranslational modification of proteins: expanding nature's inventory. 2006, Englewood, CO: Roberts and Company Publishers.
2. Felder, S., et al., SH2 domains exhibit high-affinity binding to tyrosine-phosphorylated peptides yet also exhibit rapid dissociation and exchange. *Mol Cell Biol*, 1993. **13**(3): p. 1449-55.
3. Yang, X.J., Lysine acetylation and the bromodomain: a new partnership for signaling. *Bioessays*, 2004. **26**(10): p. 1076-87.
4. Mann, M., O.N. Jensen, Proteomic analysis of post-translational modifications. *Nat Biotechnol*, 2003. **21**(3): p. 255-61.
5. Manning, G., et al., The protein kinase complement of the human genome. *Science*, 2002. **298**(5600): p. 1912-34.
6. Komander, D., et al., Breaking the chains: structure and function of the deubiquitinases. *Nat Rev Mol Cell Biol*, 2009. **10**(8): p. 550-63.
7. Wang, D., et al., Systematic trans-genomic comparison of protein kinases between Arabidopsis and Saccharomyces cerevisiae. *Plant Physiol*, 2003. **132**(4): p. 2152-65.
8. Grasbon-Frodl, E., et al., Loss of glycosylation associated with the T183A mutation in human prion disease. *Acta Neuropathol*, 2004. **108**(6): p. 476-84.
9. Rogers, M., et al., Intracellular accumulation of the cellular prion protein after mutagenesis of its Asn-linked glycosylation sites. *Glycobiology*, 1990. **1**(1): p. 101-9.
10. Thomas, M., et al., Androgen receptor acetylation site mutations cause trafficking defects, misfolding, and aggregation similar to expanded glutamine tracts. *J Biol Chem*, 2004. **279**(9): p. 8389-95.
11. Toh, K.L., et al., An hPer2 phosphorylation site mutation in familial advanced sleep phase syndrome. *Science*, 2001. **291**(5506): p. 1040-3.
12. Wang, Z., J. Moult, SNPs, protein structure, and disease. *Hum Mutat*, 2001. **17**(4): p. 263-70.
13. Iakoucheva, L.M., et al., The importance of intrinsic disorder for protein phosphorylation. *Nucleic Acids Res*, 2004. **32**(3): p. 1037-1049.

14. Daily, K.M., et al.. Intrinsic disorder and protein modifications: building an SVM predictor for methylation. IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), p. 475-481, 2005. San Diego, California, U.S.A.
15. Radivojac, P., et al., Identification, analysis and prediction of protein ubiquitination sites. *Proteins*, 2009.
16. Vogt, G., et al., Gains of glycosylation comprise an unexpectedly large group of pathogenic mutations. *Nat Genet*, 2005. **37**(7): p. 692-700.
17. Vogt, G., et al., Gain-of-glycosylation mutations. *Curr Opin Genet Dev*, 2007. **17**(3): p. 245-51.
18. Lee, T.Y., et al., dbPTM: an information repository of protein post-translational modification. *Nucleic Acids Res*, 2006. **34**(Database issue): p. D622-7.
19. Yang, C.Y., et al., PhosphoPOINT: a comprehensive human kinase interactome and phospho-protein database. *Bioinformatics*, 2008. **24**(16): p. i14-20.
20. Radivojac, P., et al., Gain and loss of phosphorylation sites in human cancer. *Bioinformatics*, 2008. **24**(16): p. i241-7.
21. Mort, M.E., et al., In silico functional profiling of human disease-associated and polymorphic amino acid substitutions. In review.
22. Li, B., et al., Automated inference of molecular mechanisms of disease from amino acid substitutions. *Bioinformatics*, 2009.
23. Bairoch, A., et al., The Universal Protein Resource (UniProt). *Nucleic Acids Res*, 2005. **33 Database Issue**: p. D154-9.
24. Keshava Prasad, T.S., et al., Human Protein Reference Database--2009 update. *Nucleic Acids Res*, 2009. **37**(Database issue): p. D767-72.
25. Diella, F., et al., Phospho.ELM: a database of experimentally verified phosphorylation sites in eukaryotic proteins. *BMC Bioinformatics*, 2004. **5**(1): p. 79.
26. Berman, H.M., et al., The protein data bank. *Nucleic Acids Res*, 2000. **28**(1): p. 235-242.
27. Gupta, R., et al., O-GLYCBASE version 4.0: a revised database of O-glycosylated proteins. *Nucleic Acids Res*, 1999. **27**(1): p. 370-2.
28. Biondi, R.M., Phosphoinositide-dependent protein kinase 1, a sensor of protein conformation. *Trends Biochem Sci*, 2004. **29**(3): p. 136-42.
29. Stenson, P.D., et al., The Human Gene Mutation Database: 2008 update. *Genome Med*, 2009. **1**(1): p. 13.
30. Lee, W., et al., Analytical methods for inferring functional effects of single base pair substitutions in human cancers. *Hum Genet*, 2009.
31. Songyang, Z., et al., Use of an oriented peptide library to determine the optimal substrates of protein kinases. *Curr Biol*, 1994. **4**(11): p. 973-982.
32. Oppliger, T., et al., Structural and functional consequences of mutations in 6-pyruvoyltetrahydropterin synthase causing hyperphenylalaninemia in humans. Phosphorylation is a requirement for in vivo activity. *J Biol Chem*, 1995. **270**(49): p. 29498-506.
33. Scherer-Oppliger, T., et al., Serine 19 of human 6-pyruvoyltetrahydropterin synthase is phosphorylated by cGMP protein kinase II. *J Biol Chem*, 1999. **274**(44): p. 31341-8.
34. Thony, B., et al., Hyperphenylalaninemia due to defects in tetrahydrobiopterin metabolism: molecular characterization of mutations in 6-pyruvoyl-tetrahydropterin synthase. *Am J Hum Genet*, 1994. **54**(5): p. 782-92.
35. Pei, J. and N.V. Grishin, AL2CO: calculation of positional conservation in a protein sequence alignment. *Bioinformatics*, 2001. **17**(8): p. 700-12.
36. Thompson, J.D., et al., CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, 1994. **22**(22): p. 4673-4680.
37. Henikoff, S., J.G. Henikoff, Position-based sequence weights. *J Mol Biol*, 1994. **243**(4): p. 574-8.
38. Ng, P.C., S. Henikoff, Predicting deleterious amino acid substitutions. *Genome Res*, 2001. **11**(5): p. 863-74.
39. Beissbarth, T. and T.P. Speed, GOstat: find statistically overrepresented Gene Ontologies within a group of genes. *Bioinformatics*, 2004. **20**(9): p. 1464-5.
40. Dalkilic, M.M., et al., From protein-disease associations to disease informatics. *Front Biosci*, 2008. **13**: p. 3391-407.
41. Ashburner, M., et al., Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 2000. **25**(1): p. 25-29.

DETECTING GENOME-WIDE HAPLOTYPE POLYMORPHISM BY COMBINED USE OF MENDELIAN CONSTRAINTS AND LOCAL POPULATION STRUCTURE

XIN LI

YIXUAN CHEN

JING LI*

Department of Electrical Engineering and Computer Science
Case Western Reserve University
Cleveland, OH 44106, United States
E-mail: {xin.li2,yixuan.chen,jingli}@case.edu

Data from current gene-disease association studies motivate changes to existing haplotype inference methodologies. Many datasets are now comprised of both pedigree and population data so it is desirable to incorporate both sources of information when inferring haplotypes. The availability of high-density SNP data also makes it possible to determine and use the precise locations of recombination events. Our proposed method reconstructs haplotype structure on a genome-wide level by jointly using the information from the Mendelian law of inheritance and local population structure. The method combines in one framework new techniques of recombination event detection, maximum likelihood optimization of population haplotype diversity and our previous algorithm of zero-recombinant haplotype reconstruction. Experiments on both real and simulated datasets prove the efficiency and accuracy of our approach in reconstructing the haplotype structure. Our method makes it possible to reveal the haplotypic variation on a genome-wide level.

Keywords: haplotypic variation, genetic linkage, Mendelian law

1. Introduction

Haplotypes are mostly obtained by computational methods from genotype data instead of directly by molecular methods due to the high cost of the current technology. Haplotypes if corrected inferred provide exact information on the linkage of SNPs, which is of substantial importance in detecting gene-disease association.[1,4,15] Typically haplotype inference from pedigree and population data is performed separately. Methods on pedigree data[12,18] make use of the Mendelian law of inheritance and some parsimony criteria on the number of recombination events. Due to the enrichment of constraints in pedigrees, such methods are usually fast. Methods on population data such as fastPHASE[16] and Beagle[5] make use of the clustering property of haplotypes in the population over short regions. Due to the enumerative nature of these methods, they are usually slow. With respect to haplotyping accuracy, previous analyses[13] have demonstrated that using family constraints alone achieves better accuracy than using population information alone. On the other hand, population information works better than family constraints in imputing missing genotypes. Therefore, it is most desirable to jointly use the family and population information. Beagle recently added the function to process data sets of many trios and parent-offspring pairs.[6] The authors reported that Beagle achieved extreme accurate results on data sets of trios compared to data sets of unrelated individuals due to the use of the rules of Mendelian inheritance.

Linkage analysis[7,9] can actually yield the maximum likelihood haplotype configuration in terms of both family constraints and population haplotype frequency by enumerating every possible inheritance pattern and every possible allele assignment. However the time complexity of this approach is exponential to either the number of markers or the pedigree size, thus it is infeasible for any reasonably large dataset. To avoid exhaustive enumeration, it is critical to represent the set of all compatible family configurations in a compact form. Li and Li[14] found that by assuming no recombination it is possible to represent the set of family configurations as a linear span of variables that can be found by solving a linear system of binary variables representing inheritance and phase. However, to apply this method to whole-genome data, we must first identify recombination positions in each family such that we can segment the chromosomes into recombinant-free regions.

*Corresponding author.

Furthermore, to find an optimal solution using population haplotype frequency, we need a computationally feasible way to search the solution space since the number of possible haplotypes is potentially exponential to the number of markers. In this paper, we present a framework combining the efforts of recombination detection, zero-recombinant haplotype inference, and local haplotype structure clustering to jointly use the family and population information. Our method makes it possible to accurately reveal the haplotype structure in human populations on a genome-wide level. The algorithm is implemented in $C++$ and is freely available upon request.

2. Methods

The overall flow of the method **MML** (**M**endalian Constrained **M**aximum **L**ikelihood) is staged in three steps as illustrated in Algorithm 2.1.

Algorithm 2.1 MML

(1) Infer recombination positions for each family and each chromosome. Partition the chromosomes according to recombination positions.
(2) On each pedigree, for each of the zero-recombinant segments, apply DSS[14] (our previously developed algorithm to handle Mendelian constraints) to establish the solution space under Mendelian and zero-recombinant constraints.
(3) Search the solution space (obtained in (2)) for the optimal solution with maximum likelihood based on population haplotype frequency.

In step 1, we infer recombination positions in each nuclear family of the pedigree by analyzing identical by descent (IBD) status of alleles between each sibling pair. Based on the inferred recombination positions, we partition chromosomes into segments such that every segment is recombinant-free. In step 2, we derive all possible configurations of a pedigree under Mendelian and zero-recombinant constraints for each recombinant-free segment obtained in step 1. This is done by using our previous algorithm DSS.[14] DSS can output a compact description of all compatible solutions as a linear space. In step 3, we use haplotype frequencies in the population to identify the optimal haplotype configuration of each pedigree. We will describe step 1, step 2 and step 3 in Sec. 2.1, Sec. 2.2 and Sec. 2.3 respectively.

2.1. *Detect Recombination Events in Families with Dense Markers*

Recombination events are implied if a common inheritance vector for a segment of loci that satisfies Mendelian constraints cannot be found. Typically, there is uncertainty as to how many recombination events occur and at which loci or in which individuals these events occur. Usually, such parsimony criteria as minimum number of recombinants[12] are used to find a possible assignment. However, with the availability of densely marked data, we can almost always fix the inheritance vector within each zero-recombinant region due to the enrichment of Mendelian constraints. Consequently, we can also develop special techniques to localize the recombination positions with minimal ambiguity.

For each nuclear family, we look at the IBD status of the alleles and its sibling pairs to detect a recombination position. The change of IBD status from one locus to another indicates a change in the inheritance pattern, that is, a recombinant. Loci of a father (similarly for a mother) can be divided into three categories depending on their informativeness in determining the paternal IBD status of a sibling pair.

(1) informative: he is heterozygous, and the phases of both children are determined at this locus.
(2) semi-informative: he is heterozygous, and at least one of the children is not phased at this locus.
(3) non-informative: he is homozygous at this locus.

In situation (1), since the father is heterozygous, the IBD status and the IBS (identical by state) status of the paternal alleles of a sibling pair are equivalent. In situation (2), the IBD status of the paternal alleles is not determined, but it is dependent on the IBD status of the maternal alleles. If we can somehow resolve the IBD status of the maternal alleles at this locus, we can also infer the IBD status of the paternal alleles. Note that in this situation, the mother must be heterozygous, otherwise all children would be phased. In situation (3), this locus provides no information about the paternal IBD status of any of the sibling pairs.

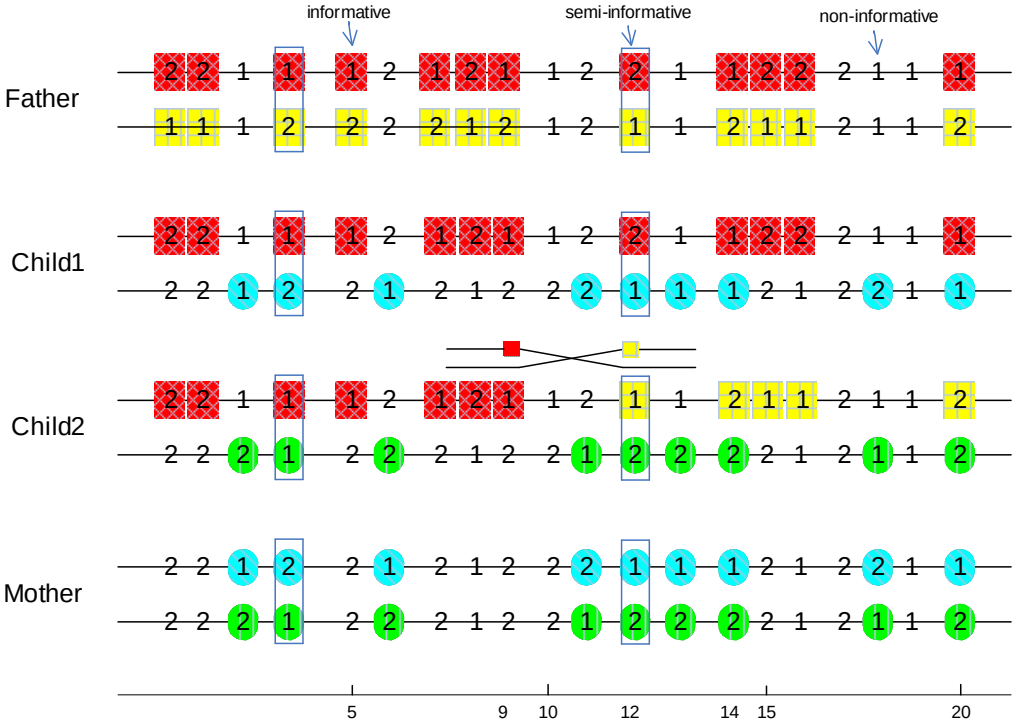


Fig. 1. A segment of a chromosome from a nuclear family with 2 children. Colored nodes are informative alleles which have determined IBD status between these two siblings, with squares and circles representing paternal and maternal alleles respectively. Remaining alleles are non-informative. At each locus, alleles colored the same are identical by descent(IBD). A frame around a pair of alleles indicates that they are semi-informative, and this pair of alleles infers IBD between two siblings if in the context of the informative loci nearby. From the IBD status between siblings, we can infer a paternal recombination event between the 9th and 12th locus, but with the ambiguity whether it happens in Child1 or Child2. We suppose the recombinant is in Child2 for illustration purposes.

Informative loci give a narrow-spaced probing on the IBD status of the whole chromosome. We can detect recombination events by observing the change of IBD status of alleles among these informative loci. By doing so, however, we may miss possible double recombination events that do not manifest a change in the IBD status between two nearby informative loci. If we assume markers are dense, however, the possibility of a double recombination event within a short distance is negligible. Fig. 1 shows an example on how to detect recombination positions in a nuclear family. By using informative markers, we could infer a paternal recombination event between the 9th and 14th locus.

Semi-informative loci can help further localize the recombination position because it is almost impossible for a paternal and a maternal recombination event to occur coincidentally within a short region. If a semi-informative locus falls between two informative loci indicating different paternal IBD status, we can assume no recombination on the maternal side and let its maternal IBD status follow that of its surrounding informative loci. By assuming the maternal IBD status, we can now infer the paternal IBD status for this semi-informative locus. For example, in Fig. 1, at the 12th loci, by assuming that Child1 and Child2 are not IBD for their

maternal alleles, we infer that the sibling pair are also not IBD for their paternal alleles, so that we could refine the recombination position to be between the 9th and 12th locus.

Since ambiguous intervals of recombination events now only contain non-informative markers which are compatible with any inheritance pattern, we may pick any position within such intervals to partition a chromosome into recombinant-free segments. Notice that for non-informative loci, the phases of all family members are actually fixed, and thus the choice of a recombination position will not influence the final haplotype configuration.

To determine the individual in which the recombination event actually occurs, we can look at the IBD status of all sibling pairs. If we observe that the IBD status changes between a specific child i and any of the other children, while there is no change among these children themselves, then child i carries the recombinant. However, if the nuclear family has only two children, then the ambiguity is unresolvable in this way. Notice that the assignment of recombination to a different child will result in a different haplotype configuration in the parent. Therefore in this situation, we can use population haplotype frequency to suggest a most probable assignment.

2.2. *Establish the Solution Space under Mendelian Constraints*

We can use two types of binary variables: p variables and h variables, to encode the Mendelian constraints in a pedigree. A p variable indicates the phase of two alleles of an individual. $p^a = 0$ means that the smaller-numbered allele ("1") of individual "a" is of paternal source, $p^a = 1$ means it is of maternal source. An h variable indicates the inheritance relationship between a parent-child pair, $h^{ab} = 0$ means that the paternal allele of individual "a" is transmitted to individual "b", $h^{ab} = 1$ means that the maternal allele is transmitted. Fig. 2 gives an example of the relationship between the p variables and the h variable of a parent-child pair under Mendelian constraints. As established in previous work,[14,18] Mendelian law can be expressed as linear equations of h variables and p variables.

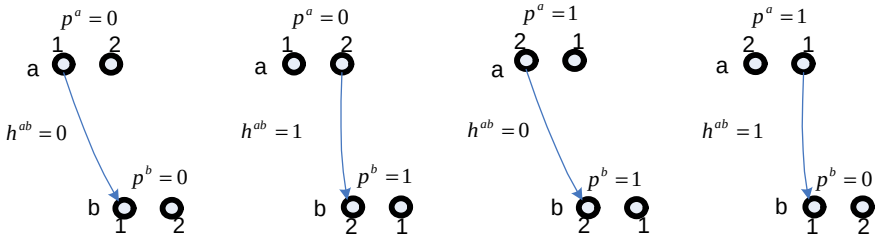


Fig. 2. Individual a is the father of individual b. In the situation where both a and b are heterozygous, relationship between p variables and the h variable can be expressed as $p^b = p^a + h^{ab}$.

If we assume no recombination within a certain number of loci, the h variable between a parent-child pair, which indicates the inheritance patterns, should be the same for each of these loci. In this case, we can put constraints on h variables from different loci together to form a single linear system. Li and Li[14] discover an almost linear time algorithm, DSS, to obtain a general solution to such a system. Here, a general solution means a description of all solutions as a linear span of variables.

The establishment of a general solution is important because it facilitates the search in the solution space for particular solutions to satisfy specific properties. The freedom in the solution space can be partitioned into two parts: the freedom of the inheritance vector (all h variables) and the freedom of the allele assignment (all p variables) under a fixed inheritance vector. Experiments[14] have shown that the inheritance vector is usually fixed for a segment of 100 or more loci. Once the inheritance is determined, the relationship between alleles of different individuals is determined with only 1 degree of freedom (if all members of the pedigree are heterozygous) or no degrees of freedom (if one or more members are homozygous). Fig. 3(a) shows an example

for the first situation. In the case of a missing genotype, there might be an increase in degrees of freedom (Fig. 3(b)). By applying the Mendelian and zero-recombinant constraints, we can greatly reduce the search space for finding the maximum likelihood solution using population local structure, which will be discussed in Sec. 2.3.

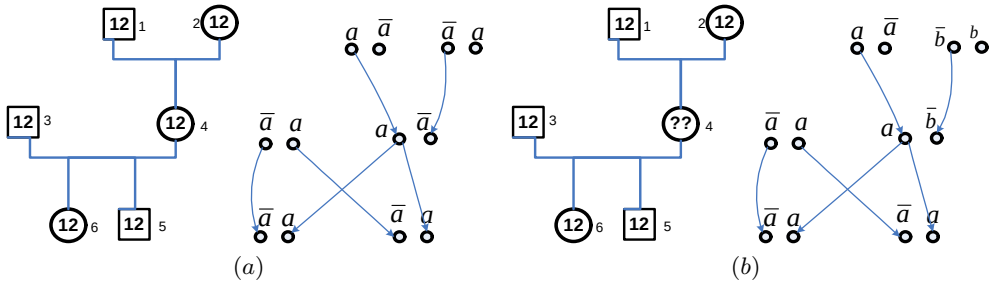


Fig. 3. (a) A pedigree of 6 individuals. All individuals are heterozygous with genotype "12" at a certain locus. For this fixed inheritance vector, the relationship between alleles of different individuals is determined. a is a variable denoting the status of an allele and $\bar{a}$ is its complementary status. (b) Same pedigree and inheritance vector as (a), but at a different locus with genotype missing at individual 4. In this case, there are two degrees of freedom represented by variables a and b.

2.3. *Maximum Likelihood Solution Based on Population Haplotype Frequency*

The inheritance vector (h variables between each parent-child pair) specifies how founder haplotypes are transmitted to every descendant of a pedigree and the configuration of a pedigree is fully determined by the inheritance vector and the founder haplotypes. If the inheritance vector is fixed, the likelihood of a configuration of a pedigree is simply the product of founder haplotype probabilities.

In Sec. 2.2, we describe how Mendelian and zero-recombinant constraints provide a small candidate set of all possible haplotype configurations for each pedigree. Next, we need to pick a solution of maximum likelihood from this candidate set based on haplotype frequencies. Since the actual haplotype frequencies in the population are unknown, we use an EM (Expectation Maximization) procedure to find the optimal solution. The procedure is described in Algorithm 2.2. The initial pool of haplotype frequencies is generated by randomly sampling from founder haplotypes within the solution space of each pedigree. In step (2), we search the solution space for an optimal configuration with the highest likelihood. In step (3), we update the haplotype frequency pool only with the optimal solution of each pedigree. It is different from conventional EM methods, where all possible solutions are updated into the pool weighted by their current likelihood. By adopting such an approximation, we can significantly hasten the optimization process by not traversing the entire solution space.

Algorithm 2.2 Haplotype Frequency EM

(1) Build the initial pool of haplotype frequencies by randomly sampling from the solution space of each pedigree.
repeat
(2) Find the optimal solution with maximum likelihood based on the current pool.
(3) Update the pool with the haplotype frequencies of optimal solutions obtained in (2).
until convergence is achieved

2.3.1. *Probabilistic prefix tree for fast branch-and-bound optimization*

We create a data structure called "probabilistic prefix tree" to facilitate the search of the optimal configuration in the solution space. A probabilistic prefix tree is essentially a binary search tree which encodes the frequencies

of each haplotype and their prefixes. It provides quick indexing for haplotype frequencies and can be updated dynamically using conventional binary search tree techniques. Each leaf node in the tree represents a haplotype and each internal node represents a prefix. The frequencies of internal nodes can be generated by simply summing up the frequencies of all leaf nodes of its subtree. Fig. 4 shows an example of a probabilistic prefix tree.

As discussed in Sec. 2.2, for a fixed inheritance vector, the relationship between alleles of different family members is fixed at each locus. On a pedigree of n founders and m markers, we do a depth first search from locus 1 to locus m of a haplotype, where for each locus we pick an assignment for all $2n$ founder alleles if there is one or more degrees of freedom. Meanwhile, we calculate the likelihood of the pedigree up to the current locus which is the product of frequencies of the founder haplotype prefixes ending at locus i: $\prod_{j=1..2n} freq(h_i^j)$. Since the frequency of any haplotype prefix is greater than that of the entire haplotype, if the likelihood drops below the bound, we backtrack for there is no possibility of a better solution. Otherwise, we move to the next locus until we reach m. If we achieve a higher likelihood, we replace the bound with the new likelihood and record the current best configuration.

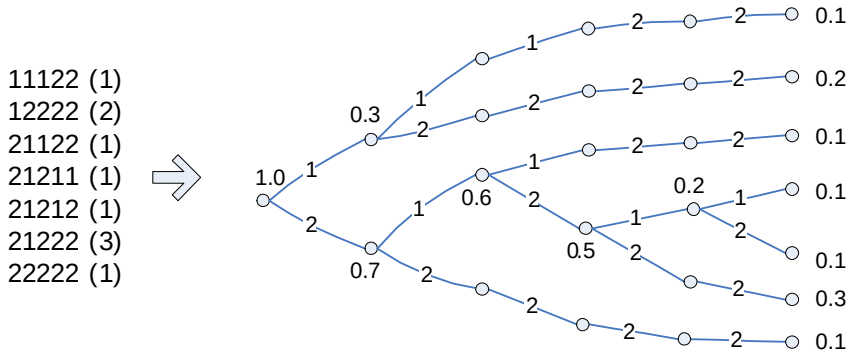


Fig. 4. On the left is the haplotypes and their count (in brackets) in the population. On the right is the probabilistic prefix tree after adding all these haplotype to an empty tree. Some nodes are annotated with the normalized frequencies of the corresponding haplotypes or haplotype prefixes.

3. Experimental Results

3.1. *Detect Recombination Events and Haplotype Diversity*

We use MML to analyze haplotype polymorphisms in a real human population. There are 32250 markers spanning a region of 170 million base pairs on chromosome 6, with an average marker interval distance of 5kb. Missing genotype rate is 0.12% and typing error rate (as reflected by Mendelian inconsistency) is 0.11%. There are 3 isolated individuals and a total of 193 nuclear families, among which 112 have 2 children and 81 have 1 child.

From 112 families with 2 children, we infer 322 paternal and 535 maternal recombination events. Fig. 6 shows the resolution of the inferred recombination positions. 82% of the recombination events can be localized within an interval less than 100kb, and 53% within an interval less than 30kb.

Fig. 5(c) shows the averaged degree of freedom at each locus of a family after applying the Mendelian and zero-recombinant constraints. Based on the actual heterozygosity rate of the current dataset, there is expected to be 1.3347 degrees of freedom on a family of 2 children or 0.9982 degrees of freedom on a family of 1 child at each locus. By exploiting these constraints first, we have eliminated more than 95% of the phasing freedom of a family. A big family size will result in fewer degrees of freedom due to the increased number of constraints.

As shown in Fig. 5(a), the haplotype diversity varies for different locations of the chromosome. In the initial sampling (Fig. 5(b)), 23.41%(8.41%) of the most common haplotypes covers 90%(80%) of the total frequency. This indicates that most of the common haplotypes are recovered and sampled multiple times.

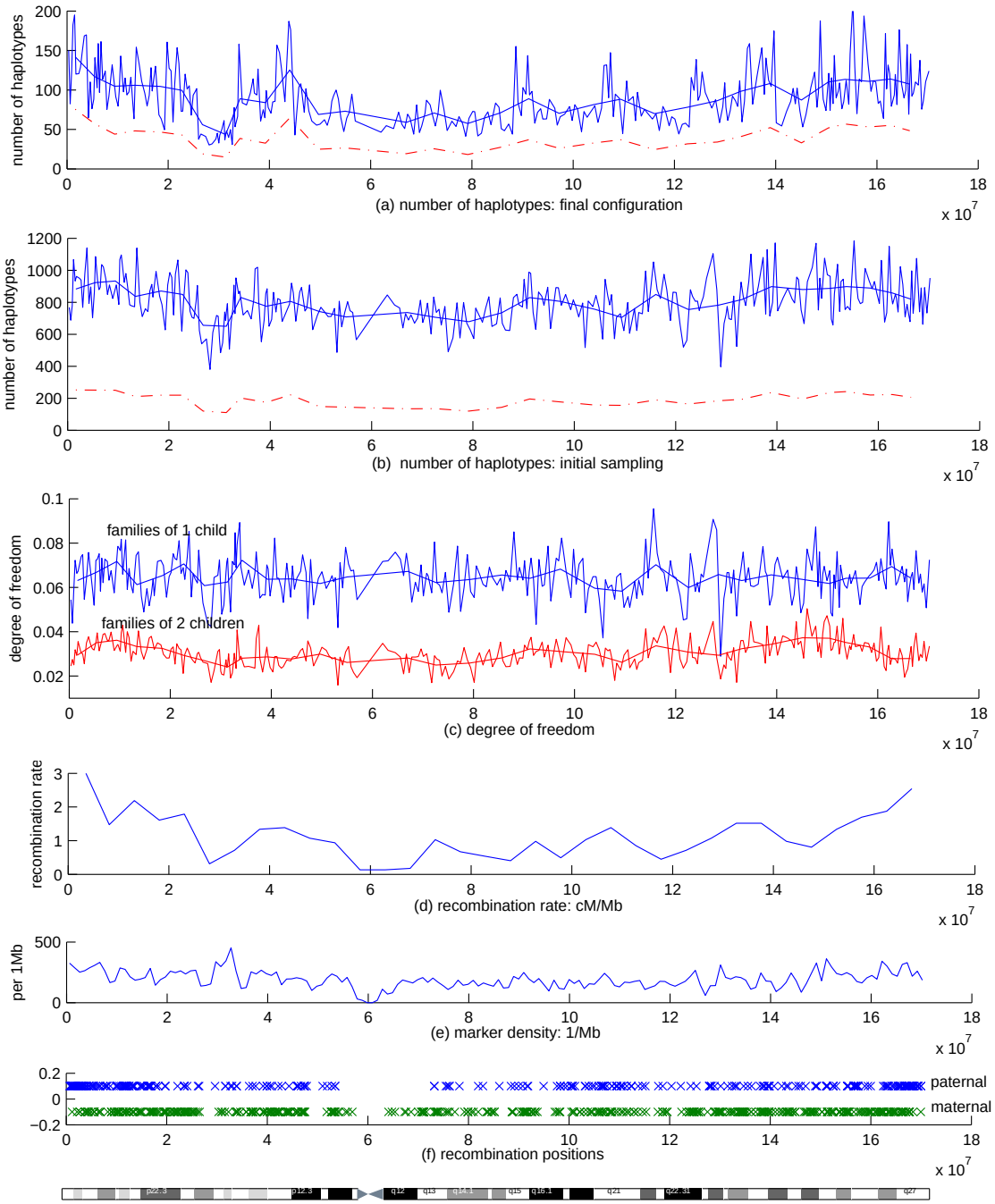


Fig. 5. X axis is the location as in base pairs on chromosome 6. (a)(b) Two charts show the number of haplotypes within each segment of 20 markers across chromosome 6, in the final configuration and the initial sampling respectively. Two solid lines are average numbers smoothed over 5 and 50 segments. The dashed line is the number of the most common haplotypes covering 90% of the total frequency. (c) Degree of freedom at each locus under Mendelian and zero-recombinant constraints. The lines are averaged values over all pedigrees of 1 child (upper) and 2 children (lower) respectively. Results are smoothed over 100 and 1000 markers. (d) Recombination rate in terms of centimorgan per million base pairs. (e)(f) The bottom two charts show the marker density, the paternal and maternal recombination positions over the whole chromosome. There are no markers around the centromere region.

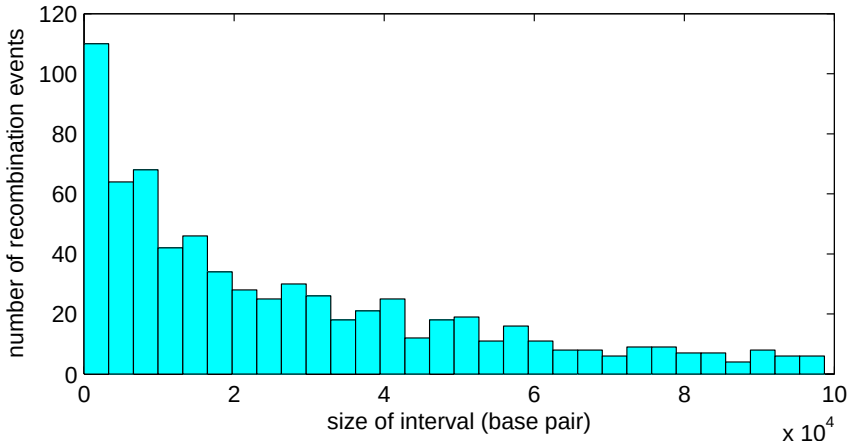


Fig. 6. The distribution of the length of ambiguous intervals of inferred recombination positions.

3.2. *Evaluation of Accuracy and Scalability*

We used the Cystic Fibrosis Transmembrane-Conductance Regulator (CFTR) Gene Data Set[8] for small scale testing and Simulated Rheumatoid Arthritis (RA) Data from Genetic Analysis Workshop (GAW) 15 for genome-wide testing of MML. We also compare the performance of our program with the conventional linkage analysis approach. Here, we use the statistical tool package Merlin.[2] Merlin can be used to perform haplotype inference on datasets of family and population mixed type. It first evaluates the inheritance vector in each family by exhaustive enumeration. Then it uses an EM approach to obtain the maximum likelihood configuration based on the population haplotype frequency. In order to deal with large numbers of markers, Merlin groups the markers by some pre-determined length and generates one single inheritance vector for each segment by assuming no recombination. However, if there does exist recombination in a segment, the program will fail. In Sec. 3.2.1, we compare MML and Merlin on small lengths of markers with different pedigree sizes to examine the efficiency by explicitly using Mendelian constraints rather than pure enumeration. On a genome-wide level (Sec. 3.2.2), our program can still successfully reveal the actual haplotype structure when Merlin is not applicable due to unresolved recombination.

3.2.1. *Influence of pedigree size, missing rate on performance*

We simulate pedigrees with no recombination to evaluate the performance of MML on zero-recombinant segments with different data settings. Pedigrees are generated with different sizes (4, 17, 29, 52) and missing rates (0.00, 0.05, 0.10, 0.15, 0.20) by using *SimPed*.[10] We pick from the CFTR data a subset of 29 distinct haplotypes of 19 markers spanning a region of 1.8Mb on chromosome 7q31. *SimPed* will assign founder haplotypes by sampling from the given set and transmit them onto other family members assuming no recombination. Each population has 500 families of a given parameter setting, and we average our results over 10 replicates of a population. The accuracy and running time comparison between MML and Merlin are shown in Fig. 7.

By explicitly exploiting the Mendelian constraints instead of enumerating all possible inheritance vectors, MML can achieve much greater time efficiency than Merlin on large pedigrees or on high missing rates (Fig. 7(b)). Both methods achieve better accuracy with large pedigrees (Fig. 7(a)) due to increased family constraints and population information. By adopting an approximate EM algorithm instead of traversing the whole search space, MML is of negligible accuracy difference to Merlin. This demonstrates the approximation approach to be a reasonable trade-off for efficiency. This is further confirmed on a large pedigree size of 52 (table in Fig. 7), where the performance of Merlin crashes with a high error rate (up to 30%) and exponentially longer running time due to too much freedom in resolving the inheritance vector. On the other hand, MML exhibits very robust consistency in both accuracy and efficiency.

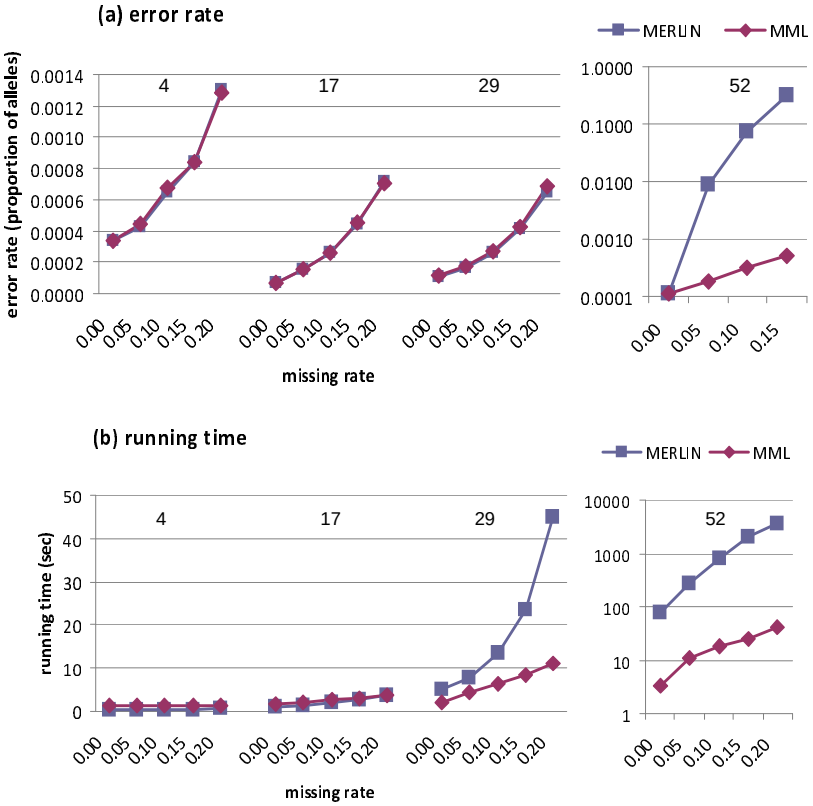
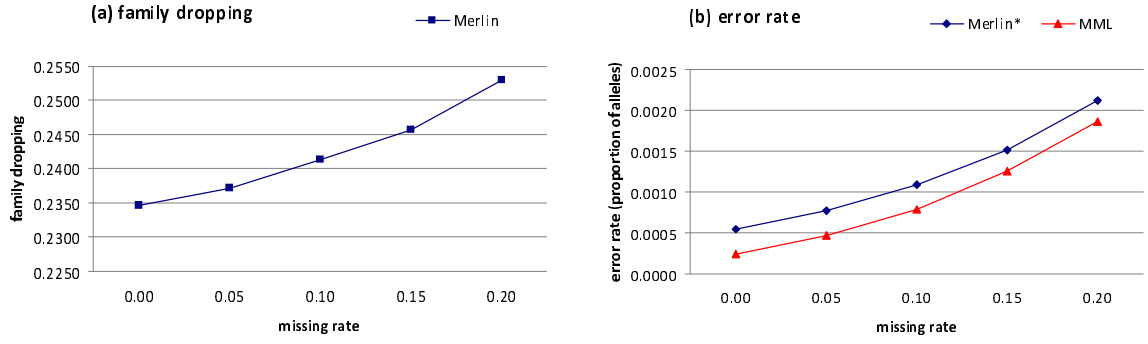


Fig. 7. Comparison of two methods on a dataset of 500 pedigrees. Performance is examined on 4 different pedigree sizes: 4, 17, 29, 52 and 5 different missing genotype rates: 0.00, 0.05, 0.10, 0.15, 0.20. Error rate is calculated by comparing the allele-by-allele difference between the inferred the haplotype and the correct haplotype.

3.2.2. *Genome-wide haplotype inference accuracy*

We tested MML and Merlin on chromosome 6 of the RA data which has 17820 SNPs with an average inter-marker spacing of 9586bp. The RA data consisted of 100 replicates, each with 1500 nuclear families (two parents and two offsprings). We used 500 out of the 1500 families and averaged our results over 10 replicates. We artificially set up to 20% genotype to missing to estimate the robustness of the methods against missing data.



(a) (b)

Fig. 8. Performance of MML and Merlin on chromosome 6 of RA data. Missing rates are 0.00, 0.05, 0.10, 0.15, 0.20. *The error rate of Merlin is based only on the families it successfully processes.

As shown in Fig. 8(b), MML can successfully reconstruct the haplotypes of each individual with an allele-by-allele error rate of 0.025% (for a missing rate of 0%) to an error rate of 0.19% (for a missing rate of 20%). Since Merlin assumes no recombination within each pre-defined segment, it fails on 25% of the 500 families (Fig. 8(a)). On the remaining families, it happens that all recombination events have ambiguous intervals riding across segment boundaries instead of contained completely in a single segment such that Merlin can still find a single inheritance vector for each segment. However, the overall accuracy of MML on all families is even better than the accuracy of Merlin on these retained families (Fig. 8(b)).

4. Conclusions

In order to reveal the human haplotypic variation on a genome-wide level, we need to overcome the computational difficulties complicated by huge numbers of markers, large pedigree and population size, and substantial numbers of missing genotypes. We applied our previous algorithm to find and compactly represent the subset of pedigree inheritance configurations that are consistent with the Mendelian law. We develop new techniques to resolve recombination positions in densely marked sequences, and a quick search strategy for detecting haplotype combinations of maximum likelihood in a population. All these techniques make it possible to handle large degrees of freedom in the pedigree inheritance patterns, the uncertainty of recombination positions, and the variety of possible haplotypes of a population. The combined exploitation of Mendelian constraints and local population structure makes the most of the current data designs to restore the underlying haplotype polymorphism. Experimental results on both real and simulated populations show that our method can successfully reconstruct the haplotypes with high accuracy and it is scalable in terms of both the pedigree (population) size and the missing genotype rate.

Acknowledgments

We would like to thank Dr. Fengyu Zhang and Matthew Hayes for helpful discussions. This research is supported by National Institutes of Health/National Library of Medicine grant LM008991, and in part by National Institutes of Health/National Center for Research Resources grant RR03655. Support for generation of the GAW15 simulated data was provided from NIH grants 5RO1-HL049609-14, 1R01-AG021917-01A1, the University of Minnesota, and the Minnesota Supercomputing Institute. We would also like to acknowledge GAW grant R01 GM031575.

References

1. The International HapMap Consortium, A second generation human haplotype map of over 3.1 million SNPs, *Nature* **449**:851–61, 2007.
2. Abecasis GR, Cherny SS, Cookson WO, Garden LR, Merlin-rapid analysis of dense genetic maps using sparse gene flow trees. *Nature Genetics* **30(1)**:97–101, 2002.
3. Abecasis GR, Wigginton JE, Handling marker-marker linkage disequilibrium pedigree analysis with clustered markers, *American Journal of Human Genetics* **77**:754–767, 2005.
4. Bader JS, The relative power of SNPs and haplotype as genetic markers for association tests, *Pharmacogenomics* **2(1)**:11–24, 2001.
5. Browning SR, Browning BL, Rapid and accurate haplotype phasing and missing data inference for whole genome association studies using localized haplotype clustering, *American Journal of Human Genetcis* **81**:1084-1097, 2007.
6. Browning BL, Browning SR, A unified approach to genotype imputation and haplotype phase inference for large data sets of trios and unrelated individuals, *American Journal of Human Genetcis* **84**:210-223, 2009.
7. Elston RC, Stewart J, A general model for the genetic analysis of pedigree data, *Human Heredity* **21**:523–542, 1971.
8. Kerem B, Rommens JM, Buchanan JA, Markiewicz D, Cox TK, Chakravarti A, Buchwald M, et al, Identification of the cystic fibrosis gene: genetic analysis, *Science* **245**:1073-1080, 1989.
9. Lander ES, Green P, Construction of multilocus genetic linkage maps in humans, *The Proceedings of the National Academy of Sciences* **84**:2363–2367, 1987.
10. Leal SM, Yan K, Müller-Myhsok B, SimPed: a simulation program to generate haplotype and genotype data for pedigree structures, *Human Heredity* **60**:119–122, 2005.

11. Li J, Jiang T, A survey on haplotyping algorithms for tightly linked markers, *Journal of Bioinformatics and Computational Biology* **6(1)**:241–259, 2008.
12. Li J, Jiang T, Computing the minimum recombinant haplotype configuration from incomplete genotype data on a pedigree by integer linear programming, *Journal of Computational Biology* **12**:719–739, 2005.
13. Li X, Li J, Comparisons of haplotype Inference from pedigree data and population data, *BMC Proceedings* **1**:S55, 2007.
14. Li X, Li J, An almost linear time algorithm for a general haplotype solution on tree pedigrees with no recombination and its extensions, Journal of Bioinformatics and Computational Biology **7(3)**:521-545, 2009.
15. Morris RW, Kaplan NL, On the advantage of haplotype analysis in the presence of multiple disease susceptibility alleles, *Genetic Epidemiology* **23**:221–233, 2002.
16. Scheet P, Stephens M, Fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase, *American Journal of Human Genetics* **78**:629–644, 2006.
17. Sobel E, Lange K, Descent graphs in pedigree analysis: applications to haplotyping, location scores, and marker-sharing statistics, *American Journal of Human Genetics* **58(6)**:1323–1337, 1996.
18. Xiao J, Liu L, Xia L, Jiang T, Fast elimination of redundant linear equations and reconstruction of recombination-free mendelian inheritance on a pedigree, *Proc. of 18th Annual ACM-SIAM Symoposium on Discrete Algorithms (SODA'07)*, 655–664, 2007.

SEQUENCE FEATURE VARIANT TYPE (SFVT) ANALYSIS OF THE HLA GENETIC ASSOCIATION IN JUVENILE IDIOPATHIC ARTHRITIS

GLENYS THOMSON[†]

Department of Integrative Biology, University of California
Berkeley, CA, USA

NISHANTH MARTHANDAN

Department of Pathology, University of Texas Southwestern Medical Center
Dallas, TX, USA

JILL A. HOLLENBACH[1], STEVEN J. MACK[1], HENRY A. ERLICH[1,2]

[1]Children's Hospital Oakland Research Institute, Oakland, CA, USA
[2]Roche Molecular Systems, Pleasanton, CA, USA

RICHARD M. SINGLE

Department of Mathematics and Statistics, University of Vermont
Burlington, VA, USA

MATTHEW J. WALLER[3], STEVEN G. E. MARSH[3,4]

[3]Anthony Nolan Research Institute, Royal Free Hospital, London, UK, and
[4]UK and UCL Cancer Institute, Royal Free Campus London, UK

PAULA A. GUIDRY[5], DAVID R. KARP[6], RICHARD H. SCHEUERMANN[3]

[5]Departments of Pathology and [6]Internal Medicine, University of Texas Southwestern Medical Center
Dallas, TX, USA

SUSAN D. THOMPSON, DAVID N. GLASS

Cincinnati Children's Hospital Medical Center
Cincinnati, OH, USA

WOLFGANG HELMBERG

Department of Blood Group Serology and Transfusion Medicine, Medical University of Graz
Graz, Austria

The immune response HLA class II DRB1 gene provides the major genetic contribution to Juvenile Idiopathic Arthritis (JIA), with a hierarchy of predisposing through intermediate to protective effects. With JIA, and the many other HLA associated diseases, it is difficult to identify the combinations of biologically relevant amino acid (AA) residues directly involved in disease due to the high level of HLA polymorphism, the pattern of AA variability, including varying degrees of linkage disequilibrium (LD), and the fact that most HLA variation occurs at functionally important sites. In a subset of JIA patients with the clinical phenotype oligoarticular-persistent (OP), we have applied a recently developed novel approach to genetic association analyses with genes/proteins sub-divided into biologically relevant smaller sequence features (SFs), and their "alleles" which are called variant types (VTs). With SFVT analysis, association tests are performed on variation at biologically relevant SFs based on structural (e.g., beta-strand 1) and functional (e.g., peptide binding site) features of the protein. We have extended the SFVT analysis pipeline to additionally include pairwise comparisons of DRB1 alleles within serogroup classes, our extension of the Salamon Unique Combinations algorithm, and LD patterns of AA variability to evaluate the SFVT results; all of which contributed additional complementary information. With JIA-OP, we identified a set of single AA SFs, and SFs in which they occur, particularly pockets of the peptide binding site, that account for the major disease risk attributable to HLA DRB1. These are (in numeric order): AAs 13 (pockets 4 and 6), 37 and 57 (both pocket 9), 67 (pocket 7), 74 (pocket 4), and 86 (pocket 1), and to a lesser extent 30 (pockets 6 and 7) and 71 (pockets 4, 5, and 7).

[†] corresponding author: glenys@berkeley.edu

1. Introduction

Genes in the major histocompatibility complex of humans, termed HLA, are known to influence susceptibility to over 300 diseases. These include: complex autoimmune and inflammatory diseases such as type 1 diabetes, rheumatoid arthritis, ankylosing spondilitis, psoriasis, multiple sclerosis, and narcolepsy; nasopharyngeal cancer, Hodgkin disease and other cancers; infectious diseases including malaria, tuberculosis, and AIDS, and diseases of unknown etiology [1]. HLA allele or haplotype and genotype associations with specific diseases are well established; the most complex pattern is seen with type 1 diabetes and HLA DRB1-DQB1 haplotypes, with a hierarchy from very predisposing, through intermediate ("neutral"), to very protective effects, with consistent patterns in associations seen across ethnic groups [2]. Specific amino acid (AA) residues, as well as combinations of AAs, have been implicated in type 1 diabetes risk, see e.g., refs. [2-4] and references contained therein, as well as many other HLA associated diseases. The strong association of the HLA DRB1 "shared epitope" set of AAs 70-74 and rheumatoid arthritis is well established; recently autoimmunity to citrullinated protein antigens has been shown to define a clinically and genetically distinct subset of rheumatoid arthritis that is specifically associated with the "shared epitope" alleles (reviewed in [5]). The recent development of a novel approach to genetic association analyses with genes/proteins sub-divided into biologically relevant smaller sequence features (SFs), and their variant types (VTs) [6], allows a systematic search focusing on the most likely actual causative genetic variants in HLA associated diseases.

HLA molecules are cell-surface proteins that present peptide fragments to T-cells to activate the recognition, and response to, foreign antigens. The classical class I (HLA-A, -B, and -C) and class II (HLA-DR, -DQ, and -DP) genes encode structurally homologous heterodimers. The processed self and foreign peptide fragments presented by the classical HLA molecules usually consist of 8–10 AAs (class I) or 12–20 AAs (class II), and are of either intracellular or extracellular origin respectively. An HLA molecule binds only to peptides conforming to certain structural requirements. A particular HLA allele is thus only able to present a subset of the available peptides to T-cells. The polymorphism of many of the HLA genes is extraordinary, with over 3,000 alleles at the class I and II genes identified to date (www.ebi.ac.uk/imgt/hla/) (and see e.g., ref. [7]), with much of the variation present at the protein level and occurring at functionally important sites. The class II HLA DRB1 gene, the subject of this study, has 623 alleles defined at the AA level; for any specific population or disease study only a fraction of these alleles will be observed. There are multiple lines of evidence for the role of balancing selection (at the allele and AA levels) in maintaining this most polymorphic set of genes in the human genome, including relatively even allele frequency distributions, see e.g., refs. [7, 8] and references therein. The causal argument presented is that individuals heterozygous for HLA genes can more effectively defend themselves from infection by successfully responding to a broader range of pathogens.

Peptide motifs important for binding to HLA molecules, including critical residues, have been defined by sequence analysis of naturally processed peptides eluted from HLA molecules, analysis with synthetic peptides, phage display libraries, and predictive inference of binding preference based on similarity of peptide-binding environments, see e.g., [9-11]. Specific AAs and combinations thereof have been identified as potentially causal in a number of HLA associated diseases. These usually rely on differential risk effects within serogroups of alleles, which involve a more restricted set of AAs compared to overall allele level comparisons, or identification of patterns from the sequence alignments of polymorphic sites at all alleles, again stratified by risk categories, e.g., DRB1 and the "shared epitope" set of AAs and rheumatoid arthritis discussed above. Determining the critical HLA AA residues involved in a specific disease can facilitate predictions about peptide epitopes, which are important for the design of novel vaccines and the understanding of autoimmunity.

Other analysis methods have been applied to type 1 diabetes and other diseases, and have successfully identified AAs important in disease risk heterogeneity, e.g., with type 1 diabetes, the Unique Combinations algorithm of Salamon et al. [3], and the Conditional Haplotype Method in Valdes et al. [4]. The aim with Sequence Feature Variant Type (SFVT) analysis [6] is to systematically perform association tests focusing on variation (VTs) at biologically relevant SFs, which are based on structural and functional features of the protein. The SFs include classical HLA allele level, and single AA, polymorphisms. The second round of SFVT analysis tests disease associations of temporary SFs (tSFs). These tSFs are constructed based on potentially informative combinations of

AAs that are identified from the first round of SFVT analysis as occurring frequently in SFs with the strongest associations with disease, including SFs composed of a single AA. With systemic sclerosis, specific AAs in pockets 4 and 7 of the peptide binding site explain much of the molecular determinant of disease risk [6].

We have applied SFVT analysis to HLA DRB1 variation and Juvenile Idiopathic Arthritis (Oligoarticular-Persistent) (JIA-OP) data [12]. We have extended SFVT analysis by creating an automated pipeline to include additional complementary and informative analyses (see Figure 1 and the Methods and Results Sections): including pairwise comparisons of HLA DRB1 alleles, and our extension of the Salamon Unique Combinations algorithm [3], to detect single AAs and combinations thereof that uniquely define different risk categories of alleles. AAs implicated from these analyses are now additionally combined with those identified by the first round of SFVT analysis in the construction of tSFs. The calculation of linkage disequilibrium (LD) patterns of AA variability in controls is another addition to the SFVT pipeline, guiding our understanding of effects that may be due to high correlation of AA variation. The final step in the analysis pipeline, which is not yet automated, is to apply a series of Conditional Haplotype Method analyses to differentiate AA effects which may be directly causative in disease versus those whose associations can be explained by LD with a causative AA or set of AAs.

2. Data and Methods

2.1. *Juvenile Idiopathic Arthritis (oligoarticular-persistent) (JIA-OP) HLA DRB1 data*

HLA DRB1 high resolution (4 digit AA level variation) data on 354 JIA-OP patients and 273 controls were analyzed (see Table 1 later). See Hollenbach et al. [12] for details on the data set, HLA typing, standard HLA association studies and results, and the general background on HLA associations with JIA and its subtypes.

2.2. *Sequence feature variant type (SFVTs) analysis*

The 181 SFs for the HLA DRB1 gene, as well as SFs for the other classical HLA genes, are defined in Karp *et al.* [6]. These range in size from the entire polypeptide sequence to single amino acids and involve:

(a) structural features: e.g., allele level variation (SF1), beta-strand 2 (SF13);
(b) biological function: e.g., peptide antigen binding site (SF127), T-cell peptide antigen binding pocket 6 (SF136);
(c) sequence alteration: all single AA positions with sequence variation, e.g., AA position 57 (SF90); and
(d) combinational structural_functional: e.g., beta-strand 2_peptide antigen binding pocket 7 (SF152).

Each HLA DRB1 allele is defined as a vector with these 181 SFs and their respective "allelic variations", which are referred to as variant types (VTs). The VTs for each SF defined in [6] use DRB1*0101 as the base; for example, for SF135 which is defined by AA positions 70 and 71, the VT1 (70Q_71R) is found in DRB1*0101, DRB1*0102, DRB1*0403, and a number of other alleles, while VT2 (70D_71E) is found in DRB1*0103, *0402, *1102 and a number of other alleles, etc. For the analysis of each specific SF, the VT frequencies are obtained by adding over the respective DRB1 alleles that carry that VT in the patient and control groups.

The DRB1 typing system used for the JIA-OP data set is based on identification of polymorphisms within exon 2, and hence we focus our attention on polymorphic AA variation within exon 2 (AAs 9-86). After initial analysis of the SFVT results, and combining information from the other analyses described below (Figure 1), so-called temporary SFs (tSFs), and their respective VTs, were defined based on sets of AAs with evidence of a role in differential disease risk.

2.3. *Chi-square heterogeneity testing*

A standard chi-square test was performed for heterogeneity testing of patient versus control DRB1 allele counts at the overall level, in pairwise comparisons, in the relative predispositional effects (RPE) analysis described below, and in SFVT analyses. Counts were combined in a "binned" category if the expected patient or control counts in the heterogeneity test were < 5. If the "binned" class had an expected control count < 5 it was not considered. The overall allelic effects in terms of disease risk are ranked by the standard Odds Ratio (OR) from most predisposing to most protective (see Table 1 later). The *p*-values are not corrected for multiple comparisons, since we are using these

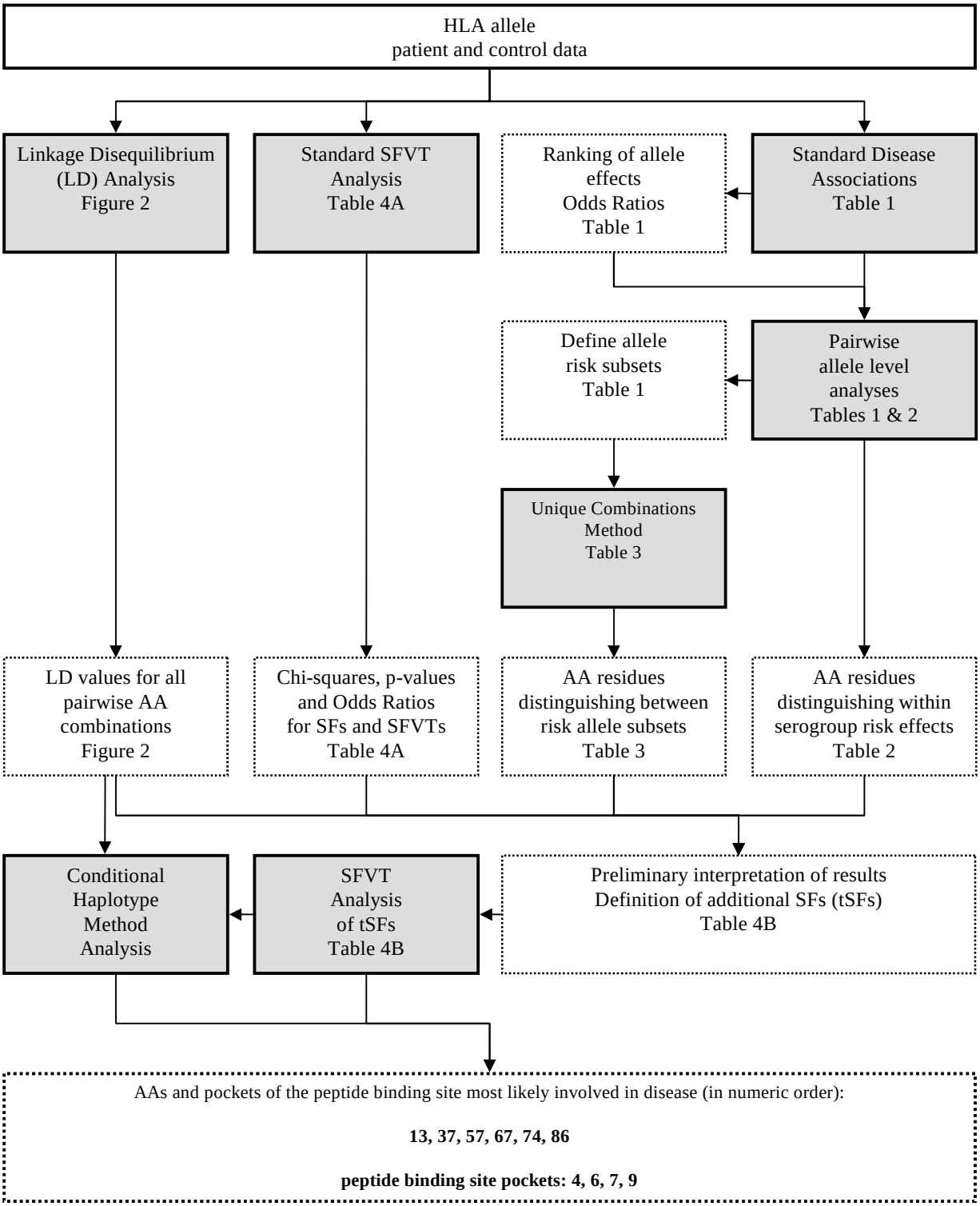


Figure 1. Analysis pipeline of SFVT analysis.

Methods are symbolised by solid boxes with grey shading, results are depicted as dotted boxes.

primarily for the purpose of relative rankings in this exploratory analysis; note that the AAs and SFs listed in this analysis as important in disease risk would still be very significant after correction if it were applied.

2.4. *Relative Predispositional Effects (RPE) analysis*

In Relative Predispositional Effects (RPE) analysis [13], alleles are sequentially removed from the standard chi-square heterogeneity test, starting with the strongest effects (based on the individual contribution of the allele to the overall chi-square), until there is no further significant heterogeneity detected either overall or for an individual allele. Note that this weights targeting and removal of the more common alleles; there is no intrinsic bias except for this factor. When there are relatively similar predisposing and protective effects—in terms of the strengths of their effects—in a particular round of analysis, these were both removed from the next round.

2.5. *Linkage disequilibrium (LD)*

The correlation measure W_n, also known as Cramer's V statistic, is used to quantify the overall (global) strength of LD among SFs; each SF is considered as a separate locus ("gene"), with the VTs the "alleles" thereof as in standard LD estimation [14]. For two bi-allelic loci, W_n is equivalent to the correlation coefficient ($r = \sqrt{D_{11}^2 / p_1 p_2 q_1 q_2}$), where p_1 and p_2 are the allele frequencies at the first locus ($p_1 + p_2 = 1$), and similarly q_1 and q_2 for the second locus, and $D_{11} = x_{11} - p_1 q_1$, where x_{11} is the frequency of the haplotype composed of these two alleles ($D_{11} = D_{12} = D_{21} = D_{22}$). The multi-allelic extension, W_n, is defined as $W_n = \frac{\sqrt{\sum_{i=1}^{I}\sum_{j=1}^{J} D_{ij}^2 / p_i q_j}}{\sqrt{\min(I-1, J-1)}}$, where p_i and q_j are the allele frequencies at the first and second loci with I and J alleles respectively, and the individual LD coefficients are $D_{ij} = x_{ij} - p_i q_j$. Both r and W_n have a range from 0 to 1. When all $D_{ij} = 0$, $r = 0$ for bi-allelic loci, and $W_n = 0$ for multi-allelic loci, and the alleles at the two loci are randomly associated. In contract, for high LD the alleles at the two loci are very highly correlated, with complete correlation when r or $W_n = 1$.

2.6. *Salamon's Unique Combinations algorithm*

The Unique Combinations algorithm developed in [3] efficiently identifies combinations of AAs that distinguish a sequence or set of sequences from a set of other sequences. See Section 3.4 in Results for more details.

3. Results

3.1. *Summary*

Given the complexity of the integration of results and incorporation of information from within and between various levels of analyses, e.g., pairwise allele level comparisons, Unique Combinations analyses, LD patterns, and SFVT analysis (Figure 1), we summarize the findings now to aid navigation through the series of results detailed below. The AAs we identify as major or important in JIA-OP disease risk, always indicated in bold below are (listed in numeric order): **AAs 13** (pockets 4 and 6), **37** and **57** (both in pocket 9), **67** (pocket 7), **74** (pocket 4), **86** (pocket 1), and those potentially involved in disease risk are (underlined): <u>30</u> (pockets 6 and 7), and <u>71</u> (pockets 4, 5, and 7).

3.2. *HLA DRB1 allele level analyses*

A total of 38 DRB1 alleles were observed: 17 were included in the "binned" category, leaving 21 frequent alleles (Table 1). (Note that for the SFVT analysis, the rare DRB1 alleles are always included.) As previously described [12] there is significant heterogeneity in allele counts between patients and controls (overall test: $p < 1.1E\text{-}27$). Two predisposing alleles: DRB1*0801 and *1104, and three protective alleles: DRB1*1501, *0701, and *0401, show very strong individual effects (based on their *p*-values), with weaker effects of DRB1*1301, *1103, *0404, and *0103. Note that for the rarer alleles, e.g., DRB1*1103 which has the highest OR, the 95% OR confidence interval (CI) spans a large range; hence conclusions from analyses that use this information are subject to this uncertainty.

The first column of Table 1 lists the alleles (labeled in categories 1-3) which were sequentially removed in successive rounds in the Relative Predispositional Effects (RPE) analysis until no significant differential effects were seen. Note that (as mentioned in Section 2.4) the higher frequency alleles are targeted with this analysis. In the second column, the more common alleles are divided into a set (A) of three differential risk categories containing high frequency alleles: I (predisposing), II (neutral), and III (protective). The boundaries, and inclusion of alleles in each category, were predicated on evidence of disease risk heterogeneity between categories, and homogeneity within categories. In the third column, the list of alleles (set B) in each risk category is expanded to include some rarer alleles, but with the boundaries still delineated by more common alleles; inclusion in each respective risk category is now indicated by Ix, IIx, and IIIx. The rare allele DRB1*0403 is not included as it cannot be classified as predisposing versus neutral, similarly the four rare alleles between sets IIx and IIIx (the neutral/protective boundary). These sets A and B of differential risk categories are used below in the Unique Combinations analyses.

Table 1. JIA-OP HLA DRB1 allele data ranked by Odds Ratio (OR)

RPE[a]	A[b]	B[c]	DRB1	JIA-OP	Controls	Chi-sq.	p-value[d]	OR	CI[e]	CI[e]
		Ix	*1103	12	1	6.80	0.01	9.40	1.22	72.49
1	I	Ix	*0801	102	13	48.61	3.1E-12	6.90	3.83	12.43
2	I	Ix	*1104	57	11	20.71	5.3E-06	4.26	2.21	8.20
			*0403	9	3	1.68	0.20	2.33	0.63	8.65
3	II	IIx	*1301	90	38	9.99	0.002	1.95	1.31	2.90
		IIx	*0102	9	5	0.35	0.55	1.39	0.46	4.18
	II	IIx	*1101	60	36	1.42	0.23	1.31	0.85	2.02
		IIx	*0901	9	6	0.08	0.78	1.16	0.41	3.28
	II	IIx	*0101	74	50	0.52	0.47	1.16	0.79	1.69
	II	IIx	*0301	89	61	0.50	0.48	1.14	0.81	1.62
		IIx	*1201	10	8	0.006	0.94	0.96	0.38	2.46
	II	IIx	*1302	28	23	0.05	0.82	0.94	0.53	1.64
			*1303	10	9	0.11	0.74	0.86	0.34	2.12
			binned[f]	27	27	0.92	0.34	0.76	0.44	1.31
			*1601	6	8	1.05	0.30	0.58	0.20	1.67
			*1401	11	18	4.05	0.04	0.46	0.22	0.99
			*1502	5	10	3.26	0.07	0.38	0.13	1.12
	III	IIIx	*0404	7	16	6.34	0.01	0.33	0.14	0.81
1	III	IIIx	*1501	38	80	28.24	1.1E-07	0.33	0.22	0.49
1	III	IIIx	*0701	30	65	23.92	1.0E-06	0.33	0.21	0.51
2	III	IIIx	*0401	21	47	18.10	2.1E-05	0.33	0.19	0.55
3		IIIx	*0103	4	11	5.42	0.02	0.28	0.09	0.87
			TOTAL	708	546	182.1	1.1E-27			

[a] Numbers denote the order of removal due to largest effect(s) in the RPE analysis
[b] Set A: The common alleles are divided into mutually exclusive, and significantly different, predisposing (I), intermediate (II), and protective (III) categories for use in the Unique Combinations comparisons
[c] Set B: The sets I, II, and III above are expanded to include rare alleles, while excluding those alleles which do not clearly fall into one of the 3 risk categories
[d] The individual *p*-values are biased (conservative with respect to finding significant effects) as the assumption of a 1 df chi-square is incorrect; the *p*-values can be used however for a relative ranking of the allelic effects
[e] The upper and lower 95% confidence intervals (CIs) for the Odds Ratio (OR) are given
[f] The binned category consists of all alleles with an expected value < 5 under the chi-square test of heterogeneity of patient and control allele counts

3.3. *HLA DRB1 allele level pairwise within serogroup analyses*

HLA nomenclature (except for DP) is such that alleles sharing the same 2 first digits generally belong within the same serogroup. Variation in the 2 last digits indicates AA differences within the serogroup. For example, DRB1*0101, *0102, and *0103 belong to the serogroup denoted *01XX. Alleles within the same serogroup are more closely related at the AA level, hence significant differences in risk within serogroups, and between specific pairs of alleles, may identify specific AAs, or a few AAs, involved in disease. Pairwise comparisons within serogroups of alleles with sufficient sample size—DRB1*01XX, *04XX, *11XX, and *13XX—were performed; this was followed by manual inspection of their respective sequences for comparisons where significant risk heterogeneity was detected. Significant results (ordered by *p*-value) are given in Table 2. The strong evidence for the role

Table 2. JIA-OP HLA DRB1 allele pairwise comparisons

Alleles compared[a]	p-value[b]	AAs[c,d]
*0403 vs *0401 + *0404	0.002	**74**
*1104 vs *1103	0.003	**86**
*0403 vs *0401	0.004	71, **74**, or **86**
*0101 vs *0103	0.02	**67**, 70, or 71
*1103 vs *1101	0.04	71 or **86**
*1301 vs *1302	0.05	**86**

[a] Within serogroup allele comaprisons
[b] Uncorrected p-value from the chi-square test of heterogeneity
[c] Amino acid residues that uniquely define these specific alleles
[d] Amino acids indicated in **bold** are those identified as playing a major or important role in disease risk, those underlined as potentially having an effect, albeit weaker

of **AA 86** in differential disease risk is of particular interest, since with SFVT analysis this AA shows *no* significant effect (see Section 3.6 and Table 4A below). There is also evidence for a direct role of **AA 74**.

3.4. *Unique Combinations comparisons*

In the original Unique Combinations algorithm of Salamon et al. [3], two categories of sequences are defined by the user: those in the "check" category are compared against those in the "group" category in order to identify combinations of sites that are unique between these two sets of sequences. However, when there are two or more sequences in the "check" category, sites that are polymorphic between these "check" sequences are excluded from consideration. We have extended the algorithm to allow inclusion of all sites that are polymorphic in the "check" category, thus expanding the utility of the method. Also, this means that the "group" and "check" categories are now interchangeable, whereas before there was an asymmetry.

This extension of the Unique Combinations algorithm provides an ordered list of a minimal number of polymorphic positions, which as a haplotype (combination of AAs on a chromosome) can differentiate between any set of sequences of alleles in the "check" versus "group" categories. Deriving the vectors of AAs that correspond to the resulting minimal unique combination generates unique sequences that either belong to the "check" category or the "group" category.

Using the subdivisions of common DRB1 alleles into the three categories defined above as sets A and B (Table 1)—I and Ix (predisposing), II and IIx (intermediate), and III and IIIx (protective)—we performed various Unique Combinations comparisons of each risk category versus the other two risk categories, e.g., I versus II + III (Table 3). The AAs identified as important in the Unique Combinations analyses are **AA 86** (as in the pairwise allele within serogroup analyses in Section 3.3 above) combined with **AAs 13** and **37**, or **13** and **67**.

Table 3: JIA-OP HLA DRB1 Unique Combinations (UC) analyses

group[a]	I	Ix	III	IIIx	II	IIx	
check[a]	II + III	IIx + IIIx	I + II	Ix+IIx	I + III	Ix+IIIx	AAs[b]
			X				**13**
				X			**13, 67**
			X	X			**37, 67**
	X	X			X		**13, 37, 86**
	X	X			X	X	**13, 67, 86**

[a] The sets of predisposing (I and Ix), intermediate (II and IIx), and protective (III and IIIx) alleles are defined in Table 1, the group and check categories define the two groups of alleles compared

[b] Only AAs in exon 2 that are consistently seen in all comparisons are listed in this column, other AAs which appear in some comparisons are: 47, 57, 70, 71, 74

3.5. *HLA DRB1 amino acid LD patterns*

The AA LD values in the control data for exon 2 of the HLA DRB1 variation (AAs 9-86) are given in Figure 2. This information is used in evaluation of the SFVT data below. The LD values show a complex pattern with (using examples from the six AAs we have identified as most strongly implicated in disease risk): (1) "blocks" of AAs where adjacent sites all have high LD with each other (**13** and 9-12); (2) individual AAs, each with high and moderate levels of LD with quite a few other AAs (**13** and **37**), and similarly but with lower levels of LD (**57** and **74**); and (3) individual AAs with very low levels of LD with most or all other AAs (**67** and **86**).

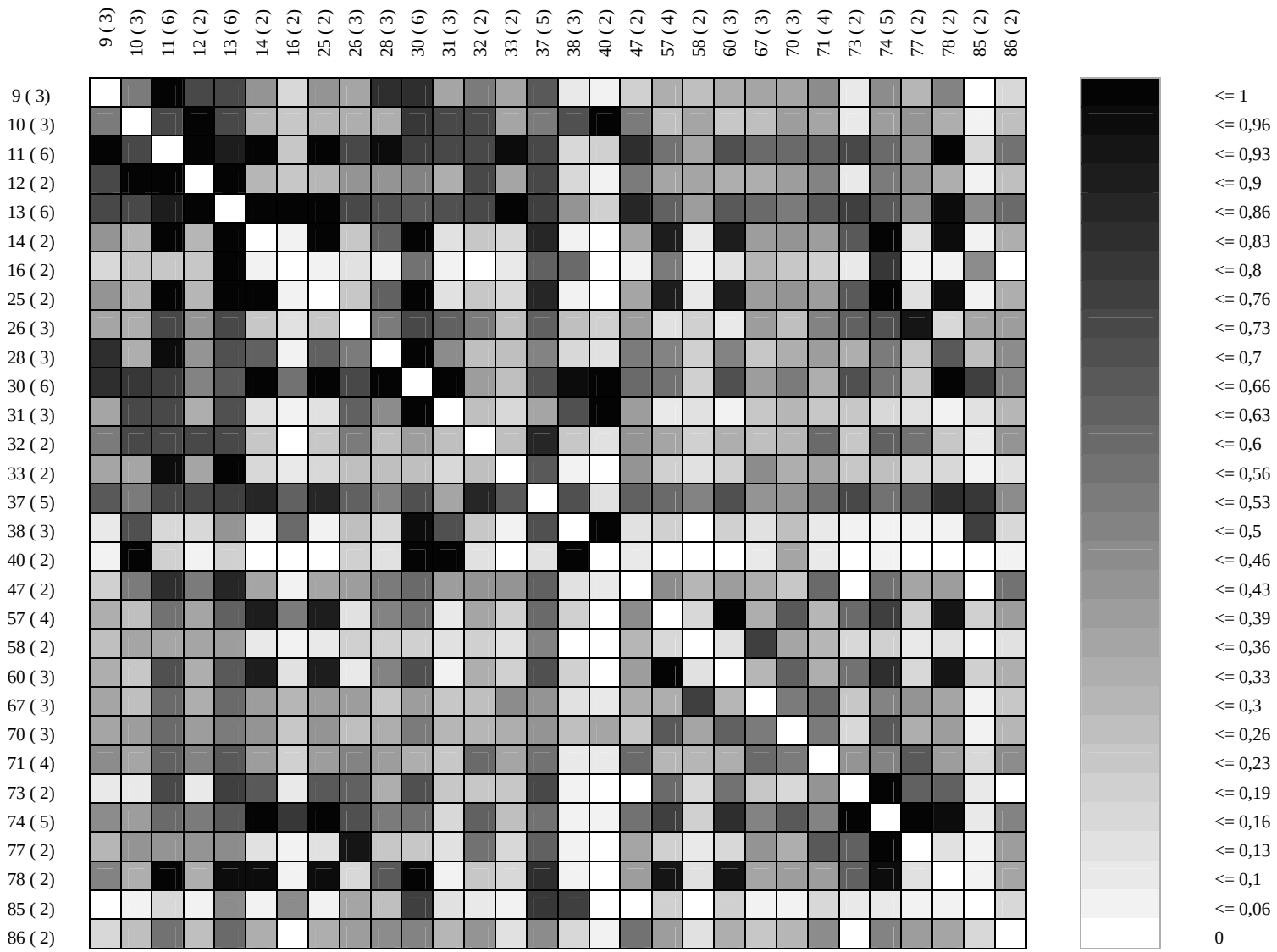


Figure 2. Linkage disequilibrium (LD) plot of polymorphic amino acids 9 - 86 of HLA DRB1 [a]

[a] The data are from the control samples; the axes give the amino acid positions and the number of residues (in parentheses) at each position

3.6. *SFVT analyses (Table 4, part A)*

The SFVT data in Table 4A are ranked by the *p*-value of the overall chi-square heterogeneity patient versus control analysis of the VTs for each SF listed. Note that any attempt to draw conclusions from minor differences in *p*-values of this magnitude is an over-interpretation of the data. Also listed are the maximum (max) and minimum (min) ORs seen for *individual* VTs for each SF, for example, for SF1 (allele) (rank 5 in Table 4A), these are 9.40 and 0.28 (see Table 1). These values are used to determine the ability of a SF to differentiate between risk categories of VTs, both with consideration of the max and min ORs and the range of the ORs for a particular SF. However, keep in mind, as mentioned above, that the highest OR of 9.40 for allele DRB1*1103 is based on a very rare allele, the next highest allele level OR is 6.90 for DRB1*0801 (see Table 1).

AA 13, pockets 4 and 6: The single **AA 13** (SF57) is rank 1 in the SFVT analysis (Table 4A), with a reasonable range in OR values (4.91 to 0.33). From Table 5, we see how **AA 13** by itself partitions the disease risk (although certainly not perfectly): the residues G and S are only seen in the predisposing and neutral allele level disease risk categories, while the other 4 residues: F, R, H and Y are only seen in the neutral and protective categories. **AA13** is the major contributor to the **pocket 6** and **pocket 4** associations (ranks 2 and 3). AA 30 (**pocket 6**) may play some role in disease risk, but the effect may be explained by LD.

The effects of the single AAs 9, 10, 11, 12 and 16, which occur in the top 20 ranked SFs in Table 4A, can be explained by LD with **AA 13** (Figure 2). These AAs are indicated in *italics* in Table 4A, and are not discussed individually below. **AA 13** is chosen over these AAs since it individually has a stronger effect, based on *p*-values and OR values and range, it occurs in more top ranked SFs than do any of these AAs, and it was identified individually over these other AAs, in combination with **AAs 37** and **86**, or **67** and **86**, in the Unique Combinations analyses.

AA 67, pocket 7: AA 67 (SF98, rank 14) has the second highest rank of the single AAs, and is the major contributor to the SFs ranked above it and below SF1 (allele) (excluding the 3 single AA SFs AAs *10*, *11*, and *12*, see above), and was identified in the Unique Combinations analyses. While AA 71 (SF102, rank 22) contributes an additional effect to **AA 67** by itself, due to greatly increasing the OR max value of **pocket 7** (rank 6), this in fact is a minor effect (in this data set) reflecting unique identification of the very *rare* predisposing DRB1*1103 allele.

AAs 13, 67, 74, and 86, and pockets 6, 4, and 7: AAs 13 and **67** together distinguish all the *significant* effects of the alleles listed in Table 1, with two exceptions which are covered by **AAs 74** and **86**. **AA 74** (the third ranked highest single AA effect (SF104, rank 16)) was identified in the allele level pairwise comparisons (Table 2) as necessary to distinguish between the *rare* neutral/predisposing DRB1*0403 allele and the protective *0401 and *0404 alleles ($p < 0.002$). Note that this effect of **AA 74** was not picked up in the Unique Combinations analyses, since our inability to definitively place DRB1*0403 in either the predisposing or neutral categories precluded its consideration in the Unique Combinations analyses. **AA 86** was identified in the allele level pairwise comparisons as necessary to distinguish between the predisposing DRB1*1104 and neutral *1101 effects ($p < 0.003$) (Table 2); it was also picked up in the Unique Combinations analyses. However, in the SFVT analysis (Table 4A), note that the individual effect of **AA 86** is *not* significant, nor is it identified as potentially involved in disease from its presence in other SFs: it is the 8th *lowest* ranked SF (SF110, rank 42), and pocket 1 (SF132) in which it occurs ranks two below this. However, as indicated in Table 2, **AA 86** is *necessary* to explain significant disease risk effects. The four **AAs 13, 67, 74,** and **86** uniquely define all the alleles listed in Table 1 (see Table 5), not considering the very rare alleles in the binned category in Table 1.

AAs 37 and 57 (pocket 9): AAs 37 (SF74, rank 17) and **57** (SF90, rank 19) are the next to enter into our consideration. The Unique Combinations results (Table 2) implicate **13**, **37**, and **86** as a potential AA combination that can explain disease risk, along with **13**, **67**, and **86** as an alternate combination. Further, to jump ahead, the AA combination **13** and **37** together have rank 1 in the analysis of tSFs discussed below (Table 4B). The combination of **AAs 13, 37, 74**, and **86** also explains all known disease risk at the allele level, except the marginally significant allele pairwise comparison of DRB1*0101 (neutral) to the *rare* *0103 (protective) ($p < 0.02$); **AA 67** distinguishes this risk, as does AA 71; as above AA 71 also further distinguishes DRB1*1103 and *1104, which in this data set are not significantly different in their effects.

Table 4. SFVT results

Rank	SF #	Description	amino acid[a]	*p*-value	max OR	min OR
A: JIA-OP HLA DRB1 SFVT analysis with SFs ranked by overall *p*-values						
1	57	**position 13**	**13**	1.8E-28	4.91	0.33
2	136	**pocket 6**	*11*, **13**, 30	3.9E-28	7.07	0.31
3	134	**pocket 4**	**13**, 26, 28, 70, 71, **74**, 78	5.7E-28	6.84	0.28
4	151	*beta1_pep ant-TCR*	*11*, **13**	9.4E-28	4.89	0.33
5	1	**allele**[b]		1.1E-27	9.40	0.28
6	137	**pocket 7**	28, 30, 47, 61, **67**, 71	8.9E-27	9.40	0.28
7	142	pep ant position 12	*9*, 56, **57**, 60, 61, **67**	4.2E-26	7.14	0.31
8	55	*position 11*	*11*	8.8E-25	3.15	0.33
9	130	pep ant & T cell rec	60, **67**, 70, 71, 77, 78, 85	7.0E-24	9.40	0.32
10	56	*position 12*	*12*	1.1E-22	3.15	0.32
11	54	*position 10*	*10*	1.9E-22	3.14	0.32
12	162	alpha2_pocket 7	**67**, 71	2.9E-21	9.40	0.33
13	19	alpha-helix 1	52..62	3.5E-18	3.92	0.44
14	98	**position 67**	**67**	3.0E-17	3.39	0.54
15	138	**pocket 9**	9, **37, 57**	3.6E-16	3.92	0.33
16	104	**position 74**	**74**	3.8E-16	6.84	0.33
17	74	**position 37**	**37**	3.7E-13	1.80	0.34
18	59	*position 16*	*16*	5.4E-13	4.91	0.20
19	90	**position 57**	**57**	5.5E-13	3.92	0.44
20	53	*position 9*	*9*	2.3E-11	2.30	0.42
21	135	pocket 5	70, 71	1.4E-10	1.79	0.33
22	102	position 71	71	1.2E-09	1.48	0.33
23	101	position 70	70	4.5E-09	2.03	0.54
24	71	position 33	33	3.1E-07	2.62	0.38
25	58	*position 14*	*14*	3.6E-07	3.05	0.33
26	63	*position 25*	*25*	3.6E-07	3.05	0.33
27	91	position 58	58	1.0E-06	2.35	0.43
28	24	position 78	78	3.3E-06	2.56	0.39
29	93	position 60	60	3.5E-06	2.36	0.44
30	68	position 30	30	2.3E-05	1.55	0.33
31	152	beta2_pocket 7	28, 30	4.3E-05	1.55	0.33
32	155	beta2_pocket 4	26, 28	4.4E-05	1.31	0.34
33	141	pep ant position 4	77, 78, 81, 82, 85	7.4E-05	1.37	0.39
34	153	beta2_pep ant bind	26, 28, 30	0.0001	1.32	0.33
35	13	beta-strand 2	23..32	0.0002	1.29	0.33
36	66	position 28	28	0.004	1.54	0.63
37	22	position 73	73	0.007	1.47	0.68
38	81	position 47	47	0.03	1.28	0.78
39	178	beta1_CD4 bind	41..56	0.03	1.28	0.78
40	70	position 32	32	0.07	1.25	0.80
41	154	beta2_alpha chain	29, 31, 32	0.13	1.27	0.80
42	110	**position 86**	**86**	0.34	1.12	0.90
43	106	position 77	77	0.41	1.16	0.86
44	132	pocket 1	82, 85, **86**, (89, 90)	0.58	1.14	0.89
45	173	alpha4_pocket1	82, 85, **86**	0.63	1.13	0.90
46	64	position 26	26	0.67	1.16	0.93
47	109	position 85	85	0.74	1.13	0.88
48	69	position 31	31	0.86	1.03	0.97
49	75	position 38	38	0.91	1.04	0.96

Rank	SF #	Description	amino acid[a]	p-value	max OR	min OR
B: JIA-OP HLA DRB1 SFVT analysis of tSFs ranked by overall *p*-values						
1	t201		**13**, **37**	5.6E-30	7.04	0.37
2	t205		**13**, **37**, **74**, **86**	2.7E-29	6.69	0.37
3	t211		**13**, 30, **37**, **74**, **86**	3.4E-29	6.69	0.37
4	t210		**13**, 30, **67**, **74**, **86**	1.2E-28	6.47	0.28
5	t206		**13**, **67**, **74**, **86**	1.2E-28	6.47	0.28
6	t224		**13**, **37**, **67**	1.3E-28	6.84	0.28
7	57	position **13**	**13**	1.8E-28	4.91	0.33
8	t225		**37**, **67**	2.2E-28	3.90	0.31
9	t203		**13**, **67**	2.3E-28	6.84	0.28
10	t204		**13**, **67**, **86**	2.5E-28	6.69	0.28
11	t218		**13**, **37**, 57, **67**, **74**, **86**	3.6E-28	6.90	0.28
12	t207		13, **37**, **67**, **74**, **86**	3.3E-28	6.47	0.28
13	136	pocket 6	11, **13**, 30	3.9E-28	7.07	0.31
14	t212		**13**, 30, **37**, **67**, **74**, **86**	4.4E-28	6.47	0.28
15	t214		**13**, 57, **67**, **74**, **86**	4.5E-28	6.90	0.28
16	151	beta1_pep ant-	*11*, **13**	9.4E-28	4.89	0.33
17	t216		**13**, 30, **37**, 57, **67**, 71, **74**, **86**	1.1E-27	9.40	0.28
18	t215		**13**, 57, **67**, 71, **74**, **86**	1.1E-27	9.40	0.28
19	1	allele		1.1E-27	9.40	0.28
20	137	pocket 7	28, 30, 47, 61, **67**, 71	8.9E-27	9.40	0.28

[a] Amino acids indicated in **bold** are those identified as playing a major or important role in disease risk, those underlined as potentially having an effect, albeit weaker, and for those in *italics* their effect may be explained by LD with **AA 13**.

[b] SF127 (peptide antigen binding site) AAs: 9, 11, 13, 26, 28, 30, 37, 47, 56, 57, 60, 61, 67, 70, 71, 74, 77, 78, 81, 82, 85, 86, 89, 90 (not listed), has identical VT counts as SF1 (allele)

Other AAs: Single AAs and other SFs listed below rank 20 in Table 4A either do not themselves have a wide range of OR values, and/or their effects may be explained by LD with other AAs or SFs, or they do not have a significant overall effect. However, we note that AAs 9, 10, 11, 12, 26, 28, 47, 56, 58, 60, and 85 also show up frequently in the top SFVT results, including comparisons based on *p*-values for individual VTs of each SF, and these should be considered in additional analyses.

3.7. *SFVT Analysis of Temporary SFs (tSFs) (Table 4, part B)*

The SFVT data in Table 4B are the most relevant data from analyses of tSFs defined by specific potentially informative combinations of the **AAs 13, 37, 57, 67, 74, 86,** 30, and 71 (based on the analyses above). Again, these are ordered by *p*-value, but note that the range of *p*-values in Table 4B shows only minor differences, and the effects we will concentrate on hence focus more on the max and min OR and the range of OR values. The SFs from Table 4A which include the span of *p*-values of the tSFs are also included in Table 4B for comparison.

Table 5: JIA-OP HLA DRB1 Amino Acid Residue Variation

	AA position	13	67	74	86	37	57	30	71
	# alleles	6	3	4	2	5	5	5	4
OR[a]	DRB1 alleles								
9.4	DRB1*1103	S	F	A	V	Y	D	Y	E
6.9	DRB1*0801	G	F	L	G	Y	S	Y	R
4.26	DRB1*1104	S	F	A	V	Y	D	Y	R
2.33	DRB1*0403	H	L	E	V	Y	D	Y	R
1.95	DRB1*1301	S	I	A	V	N	D	Y	E
1.39	DRB1*0102	F	L	A	V	S	D	C	R
1.31	DRB1*1101	S	F	A	G	Y	D	Y	R
1.16	DRB1*0901	F	F	E	G	N	V	G	R
1.16	DRB1*0101	F	L	A	G	S	D	C	R
1.14	DRB1*0301	S	L	R	V	N	D	Y	K
0.96	DRB1*1201	G	I	A	V	L	V	H	R
0.94	DRB1*1302	S	I	A	G	N	D	Y	E
0.86	DRB1*1303	S	I	A	G	Y	S	Y	K
0.58	DRB1*1601	R	F	A	G	S	D	Y	R
0.46	DRB1*1401	S	L	E	V	F	A	Y	R
0.38	DRB1*1502	R	I	A	G	S	D	Y	A
0.33	DRB1*0404	H	L	A	V	Y	D	Y	R
0.33	DRB1*1501	R	I	A	V	S	D	Y	A
0.33	DRB1*0701	Y	I	Q	G	F	V	L	R
0.33	DRB1*0401	H	L	A	G	Y	D	Y	K
0.28	DRB1*0103	F	I	A	G	S	D	C	E

[a] Odds Ratio (OR), see Table 1

The pair of **AAs 13** and **37** (SFt201, rank 1 in Table 4B) captures the JIA-OP disease risk with 11 VTs, including a good range of OR values, as does the pair **13 and 67** (SFt203, rank 9) with 13 VTs. Both increase the max OR, but barely change the min OR, compared to **AA 13** by itself (SF57, rank 7, 6 VTs). The addition of **AAs 74 and 86** to both these combinations (SFt205, rank 2: and SFt206, rank 5) adds no further discrimination based on the SFVT analysis, but as noted above are required to account for all disease risk heterogeneity.

Only addition of AA 71 gives the full range of ORs seen with DRB1 allele level variation (SF1, rank 19) (see SFt216, rank 17 and SFt215, rank 18); as noted above, AA 71 distinguishes the rare highest predisposing risk DRB1*1103 allele from the predisposing (common) DRB1*1104 allele.

4. Discussion

While many HLA associations at the allelic level are well documented and consistently found, identification of the AAs directly involved in disease risk is difficult, due to the nature of HLA polymorphism (discussed in the Introduction). SFVT analysis was designed to facilitate the analysis of HLA disease associations by focusing on structurally and functionally important subsets of HLA variation [6]. SFVT analyses can also be informative in study of other highly polymorphic genes such as HIV gp160 and influenza hemagglutinin.

Our SFVT analysis of HLA DRB1 and JIA-OP, with an expanded pipeline (Figure 1) to integrate results from within serogroup comparisons of alleles, Unique Combinations analysis of alleles in distinct risk categories, and study of LD patterns of individual AAs, identified the following HLA DRB1 AAs as major or important JIA-OP risk factors

(listed in numeric order): **13**, **37**, **57, 67**, **74**, and **86**; with lesser effects of 30 and 71. These AAs are all contained within the peptide binding site, and explain the increased disease risk associated with **pockets 4, 6, 7, and 9** of the peptide binding site. When there is high LD between AAs, it is difficult to distinguish the causative agent, e.g., **AA 13** is in high LD with a number of other AAs, and further study of these AAs is required. Note however, that when the LD is very high it may be impossible to distinguish between highly correlated AA sites; studies in other ethnic groups where the LD pattern may be different are then useful. The AAs we have focused on certainly do not represent all of the HLA DRB1 contribution to JIA-OP. Note that defining a minimal set of AAs that describe all AA allele level variation does not mean that additional AAs do not have effects on disease risk.

Further study of the various combinations of the AAs we have identified and their contributions to JIA-OP risk is also required. Preliminary analyses, using the Conditional Haplotype Method [4, 15 and references therein] show significant heterogeneity, albeit with somewhat weak effects, of **13 + 37** versus **67**, and **13 + 67** versus **37**, i.e., neither of the pairwise combinations alone can explain all of the disease risk. However, the two combinations of **AAs: 13, 37, 74, 86** and **13, 67, 74, 86** (Table 5); both uniquely define all JIA-OP disease risk as well as all the alleles listed in Table 1, except for the non-significant effects of the predisposing DRB1*1103 (*rare*) and *1104 alleles, which is distinguished by AA 71.

We have shown the power of inclusion of complementary analyses in the SFVT pipeline (Figure 1), some AAs may be detected via one analysis and not another, e.g., **AA 86** which was not identified in the initial SFVT results. There is consistency in the results from the analyses when all information is integrated.

We have fully automated all the analysis methods outlined in the pipeline in Figure 1, except for the Conditional Haplotype Method, which will be added in the future. At this stage, the different methods are run separately, with the researcher providing input when results from one method are required as input for another method. While many of these steps may be automated in the future, nonetheless it may not be feasible, or wise, to completely do away with oversight of some of these steps by the researcher.

Future studies of this JIA data set [12] will include SFVT analysis of: other common clinical subsets of JIA which show heterogeneity in their HLA DRB1 associated alleles; HLA class II DRB1-DQA1-DQB1 data (no effect of DQB1 is currently seen with allele and haplotype level analyses); and other classical HLA genes which via conditional haplotype method analysis have shown significant effects on disease risk, e.g., the class II DPB1 gene, as well as class I genes.

Acknowledgements

This research was supported by NIH contracts and grants: AI40076 (GT, NM, RMS, PAG, DRK, RHS), AI67068 (JAH, SJM, HAE), AI67150, AR42272, and AR47363 (SDT, DNG).

References

1. E. Thorsby, *Hum Immunol.* **53**, 1 (1997)
2. G. Thomson, A. M. Valdes, J. A. Noble JA, *et al.*, *Tissue Antigens* **70**, 110 (2007).
3. H. Salamon, J. Tarhio, K. Rønningen and G. Thomson, *J. Comp Biol.*.**3**, 497 (1996).
4. A. M. Valdes, S. McWeeney and G. Thomson, *Am. J. Hum. Genet.* **60**, 717 (1997)
5. J. B. Imboden, *Annu. Rev. Pathol. Mech. Dis.* **4**, 417 (2009).
6. D. R. Karp, N. Marthandan, S. G. E. Marsh, *et al.*, *Hum. Mol. Genet.* under revision (2009).
7. O. D. Solberg, S. J. Mack, A. K. Lancaster, *et al.*, *Hum. Immunol.* **69**, 443 (2008).
8. D. Meyer and G. Thomson, *Ann. Hum. Genet.* **65**, 1 (2001).
9. C. Leisner, N. Loeth, K. Lamberth, *et al.*, *PLoS One* **27**, e1678 (2008)
10. M. Nielsen, C. Lundegaard, P. Worning, *et al.*, *Bioinformatics* **20**, 1388 (2004)
11. N. Frahm, B. Baker, C. Brander, *In HIV Molecular Immunology200, Los Alamos, NM,* 3 (2008)
12. J. A. Hollenbach, S. D. Thompson, T. L. Bugawan, *et al.*, *Rheu. Arth.* under revision (2009)
13. H. Payami, S. Joe, N. R. Farid, *et al.*, *Am. J. Hum. Genet.* **45**, 541 (1989)
14. P. W. Hedrick, *Genetics* **117**, 331 (1987)
15. G. Thomson, L. F. Barcellos, A. M. Valdes. *Advances in Genetics* **60**, 255 (2008)

ÇOKGEN: A SOFTWARE FOR THE IDENTIFICATION OF RARE COPY NUMBER VARIATION FROM SNP MICROARRAYS

GÖKHAN YAVAŞ[1], MEHMET KOYUTÜRK[1,3], MERAL ÖZSOYOĞLU[1], MEETHA P. GOULD[2], THOMAS LAFRAMBOISE[2,3]

[1]*Department of Electrical Engineering & Computer Science,* [2]*Department of Genetics,* [3]*Center for Proteomics & Bioinformatics*

Case Western Reserve University, Cleveland, OH, 44106, USA

Until fairly recently, it was believed that essentially all human cells harbor two copies of each locus in the autosomal genome. However, studies have now shown that there are segments of the genome that are polymorphic with regard to genomic copy number. These copy number variations (CNVs) have a role in various diseases such as Alzheimer disease, Crohn's disease, autism and schizophrenia. In the effort to scan the entire genome for these gains and losses of DNA, single nucleotide polymorphism (SNP) arrays have emerged as an important tool. As such, CNV identification from SNP array data is attracting considerable attention as an algorithmic problem, and many methods have been published over the last few years. However, many of the existing model-based methods train their models based on common variations and are therefore less successful in the identification of rare CNVs, detection of which may be very important in personalized genomics applications. In this paper, we formulate CNV identification explicitly as an optimization problem with an objective function that is characterized by several adjustable parameters. These parameters can be configured based on the characteristics of the experimental platform and target application, so that the solution to the optimization problem is the most accurate set of CNV calls. Our method, termed ÇOKGEN, efficiently solves this problem using a variant of the well-known heuristic simulated annealing. We apply ÇOKGEN to data from hundreds of samples, and demonstrate its ability to detect known CNVs at a high level of sensitivity without sacrificing specificity, not only for common but also rare CNVs. Furthermore, we show that it performs better than other publicly-available methods. The configurability of ÇOKGEN, its computational efficiency, and its accuracy in calling rare CNVs make it particularly useful for personalized genomics applications. ÇOKGEN is implemented as an R package and is freely available at http://mendel.gene.cwru.edu/laframboiselab/software.php.

1. Introduction

Identification of DNA variants that contribute to disease is a central aim in human genetics research and has immediate applications in personalized genomics. Pinpointing these causal loci requires the ability to accurately assess DNA sequence variation, on a genome-wide scale. In recent years, considerable progress has been made in identifying and cataloging single-nucleotide polymorphisms (SNPs) in many populations [1]. Commercial SNP microarray platforms can now genotype, with >99% accuracy, over one million SNPs in an individual in one assay [2, 3].

The discovery of copy number variants (CNVs) as a significant source of variation has complicated the identification of genetic differences among humans. CNVs are defined as chromosomal segments, at least 1000 bases (1 kb) in length that vary in number of copies from human to human [4-8]. Since their discovery, several high-profile studies have been published associating copy number variation in the genome with a variety of common diseases. Recent examples include Alzheimer disease [9], Crohn's disease [10], autism [11], and schizophrenia [12]. The significance of the gains (copy number greater than two) and losses (copy number less than two) that comprise these variants is increasingly evident, and cataloging them and assessing their frequencies has become an important goal.

SNP arrays contain hundreds of thousands of unique nucleotide probe sequences, each designed to hybridize to a target DNA sequence. When a DNA sample is properly prepared and applied to the array, specialized equipment can produce a measure of the intensity of hybridization between each probe and its target in the sample. The

underlying principle is that the hybridization intensity depends upon the amount of target DNA in the sample, as well as the affinity between target and probe. Extensive processing and analysis of these raw intensity measures yield estimates of some characteristic of the target sequences in the sample - either target quantity [13, 14], base composition [15, 16], or both. In copy number inference, the objective is to identify chromosomal regions at which the number of copies per cell deviates from two. These include gains and losses.

There is now a large body of literature describing algorithms to infer copy number from SNP array data. All such algorithms address one or more of the three general steps: normalization, raw copy extraction, and CNV calling. Normalization is performed on the raw array intensity data in order to be able to compare these values fairly, thereby taking into account differences in overall array brightness and additional sources of nuisance variation. Raw copy number extraction entails converting the multiple measurements for each genomic site into a single raw measure of copy number. The word "raw" here indicates that measurements from surrounding loci are not yet taken into account, and the measure is permitted to be non-integer. However, since gains and losses occur in discrete segments often encompassing several such loci, true copy number is locally constant. Consequently, the final CNV calling step takes advantage of this fact, smoothing or segmenting the raw copy numbers into discrete segments of consistent copy number.

For the Affymetrix platform, the community has largely settled upon quantile normalization [17] as a simple, yet effective, normalization method. The next step, raw copy number extraction, typically entails fitting some model to raw probe intensity data [18-21]. Methods devoted the final step – making CNV calls from raw copy number data – are numerous, and employ various strategies. Three commonly-used strategies are hidden Markov models (HMMs) [21, 22], circular binary segmentation [23, 24], and adapted weight smoothing [25, 26]. Although these methods appear to be quite different from one another in terms of the computational or statistical model they incorporate, at the core of each is an objective function whose optimum solution yields the method's copy number inference for a region. Each objective function is defined by the observed data (raw copy number) and is a function of inferred state (copy number call). The sequence of copy number calls (states) that optimizes the objective function gives the CNV call for each method.

In this paper, we describe a software tool, ÇOKGEN, which implements a novel optimization algorithm for identification of CNVs from raw copy number, based on an objective function that is composed of several explicitly formulated objective criteria. These criteria are carefully designed to quantify the desirability of a CNV assignment with respect to various biological insights and experimental considerations. Our general approach is to first apply a signal processing method to aggressively flag candidate gains and losses. The objective function is then optimized on each region and flanking sequence, yielding final CNV calls and boundaries. Note that the optimization process also filters out many candidate regions; that is, complete rejection of a candidate region is quite possible, as it is part of the solution space for the corresponding optimization problem. This two-step procedure has the advantages of drastically reducing the computational time necessary to find the set of solutions, while identifying precise boundaries for each putative CNV.

A key feature of our method is that it is highly configurable, allowing researchers to define their own objective functions and tune parameters to emphasize relative importance of different objective criteria. We demonstrate with a simple objective function involving a linear combination of variability, parsimony, and length, which performs surprisingly well. We evaluate the performance of our method on Affymetrix 6.0 array data from 270 HapMap individuals [1]. These samples are increasingly well characterized with regard to CNVs and include 60 mother-father-child trios. Therefore, they serve as an excellent benchmark data set. We show via systematic *in silico* studies that it compares favorably with two methods that are currently publicly available. These results demonstrate the proposed method's potential to uncover human genetic variation that other computational approaches may miss.

ÇOKGEN is implemented as an R package that works from the raw binary .CEL files produced by the Affymetrix protocol. It performs the steps including intensity extraction, quantile normalization, raw copy extraction, and CNV extraction (wherein the user may specify the desired objective function). Its graphical tools also allow the user to manually inspect the raw copy number data to gauge confidence in each putative aberration.

2. Methods

ÇOKGEN takes as input the raw .CEL files, and produces a table of inferred gains and losses, genome-wide. It provides a configurable platform for CNV identification, in that it allows users to (i) adjust the parameters of our default formulation to tune the behavior of the method to the target application (e.g., aggressive vs. conservative in calling CNVs), and (ii) specify their own target objective functions. ÇOKGEN also produces "zoomable" plots of raw copy number at the chromosome and sub-chromosome level for manual inspection of identified copy numbers. Details for each step of the framework implemented in ÇOKGEN are described in the following subsections.

2.1. *Intensity Extraction and Normalization of Raw Data*

The raw probe intensities for each array are encoded in the binary .CEL files output by the Affymetrix instrument, one file for each array. As a first step, we use the R package *affxparser* [27] to extract the intensities for each array locus from .CEL files. Next, we quantile normalize [17] the intensities across all arrays in the experiment. This enables fair comparison of intensities, taking into account systematic non-biological differences such as overall array brightness.

2.2. *Raw Copy Number for SNP and CN Markers*

The genomic loci interrogated on the Affymetrix 6.0 array fall into two categories – SNP markers and copy number (CN) markers. The array contains 887,876 autosomal CN and 869,224 autosomal SNP markers, for a total of 1,757,100 (we discard the X and Y chromosomes to avoid gender complications, as well as mitochondrial markers). The markers are ordered from i = 1 to ~1.8 million according to genomic coordinates. A SNP marker is interrogated by either six or eight probes – half for each of the *A* and *B* alleles – and hence produces six or eight normalized intensity measurements for each array. Since the vast majority of SNP markers have six probes, we present that case here. Let A_{i1}, A_{i2}, A_{i3}, B_{i1}, B_{i2}, and B_{i3} denote the three *A* allele and three *B* allele measurements for a SNP marker *i*. Our aim is to produce allele-specific raw copy numbers A_i and B_i for the two alleles such that the distance from the origin in (*A*, *B*) Cartesian coordinates produces a raw measure of the copy number at the i^{th} marker. Toward this end, we linearly rescale the intensities so that $\sqrt{A_i^2 + B_i^2}$ is approximately equal to 2.0, regardless of genotype, for markers that are already deemed to have normal copy numbers (i.e., two copies).

We fit the model

$$Z_i^{(A)} = \alpha_{i1}^{(A)} A_{i1} + \alpha_{i2}^{(A)} A_{i2} + \alpha_{i3}^{(A)} A_{i3} + \beta_{i1}^{(A)} B_{i3} + \beta_{i2}^{(A)} B_{i2} + \beta_{i3}^{(A)} B_{i3} + e_i^{(A)} \quad (1)$$

via least-squares regression, where $Z_i^{(A)}$ is the rescaled (true) copy number for allele A at SNP *i*; $\alpha_{ij}^{(A)}, \beta_{ij}^{(A)}$ for $1 \le j \le 3$ are model parameters, and $e_i^{(A)}$ is the error term. More specifically, in the absence of copy number variation, $Z_i^{(A)}$ is 2.0 for an *AA* genotype, $\sqrt{2}$ for an *AB* genotype, and 0 for a *BB* genotype. The fitting procedure yields estimates $\hat{\alpha}_{i1}^{(A)}, \hat{\alpha}_{i2}^{(A)}, \hat{\alpha}_{i3}^{(A)}, \hat{\beta}_{i1}^{(A)}, \hat{\beta}_{i2}^{(A)}, \hat{\beta}_{i3}^{(A)}$ for the model parameters. We model *B* allele copy number in a similar manner, and obtain estimates $\hat{\alpha}_{i1}^{(B)}, \hat{\alpha}_{i2}^{(B)}, \hat{\alpha}_{i3}^{(B)}, \hat{\beta}_{i1}^{(B)}, \hat{\beta}_{i2}^{(B)}, \hat{\beta}_{i3}^{(B)}$ for the model parameters, quantifying the relationship between *B* allele copy number and the six probe intensities. The objective here is to capture the individual responsiveness of each probe to varying quantities of DNA harboring the *A* and *B* alleles.

Note, however, that fitting the models requires *a priori* knowledge of the genotypes for the values of true allelic copy numbers $Z_i^{(A)}$ and $Z_i^{(B)}$. Affymetrix's default algorithm is quite precise (over 99.5% accurate) for diploid genotyping. Hence, if we were able to avoid samples with duplications and deletions, we could use the genotypes generated by Affymetrix as observed values of *A* and *B* copy numbers. Obviously, we cannot assume knowledge of which samples harbor gains and losses. However, we can utilize basic knowledge on the distribution of copy numbers as evidence suggests that gain and loss events almost always appear in the small minority variant in the population [28]. Therefore, if we define total probe intensity at marker *i* as $PI_i = \sum_{j=1}^{3} A_{ij} + \sum_{j=1}^{3} B_{ij}$, we can safely assume in general that most of the middle two quartiles, across all samples, of PI_i are from individuals with two copies of the chromosomal segment that contains marker *i*. In other words, the individuals that fall into these quartiles for the corresponding marker are likely to carry diploid genotypes *AA*, *AB*, or *BB*. Consequently, we fit

the model based on these samples' genotypes. Note that, in rare cases, it is possible that the dominant allele in the population may deviate from copy number two. In these cases, the proposed method will still detect the CNV but copy number two individuals will appear as having losses or gains at those loci.

Given the 12 parameter estimates for a SNP marker i, we generate raw estimates of A and B copy numbers for all samples by re-applying the model to each sample's six probe intensities. That is, for a sample with probe intensity values A_{i1}, A_{i2}, A_{i3}, B_{i1}, B_{i2}, and B_{i3}, the raw A and B allele copy estimates are A_i and B_i where

$$A_i = \hat{\alpha}_{i1}^{(A)} A_{i1} + \hat{\alpha}_{i2}^{(A)} A_{i2} + \hat{\alpha}_{i3}^{(A)} A_{i3} + \hat{\beta}_{i1}^{(A)} B_{i1} + \hat{\beta}_{i2}^{(A)} B_{i2} + \hat{\beta}_{i3}^{(A)} B_{i3} \quad (2)$$

$$B_i = \hat{\alpha}_{i1}^{(B)} A_{i1} + \hat{\alpha}_{i2}^{(B)} A_{i2} + \hat{\alpha}_{i3}^{(B)} A_{i3} + \hat{\beta}_{i1}^{(B)} B_{i1} + \hat{\beta}_{i2}^{(B)} B_{i2} + \hat{\beta}_{i3}^{(B)} B_{i3} \quad (3)$$

Finally, using these estimates, we calculate the raw copy number R_i at marker i as the distance from the origin in the (A, B)-plane: $R_i = \sqrt{A_i^2 + B_i^2}$.

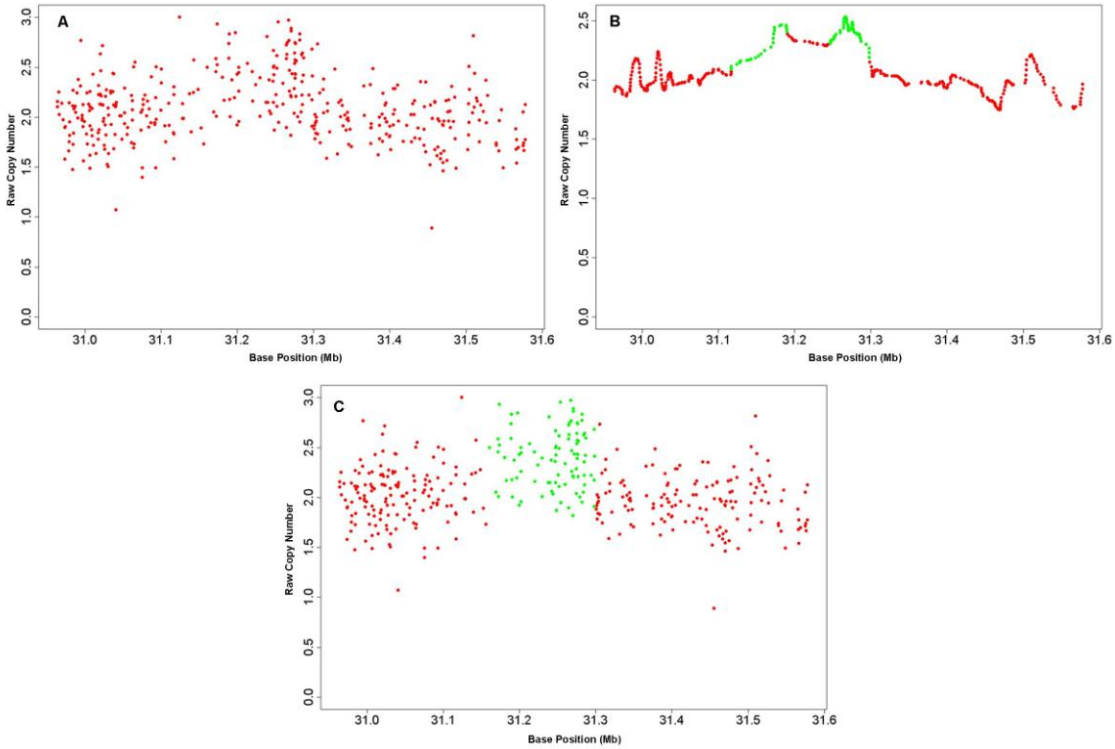


Figure 1. Raw copy numbers for sample NA12763 in a chromosome 12 region. In (A), the raw copy numbers, R_i, for the specified region, are presented. (B) The identified candidate gain regions on the smooth signal R_i^*, which is obtained by applying a low pass filter to R_i. The green colored markers indicate a "gain" class value assignment, whereas the red markers indicate "normal" class assignment by the edge detection algorithm. (C) Optimization of the objective function using simulated annealing makes the final assignments to the markers.

Note that approximately half of the marker loci represented on the 6.0 array correspond to CN markers. Since these markers are each measured by only one probe, they must be treated separately. As above, we consider the samples within the middle two quartiles of (normalized) total probe intensity for the marker to be representative of individuals with copy number two. Therefore, the scaling factor $\hat{\beta}_i$ for CN marker i is the least-squares estimate of the parameter β_i from the model

$$2 = \beta_i PI_i + e_i \quad (4)$$

fit to the middle two quartiles of the normalized probe intensities PI_i. Again, e_i is the error term. The raw copy number for a sample with CN probe intensity PI_i is then calculated as $R_i = \hat{\beta}_i PI_i$.

Using these two separate procedures for SNP and CNV markers yields raw copy numbers R_i for all markers i from 1 to ~1.8 million. Figure 1A, which is generated by ÇOKGEN, gives an example of raw copy numbers for a 394-marker region.

2.3. *Copy Number Variant Detection using Optimization*

Key to our approach is the observation that CNV identification can be formulated explicitly as an optimization problem without any requirement of reference models or training data. Based on general knowledge of the microarray technology and basic biological insights on copy number variation, we specify various quantitative measures that gauge the suitability of copy number assignments based on observed array intensities. We then formulate an objective function that captures the trade-off between these measures, so that the minima of this function represent optimal CNV assignments. This function is characterized by user-defined parameters, allowing the user to tune the performance of algorithms based on the requirements of the specific application (*e.g.,* minimizing false positives due to the cost of experimental verification *vs.* minimizing false negatives to capture existing variation comprehensively).

Formally, the objective of CNV identification is to find a mapping S: {1,..., N} → C, where {1,..., N} denotes the ordered set of markers for the whole genome and C = {C_+, C_0, C_-} is the set of the gain, normal and loss classes, denoted respectively as C_+, C_0 and C_-. Thus, our objective is to assign a class value from C to each marker on genome based on the R_i values such that the class assignment of consecutive markers and their raw copy number estimates are as consistent as possible.

In next subsections, we introduce the objective criteria that are included in the default objective function implemented in ÇOKGEN and the motivation behind these criteria. Researchers may wish to design an objective function of their choice, and indeed our software takes the objective function as an argument precisely to accommodate this. We describe the function as applied to a chromosome with M markers since each chromosome is processed separately.

2.3.1. Variability in raw copy numbers within each copy class should be minimized

The R_i for markers in each gain or loss region should be separable from normal regions. Therefore, CNV identification lends itself to a clustering-like problem – one of partitioning the R_i's into three classes so as to minimize the internal variability of each class. For a given CNV assignment S, we define the set of markers assigned to class c on a chromosome with M markers as $\Pi(c) = \{i \in \{1,\ldots,M\} : S(i) = c\}$ and $\mu_c = \frac{\sum_{k\in\Pi(c)} R_k}{|\Pi(c)|}$ denotes the mean raw copy number for class c. Then, the total intra-class variability induced by this assignment is given by

$$\sigma(S) = \sum_{c\in\{C_+,C_0,C_-\}} \sum_{k\in\Pi(c)} |R_k - \mu_c| \qquad (5)$$

Consequently, a desirable S is expected to minimize $\sigma(S)$ (subject to other constraints). Note that this formulation does not make any assumption about the expected raw copy numbers of the markers and therefore is robust to any systematic bias that might be encountered in measurement and normalization of R_i.

2.3.2. Parsimony principle: Observed variability should be explained via minimum number of anomalies

In general, there are relatively few regions of gain or loss in an individual's genome, relative to normal regions. Therefore, the CNV calls should be as contiguous as possible. Motivated by this observation, we formulate the parsimony principle as a criterion that seeks to minimize the total number of copy number state changes induced by a CNV assignment on the chromosome. Formally, for given CNV assignment S, we define total cut as the number of pairs of adjacent markers that are assigned different copy numbers, $\chi(S) = \sum_{k=1}^{M-1} I(S(k) \neq S(k+1))$. Here I(.) denotes the indicator function (i.e., it is equal to 1 if the statement being evaluated is true, and 0 otherwise).

2.3.3. Filtering out noise by eliminating smaller regions

Longer CNVs indicate higher confidence as it can be statistically argued that shorter sequences of markers with deviant raw copy numbers are more likely to be observed due to noise. Thus, we explicitly consider CNV length as

an additional objective criterion. To do so, we first define a CNV region, r, as a maximal set of contiguous markers all assigned to the same copy number state in $\{C_+, C_-\}$, and $\varsigma(S)$ denotes the set of all CNV regions. Furthermore, we denote the number of markers in the CNV region r by $l(r)$. We then define $\lambda(S) = \sum_{r \in \varsigma(S)} \frac{1}{e^{l(r)}}$ as an objective criterion that penalizes shorter CNVs (e denotes the natural logarithmic base).

2.3.4. Filtering out noise by eliminating possible false positives

Candidate CNVs with a median raw copy number much larger or much smaller than two indicate higher confidence since a CNV region with median raw copy number close to two is less likely to be valid. For this reason, we require that the median raw copy number of a called loss be below a certain threshold (T_{loss}) and the median for a called gain be above a certain threshold (T_{gain}). We define $\varsigma^+(S)$ and $\varsigma^-(S)$ as the set of all CNV gain and loss regions, induced by assignment S, respectively. Furthermore, median(r) denotes the median raw copy number value of the markers in the region r. We now incorporate $\delta(S) = \sum_{r \in \varsigma^+(S)} I(\text{median}(r) < T_{\text{gain}}) + \sum_{r \in \varsigma^-(S)} I(\text{median}(r) > T_{\text{loss}})$ into the objective function to minimize the effects of the noisy signal. Here, T_{gain} and T_{loss} are user-defined parameters which basically define the upper and lower limits for the raw copy number of markers in the set $\Pi(C_0)$ (i.e., the set of markers assigned to the normal class). As T_{gain} is increased and T_{loss} is decreased, candidate regions are penalized more harshly. In our experiments, we use 2.35 and 1.65 for T_{gain} and T_{loss} respectively, since these values provide reasonable performance.

2.3.5. Putting the pieces together: A single objective function for CNV identification

We use a linear combination of the criteria above as an objective function. Namely, we define the optimal copy number assignment as the mapping S^*: $\{1,\ldots, N\} \rightarrow C = \{C_+, C_0, C_-\}$ such that the function

$$f(S) = k_\sigma \sigma(S) + k_\chi \chi(S) + k_\lambda \lambda(S) + k_\delta \delta(S) \qquad (6)$$

is minimized at $S = S^*$. We briefly talk about how these parameters are adjusted in section 3.5.

2.4. *Two Phase CNV Identification*

Since the solution space of the optimal copy number assignment problem is exponential in the number of markers we require a good initial solution and a heuristic algorithm which iteratively improves the solution. For this purpose, we use a two-phase algorithm: (i) we first determine a set of candidate gain and deletion regions via a filtering and aggressive edge detection procedure which we consider as an initial CNV assignment, $S^{(0)}$; (ii) we employ an iterative improvement based algorithm to adjust the boundaries of duplications and deletions accurately, and eliminate false positives.

In order to identify the boundaries for CNV regions, it is necessary to smooth the raw copy number signal since it is highly noisy. We use a simple discrete low-pass filter with filter kernel [1/3; 1/3; 1/3], i.e., the first filtered copy number estimate is given by $R_i^{(1)} = \frac{R_{i-1} + R_i + R_{i+1}}{3}$. Applying the filter for a second time, we obtain $R_i^{(2)} = \frac{R_{i-1}^{(1)} + R_i^{(1)} + R_{i+1}^{(1)}}{3} = \frac{R_{i-2} + 2R_{i-1} + 3R_i + 2R_{i+1} + R_{i+2}}{9}$. Consequently, introducing an adjustable repetition parameter W, we obtain $R_i^* = R_i^{(W)}$ as a smooth version of the copy number intensity for a user defined value of W. Here, larger W provides smoother signals, thereby eliminating false positives, at the cost of missing true CNVs that span a smaller number of markers. For the ÇOKGEN's default value, we chose W=20, for which we obtain reasonable results. Figure 1B demonstrates how the raw copy numbers R_i in Figure 1A is converted into a smooth signal R_i^* using the low pass filter.

2.4.1. Identification of Candidate CNV Regions via Edge Detection

Based on the observation that gains and losses manifest themselves as (respectively up or down) concavities in raw copy number of the low-pass filtered data, an edge detection scheme, which we describe below, is a useful tool for the identification of initial CNV assignment $S^{(0)}$. Thus, after low-pass filtering, we apply our edge detection algorithm on the smoothed signal, first identifying high gradient markers that may correspond to transitions between regions with different copy numbers. For this purpose, we interpolate the discrete signal to obtain a real-valued function on the continuous interval $\hat{R}:[0, \mathrm{M}] \to \Re$. This task is performed using the built-in *splinefun* function of R language, which performs cubic spline interpolation of given data points. Next, we generate two sets of high-gradient markers, denoted $D_{\max}$ and $D_{\min}$, for which the function $\hat{R}(i)$ attains maximum increase and maximum decrease, respectively. Specifically, we define

$$\begin{aligned} D_{\max} &= \{i \in 2,\ldots,\mathrm{M} \mid \hat{R}'(i) > \hat{R}'(i-1) \text{ and } \hat{R}'(i) > \hat{R}'(i+1)\} \\ D_{\min} &= \{i \in 2,\ldots,\mathrm{M} \mid \hat{R}'(i) < \hat{R}'(i-1) \text{ and } \hat{R}'(i) < \hat{R}'(i+1)\} \end{aligned} \tag{7}$$

where $\hat{R}'(i)$ denotes the derivative of $\hat{R}(i)$ at marker i. These markers are the approximate inflection points of the signal $\hat{R}(i)$.

Now let Q_{ij} denote the indices corresponding to the set of contiguous markers on the genome starting from marker i and ending at marker j, where $i \le j$. Given the user defined thresholding parameter T_{gain} (see above), we designate Q_{ij} as a candidate gain region (i.e., $\forall k \in Q_{ij}$, $S^{(0)}(k) = C_+$) if it satisfies the following conditions:

1. $i \in D_{\max}$ and $j \in D_{\min}$
2. there exists at least one marker p, $i \le p \le j$, such that $\hat{R}(p) \ge T_{\text{gain}}$
3. $\max(Q_{ij} \cap D_{\max}) < \min(Q_{ij} \cap D_{\min})$
4. Q_{ij} is a maximal set of contiguous markers satisfying the above 3 conditions.

The first condition ensures that the region starts with a marker with locally maximal positive gradient and ends with a marker with locally maximal negative gradient in terms of the raw copy number values. The second condition guarantees that the region contains markers with copy number estimates that might indeed correspond to a gain. The third condition specifies that the region does not contain any interior concavities, i.e. all maximum positive gradient markers in Q_{ij} appear before any maximum negative gradient marker in the region. Finally, condition 4 ensures that Q_{ij} can be enlarged neither at the right nor the left borders. The designation of Q_{ij} as a candidate loss region is done in a completely analogous manner.

All markers m that are not included in a candidate loss or gain region are preliminarily designated as "normal", i.e., $S^{(0)}(m)$ is set to C_0. As a special case, if a candidate gain/loss region identified by edge detection is very close to another candidate region of its type, then we merge these two candidate regions into a single region, since they are likely to correspond to the same aberration.

This procedure gives us an initial CNV identification assignment $S^{(0)}$. As an example, two candidate gain regions identified by the edge detection algorithm are presented in Figure 1B. The green colored markers indicate a "gain" class value assignment, whereas the red markers indicate "normal" class assignment by the described algorithm. Note that, although there are no "loss" class valued markers in the figure, they are colored with blue by ÇOKGEN's visualization tool.

The initial solution is quite aggressive in the sense that many truly normal (copy number two) markers are likely to be placed in the gain or loss classes. To eliminate these false positives and obtain S^*, we use an optimization-based algorithm to tune the boundaries of candidate gain and deletion regions as discussed in the next section.

2.4.2. Fine Tuning of the Region Boundaries using Optimization with Simulated Annealing

This phase of the algorithm begins with initial class assignments, $S^{(0)}$, and iteratively improves it with regard to the value of the objective function f by making moves in a way to quickly reach an optimum and avoid being trapped into undesirable local optima. Note that, while we assume here that $S^{(0)}$ is obtained using the edge detection

procedure presented in the previous section, the optimization procedure presented in this section can be used to refine boundaries generated by any initial segmentation procedure, such as [23, 24].

For a given copy number assignment *S*, we define a *move* as the extension or contraction of a CNV region's boundaries by changing the copy number states assigned to a contiguous group of markers (either inside or outside the region) bordering the region. In short, at each iteration of the algorithm, a random number of contiguous markers is selected from the right or left boundary of a candidate region $Q_{ij} \in \varsigma(S)$ and the corresponding move is defined as the assignment of these markers to either the class of neighboring markers (if the selected markers belong to Q_{ij}) or to Q_{ij}'s class (if the selected markers are outside of Q_{ij}). The concept of a move is illustrated in Figure 2. As seen in the figure, we restrict possible moves to those that can enlarge, shrink, or merge candidate aberrant regions, but can never create a candidate region from scratch or divide a candidate region into two candidate regions. Indeed, we observe that the average distance between two consecutive CNPs reported by McCarroll *et al.* [28] is 2.11 Mb, indicating that it is unlikely for edge detection to misidentify two disjoint CNVs as a single merged CNV.

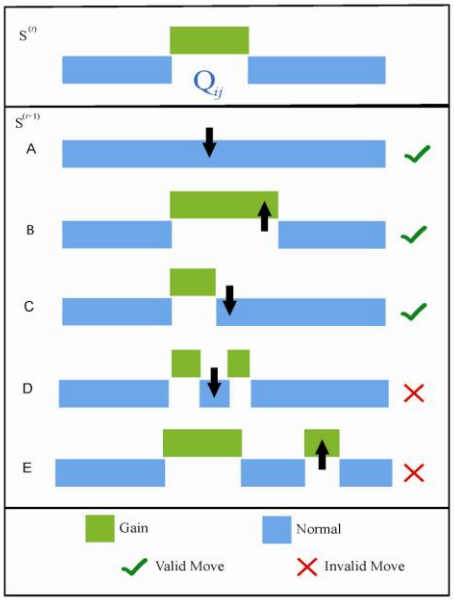


Figure 2. Illustration of the concept of "move" for the proposed iterative improvement algorithm. A valid move is defined as the reassignment of the class of a contiguous group of markers that is within or near a candidate CNV region. At each step of the algorithm, a valid potential move is selected randomly and it is done if it improves the objective function. If it does not improve the objective function, then it is done with probability inversely proportional to its cost on the objective function.

We quantify the quality of a potential move in terms of the difference between the value of the objective function before and after the move, commonly referred to as the *gain* of a move. The gain associated with move *v* is defined as $\gamma(v) = f(S^{(t)}) - f(S^{(t+1)})$ where $S^{(t+1)}$ denotes the copy number assignment if the move *v* were made and $S^{(t)}$ is the current copy number assignment. While it is possible to find the move with maximum gain by exhaustively searching the valid move set, instead we use a stochastic algorithm that is based on simulated annealing [29]. Simulated annealing is an iterative improvement heuristic that proceeds by repeated moves to improve the quality of the solution. Key to its efficiency is the stochastic nature of the selection of moves. At each step, the algorithm first randomly chooses a candidate gain or loss region, Q_{ij}, from the set $\varsigma(S)$ and then chooses a move *v* from the set of all moves that are validly defined on Q_{ij}. If the gain $\gamma(v)$ associated with the candidate move is positive, then the move is made. If the gain is not positive, the move is still made with a certain probability, which is proportional to the gain and declines as a function of time in the course of the algorithm. Therefore, simulated annealing starts its course with aggressive moves to jump out of undesirable local optima, and becomes more conservative as the algorithm proceeds, smoothly converging to a locally optimum solution. In our application, we set an upper limit of five (in terms of number of markers) on the permissible expansion of a CNV region. The procedure is repeated until either there is no positive gain move left to be done on the current solution or a user-defined number of negative gain moves, τ, are already done consecutively, (for our default, we use $\tau = 5$). The mapping obtained at the end of the procedure is reported as *S**.

3. Results and Discussion

We applied our algorithm to Affymetrix 6.0 array data from 270 HapMap individuals. The HapMap samples are divided into African (YRI), Caucasian (CEU) and Asian (CHB/JPT) ethnicities. ÇOKGEN identified a total of 16739 autosomal CNVs over all the samples, for an average of 62 CNVs per individual. Of the 16739 CNVs, 1033 are singletons found uniquely in one individual. A recent study by McCarroll *et al.* [28] identified 1292 autosomal copy number polymorphism (CNP) regions in 270 HapMap samples. Nearly 25% of these CNPs were also

identified by ÇOKGEN The distribution of the CNVs among different ethnicities in the population, as well as the overlap and difference in between the McCarroll *et al.* study and ÇOKGEN are presented in Table 1.

Table 1. The statistics of CNVs identified by ÇOKGEN. The distribution of identified CNVs by ethnicity is shown on the four left-most columns. The comparison of CNPs reported by McCarroll *et al.* and CNVs identified by ÇOKGEN is shown on the two right-most columns. Here McCarroll ∩ ÇOKGEN indicates the counts of CNVs identified by both, whereas McCarroll \ ÇOKGEN indicates the counts identified only by the McCarroll study.

	CEU	**YRI**	**JPT**	**CHB**	**Total**	**McCarroll ∪ ÇOKGEN**	**McCarroll \ ÇOKGEN**
Gains	1711	229	985	786	5777	1357	6145
Losses	3749	370	172	1783	10962	8972	26043
Total	5460	600	271	2569	16739	10329	32188

3.1. *Trio Discordance as a CNV Detection Assessment Tool*

Although CNVs can arise in a *de novo* manner, it is believed that at least 99% of all CNVs in an individual's genome are inherited [28]. The 60 mother-father-child trios in the HapMap data set therefore provide an opportunity to assess the accuracy of CNV detection algorithms by measuring the rate of Mendelian concordance. A CNV in a trio child is said to be Mendelian concordant if it appears in at least one of the parents. Unless the CNV is *de novo*, any discordance is either the result of a false positive call in the child or a false negative call in one of the parents (in rare cases, discordance could also result from a parent harboring a duplication and a deletion at the same locus but on different chromosomal homologs). Discordance rate, while useful, is imperfect as an assessment measure. In particular, it is possible for a CNV identification algorithm to have artificially low discordance rates by calling each CNV in a large number of samples. Even if the samples in which a gain or loss is called are randomly selected, frequently called CNVs will have a lower discordance rate, simply by chance.

3.2. *Performance of ÇOKGEN in Comparison to Existing Software*

We compared the performance of our algorithm with that of two other software packages. The DNA-Chip Analyzer (dChip) [30] is a Windows software package for high-level analysis of gene expression microarrays and SNP microarrays [18, 31]. Birdseye [21] is a rare CNV identification tool based on the hidden Markov models. It is part of the Birdsuite platform [21] ,which is a fully open-source set of tools to detect and report SNP genotypes, common copy number polymorphisms (CNPs), and novel, rare, or *de novo* CNVs in samples processed with the Affymetrix platform.

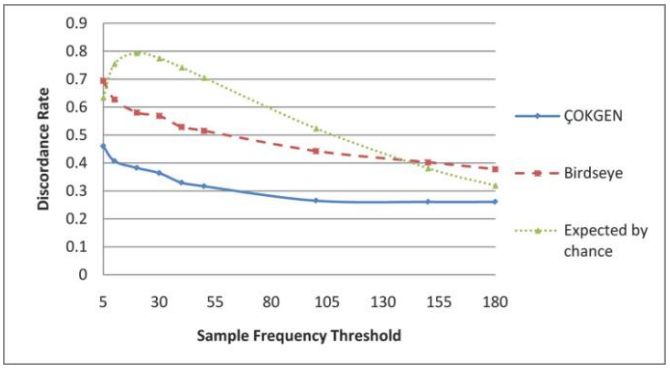


Figure 3. Discordance rate as a function of call frequency strata. For each value of sample frequency threshold, the y-axis shows the average discordance rate for all copy number calls that are less frequent than the threshold.

ÇOKGEN outperformed both Birdseye and dChip in terms of general trio discordance. Overall, it has a 26% discordance rate whereas Birdseye and dChip demonstrate discordance rates of 37% and 93%, respectively on the same array data. dChip was originally optimized for detecting somatic copy number aberrations in cancer cells from earlier versions of the Affymetrix platform. Therefore, Birdseye and ÇOKGEN's superior performance compared to dChip is not surprising. For this reason, we restrict our assessment to ÇOKGEN and Birdseye for the remainder of this section.

As discussed in previous section, the expected discordance rate of an algorithm approaches zero as it calls a CNV in more samples in a data set of trios.

At the extreme, if the algorithm identifies a CNV in all samples, the discordance rate will be zero. Therefore, a more precise assessment of accuracy can be achieved by stratifying discordance rate by call frequency. For this purpose, in Figure 3, we first examine how the discordance rate changes across call frequency strata for ÇOKGEN and Birdseye. As a reference, we also display the expected discordance of randomly called CNVs in this figure. Note that discordance rate is plotted for CNVs with frequencies *at most* the corresponding value on the x-axis. As expected, the performance of both algorithms improves when we consider more frequent CNVs. Nevertheless, it is clear in the Figure 3 that ÇOKGEN outperforms Birdseye significantly at all strata. Furthermore, for up to a frequency threshold of five samples, if the CNV calls were to be done totally at random, it is possible to obtain a better discordance rate than Birdseye. Similarly, for more frequent CNVs, (for frequency threshold larger than 150) random CNV calling performs better than Birdseye at all strata. However, ÇOKGEN performs consistently better than random CNV assignment at all strata which shows its superior performance is not an artifact of the frequency of the CNVs it calls.

Another feature of Figure 3 is Birdseye's sharper decline in discordance rate as the frequency threshold increases. This is likely due to its higher average call frequency as compared to ÇOKGEN. We find that 40% of the concordant CNVs identified by Birdseye have a sample frequency larger than 60, whereas only 20% of the concordant CNVs identified by our algorithm have frequency larger than 60. Concordant CNVs with sample frequency larger than 90 make up 4% of those called by our algorithm as compared to 27% for Birdseye. This clearly shows that ÇOKGEN does not achieve its high concordance rate by overcalling a CNV in multiple samples. When we analyze the density distribution of the discordant CNVs as a function sample frequency for both algorithms, we observe that most of the discordant CNVs for Birdseye are rare whereas more frequent CNVs called by our algorithm turn out to be discordant. These two observations clearly show that ÇOKGEN's performance depends less on the sample frequency and demonstrate its ability to accurately detect rare events.

3.3. *Sensitivity comparison across methods*

Trio discordance is a good hybrid measure of sensitivity (recall) and specificity (precision), but these two measures cannot be easily decoupled based only on discordance rate. A recent study [32] assembled a "stringent dataset" which contains CNVs identified by at least two independent algorithms. The data set contains a total of 808 autosomal CNV regions reported by the study to be harbored in at least one of the 270 HapMap individuals. We use this as a "gold standard" data set in which to evaluate the sensitivity of our method.

ÇOKGEN detects 725 of 808 (≈90%) CNVs from the study presented in [32]. Birdseye obtains the best result by identifying 760 of 808 (≈94%) CNVs. dChip achieves an 89% success rate which is comparable to our method. Therefore, Birdseye seems slightly more sensitive than ÇOKGEN; however, as shown above, this is likely at the cost of a higher false positive rate.

3.4. *Experimental Validation of CNVs not Previously Reported*

To gauge the ability of ÇOKGEN to uncover novel gains and losses, we also compared the CNVs discovered by our method with those in the version 6 (November 2008) of Database of Genomic Variants (DGV) [33]. We used multiplex ligation-dependent probe amplification (MLPA) [34] to verify some of the CNVs which are not reported in the DGV but are identified by ÇOKGEN. The results of these experiments are shown in Table 2. As seen in the table, the copy numbers estimated by MLPA for each of these regions are concordant with the predictions of ÇOKGEN.

Table 2. MLPA results for some of the copy number variantsidentified by ÇOKGEN, which were not previously reported.

Chr	Sample	Bp Start	Bp End	Length (bp)	MLPA Probe Pos.	Type	MLPA
5	NA11830	59753489	59816458	62969	59766589	Gain	2.4
5	NA10846	101261596	101308054	46458	101261461	Loss	1.35
5	NA12144	101256012	101308054	52042	101279312	Loss	1.18
6	NA10846	99225525	99249603	24078	99237564	Loss	1.44

3.5. *Parameter Adjustment*

As explained in section 2.3.5, we have tunable coefficients $k_\sigma, k_\chi, k_\lambda, k_\delta$ that adjust the relative importance of the objective criteria with respect to each other in our objective function. In our experiments, for k_λ and k_δ, we choose large values such as 10^5 and 10^6, respectively, to prohibitively eliminate candidate regions that are likely to be false positive during the course of the algorithm (as opposed to filtering them out in a post-processing phase).

The parameters k_σ and k_χ are used to adjust the apparent trade-off between the "parsimony" and the "variability" components of the objective function. Variability favors the genetic diversity on the genome by permitting many CNVs. On the other hand, according to the parsimony criterion, the variability in the raw copy estimates of markers should be explained via as few CNVs as possible, hence minimizing the number of evolutionary events that have had to occur. Without loss of generality, we require that $k_\sigma + k_\chi = 1$ to highlight the trade-off between these two criteria. To systematically evaluate the effect of these two parameters on performance and determine the best k_σ and k_χ values based on our benchmarking data, we have conducted a series of computational experiments on the sensitivity and trio discordance. Note that lower discordance is desirable, while we want to maximize sensitivity. In our experiments, we have observed that at $k_\sigma = 0.35$ and $k_\chi = 0.65$, trio discordance curve reaches a global minimum and sensitivity starts saturating after a rapid improvement. As k_σ is increased, ÇOKGEN starts behaving less conservatively, which results in a larger number of identified CNVs and improved sensitivity. On the other hand, increased number of CNVs comes with the expense of increased rate of false positives and this manifests itself as a decline in the discordance rate from a certain value of k_σ (in our case, $k_\sigma = 0.35$). Based on these observations, we set $k_\sigma = 0.35$ and $k_\chi = 0.65$ as our defaults.

3.6. *The Software Package*

Our software package, ÇOKGEN, which is implemented in R, processes each sample individually. When a new sample is to be processed, ÇOKGEN first normalizes its probe values to the HapMap distribution, then uses the coefficients obtained from HapMap samples to get raw copy numbers for the new sample. Next, candidate regions are identified using edge detection and marker class assignments are finalized using optimization with simulated annealing. ÇOKGEN is able to output its results in two forms: tabular and graphical. The tabular output is a table of CNV entries with columns: sample ID, chromosome number, CNV start base position, CNV stop base position, and the CNV type. The graphical output allows the user to visualize the results of our CNV identification algorithm. The user can inspect the raw copy signal at any specified part of the genome along with the assigned class values, color-coded (examples are shown in Figure 1). Another aspect of the graphical output is the visualization of the signals of a family together, in which each member represented by a different plotting symbol. This allows the user to see the CNV pattern for the whole family at the same locus of the genome and evaluate the algorithm's trio concordance visually.

Besides its configurability in terms of tuning of parameters, ÇOKGEN also provides the users with the ability to specify their own objective criteria. With this functionality, users can construct their own objective functions that will best suit the characteristics and needs of their own experimental platform and application.

4. Conclusion

We have presented a method to detect germline copy number variants from Affymetrix 6.0 SNP Array data. Our approach, with its accompanying software, will be useful for researchers querying constitutional DNA for association of gains and losses with disease. Indeed, CNVs are emerging as important factors in a growing number of diseases. Although in this paper, we test the ÇOKGEN's performance on only Affymetrix 6.0 SNP array due to its high genome-wide resolution compared to other commercially-available platforms, our software can be easily applied to any platform using different raw copy extraction methods that are suitable for that platform. ÇOKGEN's ability to uncover rare variants is particularly crucial in the context of personalized genomics, as individual-level variation may have a significant impact on human disease. The current work shows that the problem of detecting CNVs from raw array data may be recast as an optimization problem with an explicit objective function. The

objective function chosen here is quite simple and intuitive, but its effectiveness is clear. Our method is wholly contained in a freely-available and flexible software package [35] that efficiently processes raw probe-level .CEL files to produce lists of inferred gains and losses. The software allows the user to tune parameters for the desired specificity-sensitivity balance. With detailed experimental studies on HapMap dataset, we have demonstrated its sensitivity to detect both previously-reported and novel CNVs, while keeping a low false positive rate, as demonstrated by high Mendelian consistency in trios.

Acknowledgments

This work is supported in part by National Science Foundation Award IIS-0916102.

References

1. International HapMap Consortium. *Nature*, **437(7063)**: 1241-2 (2005).
2. Affymetrix. Genome-Wide Human SNP Array 6.0 data sheet. Santa Clara (California) (2007).
3. Illumina. Human1M-duo beadchip data sheet. San Diego (California) (2007).
4. Feuk L, Carson AR, Scherer SW. *Nat Rev Genet*, **7(2)**: 85-97 (2006).
5. Iafrate AJ, Feuk L, Rivera MN, et al. *Nat Genet*, **36(9)**:949-51 (2004).
6. Tuzun E, Sharp AJ, Bailey JA, et al. *Nat Genet*, **37(7)**:727-32 (2005).
7. Redon R, Ishikawa S, Fitch KR, et al. *Nature*, **444(7118)**:444-54 (2006).
8. Korbel JO, Urban AE, Affourtit JP, et al. *Science*, **318(5849)**:420-6 (2007).
9. Rovelet-Lecrux A, Hannequin D, Raux G, et al. *Nat Genet*, **38(1)**:24-6 (2006).
10. Fellermann K, Stange DE, Schaeffeler E, et al. *Am J Hum Genet*, **79(3)**:439-48 (2006).
11. Sebat J, Lakshmi B, Malhotra D, et al. *Science*, **316(5823)**:445-9 (2007).
12. Xu B, Roos JL, Levy S, et al. *Nat Genet*, **40(7)**:880-5 (2008).
13. Zhao X, Li C, Paez JG, et al. *Cancer Res*, **64(9)**:3060-71 (2004).
14. Peiffer DA, Le JM, Steemers FJ, et al. *Genome Res*, **16(9)**:1136-48 (2006).
15. Gunderson KL, Steemers FJ, Lee G, et al. *Nat Genet*, **37(5)**:549-54 (2005).
16. Lindblad-Toh K, Tanenbaum DM, Daly MJ, et al. *Nat Biotechnol*, **18(9)**:1001-5 (2000).
17. Bolstad BM, Irizarry RA, Astrand M, et al. *Bioinformatics*, **19(2)**:185-93 (2003).
18. Lin M, Wei LJ, Sellers WR, et al. *Bioinformatics*, **20**:1233-40 (2004).
19. Laframboise T, Harrington D, Weir BA. *Biostatistics*, **8(2)**:323-36 (2007).
20. Bengtsson H, Irizarry R, Carvalho B, et al. *Bioinformatics*, **24(6)**:759-67 (2008).
21. Korn JM, Kuruvilla FG, McCarroll SA, et al. *Nat Genet*, **40(10)**:1253-60 (2008).
22. Zhao X, Weir BA, LaFramboise T, et al. *Cancer Res*, **65(13)**:5561-70 (2005).
23. Olshen AB, Venkatraman ES, Lucito R, et al. *Biostatistics*, **5(4)**:557-72 (2004).
24. Venkatraman ES, Olshen AB. *Bioinformatics*, **23(6)**:657-63 (2007).
25. Polzehl J, Spokoiny S. *J R Stat Soc, Ser B*, **62(2)**:335–354 (2000).
26. Hupé P, Stransky N, Thiery JP, et al. *Bioinformatics*, **20(18)**:3413-22 (2004).
27. Affymetrix File Parsing SDK [http://www.bioconductor.org/packages/2.2/bioc/html/affxparser.html].
28. McCarroll SA, Kuruvilla FG, Korn JM, et al. *Nat Genet*, **40(10)**:1166-74 (2008).
29. Kirkpatrick S, Gelatt CD Jr, Vecchi MP. *Science*, **220(4598)**:671-680 (1983).
30. dChip Software Website [http://www.dchip.org].
31. Li C, Wong WH. *PNAS*, **98**:31-6 (2001).
32. Pinto D, Marshall C, Feuk L, et al. *Hum Mol Genet*, **16 Spec No. 2**:R168-73 (2007).
33. Database of Genomic Variants [http://projects.tcag.ca/variation/].
34. Schouten JP, McElgunn CJ, Waaijer R, et al. *Nucleic Acids Res*, **30(12)**: e57 (2002).
35. LaFramboise Lab Software Website [http://mendel.gene.cwru.edu/laframboiselab/software.php].

REVERSE ENGINEERING AND SYNTHESIS OF BIOMOLECULAR SYSTEMS

GIL ALTEROVITZ

Division of Health Sciences and Technology, Harvard University/Massachusetts Institute of Technology, Cambridge, MA 02139, USA. Children's Hospital Informatics Program, Boston, MA 02115, USA. Department of Electrical Engineering and Computer Science, Cambridge, MA 02139, USA. Partners Healthcare Center for Personalized Genetic Medicine, Harvard Medical School, Boston, MA 02115, USA. gil@mit.edu

SILVIO CAVALCANTI

Department of Bioengineering, University of Bologna, Bologna, Italy.

TARO M. MUSO

Partners Healthcare Center for Personalized Genetic Medicine, Harvard Medical School, Boston, MA 02115, USA.

MARCO F. RAMONI

Children's Hospital Informatics Program, Boston, MA 02115, USA. Division of Health Sciences and Technology, Harvard University/Massachusetts Institute of Technology, Cambridge, MA 02139, USA. Partners Healthcare Center for Personalized Genetic Medicine, Harvard Medical School, Boston, MA 02115, USA.

MAY WANG

Department of Biomedical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.

1. Introduction

Synthetic biology is the new frontier of biological engineering. Instead of incrementally altering living organisms, synthetic biologists propose to use biological knowledge, modular biological parts, and computer-aided design to quickly develop systems capable of unprecedented biochemical feats. Synthetic biology therefore promises dramatic improvements in green chemistry [1], alternative energy [2], drug manufacture [3,4], and therapeutirs [5].

There have been numerous recent advancements in synthetic biology. The need for accuracy at the design and simulation stage have inspired dialogue on how to add functional characterizations to parts documentation in the Registry of Standard Biological parts [6,7]. In addition, a design strategy -- constructing networks from quantitatively characterized libraries of diversified components -- has been proposed [8]. A synthetic network must be integrated into an engineering chassis. To this end the development of evolved ribosome-mRNA pairs may be the first step towards an orthogonal cellular network [9 10 11 12].

Although scientists have made significant progress in synthetic biology, the field must still overcome a number of challenges. To this end, this session offers novel methodologies in three general areas: namely, in designing synthetic systems, in developing novel biological parts, and in analyzing complex networks.

2. Session Papers

Design principles and development strategies from other engineering disciplines must be adjusted to the peculiarities of biological systems. **Ball *et al***. propose to approach synthetic biology with heterogeneous design strategies that are common in other engineering fields. **Shea *et al***. propose and demonstrate a compilation strategy for building iterative arithmetic computers based on biochemical reactions. **Ceroni *et al***. propose a tool to predict the response of circuits, such as a synthetic circuit with inducible gene expression. **Tari *et al***. have developed an automated pathway synthesis method that uses knowledge bases and Medline abstracts to automatically synthesize pharmacokinetic pathways.

The design of synthetic systems, of course, is dependent on the availability of well-characterized biological parts. **Davidson *et al***. use an emulsion approach in the development of a

2

library of T7 promoters of varying strength. **Corradin *et al***. explore the potential of a retrovirus HTLV-1 gene circuit as a relaxation oscillator that is deliverable into eukaryotes.

The complexity inherent in synthetic biology implies the need for sophisticated analytical tools. **Ramesh *et al***. employ graph clustering techniques to detect modularity in highly complex gene regulatory networks. **Biasiolo *et al***. study transcriptional and post-transcriptional networks of multiple myeloma samples by measuring the drop in network performance caused by deactivation of putative regulatory elements.

Acknowledgments

Thank you to all the authors who submitted their work to this session and to the reviewers who graciously contributed their time.

References

1. Marguet, P., Balagadde, F., Tan, C. & You, L., *J R Soc Interface* **4**, 607-23 (2007).
2. Lee, S.K., Chou, H., Ham, T.S., Lee, T.S. & Keasling, J.D., *Current Opinion in Biotechnology* **19**, 556-563 (2008).
3. Chang, M.C.Y. & Keasling, J.D., *Nat Chem Biol* **2**, 674-681 (2006).
4. Weber, W., Schoenmakers, R., Keller, B., Gitzinger, M., Grau, T. *et al.*, *Proceedings of the National Academy of Sciences* **105**, 9994-9998 (2008).
5. Lu, T.K. & Collins, J.J., *Proceedings of the National Academy of Sciences* **106**, 4629-4634 (2009).
6. Canton, B., Labno, A. & Endy, D., *Nat Biotech* **26**, 787-793 (2008).
7. Purnick, P.E.M. & Weiss, R., *Nat Rev Mol Cell Biol* **10**, 410-422 (2009).
8. Ellis, T., Wang, X. & Collins, J.J., *Nat Biotech* **27**, 465-471 (2009).
9. Rackham, O. & Chin, J.W., *Biochem Soc Trans* **34**, 328-9 (2006).
10. Rackham, O. & Chin, J.W., *Nat Chem Biol* **1**, 159-166 (2005).
11. An, W. & Chin, J.W., *Proceedings of the National Academy of Sciences* **106**, 8477-8482 (2009).
12. Filipovska, A. & Rackham, O., *ACS Chemical Biology* **3**, 51-63 (2008).

CO-DESIGN IN SYNTHETIC BIOLOGY: A SYSTEM-LEVEL ANALYSIS OF THE DEVELOPMENT OF AN ENVIRONMENTAL SENSING DEVICE

DAVID A. BALL[1*], MATTHEW W. LUX[1*], RUSSELL R. GRAEF[2], MATTHEW W. PETERSON[2], JANE D. VALENTI[1], JOHN DILEO[2], JEAN PECCOUD[1†]

[1]*Virginia Bioinformatics Institute, Virginia Polytechnic Institute and State University, Blacksburg, VA (USA)*
[2]*Emerging Technologies Office, The MITRE Corporation, McLean, VA (USA)*

The concept of co-design is common in engineering, where it is necessary, for example, to determine the optimal partitioning between hardware and software of the implementation of a system features. Here we propose to adapt co-design methodologies for synthetic biology. As a test case, we have designed an environmental sensing device that detects the presence of three chemicals, and returns an output only if at least two of the three chemicals are present. We show that the logical operations can be implemented in three different design domains: (1) the transcriptional domain using synthetically designed hybrid promoters, (2) the protein domain using bi-molecular fluorescence complementation, and (3) the fluorescence domain using spectral unmixing and relying on electronic processing. We discuss how these heterogeneous design strategies could be formalized to develop co-design algorithms capable of identifying optimal designs meeting user specifications.

1. Introduction

1.1. *The need for co-design of synthetic biology systems*

A major focus in the field of synthetic biology from the field's inception has been the design of biological "devices" [1]. To date, many biological analogs of common electronics parts have been developed. Some notable biological devices analogous to those commonly used for electronics system design include logic gates [2-4], asynchronous logic components [5], switches [6-8], oscillators [8-12], memory circuits [13] and most recently genetic counters [14]. While these devices have been excellent proofs-of-concept for the application of traditional engineering design to the design of biological systems, little progress has been made in the design of larger, more complex, biological constructs [15]. This was seen in 2004, when Blue Heron Biotechnology did not receive a single submission to their Big DNA Contest, which offered free synthesis of the most "interesting" DNA construct of over 40kb in length (http://tinyurl.com/bigdna).

The designs of early artificial gene networks were restricted to a single design domain: protein-DNA interactions regulating transcription [6,10,16]; intra-molecular interactions within RNA molecules [3,17-20], or even interaction between DNA molecules [21,22]. More recent publications however, report the combined use of multiple domains in a single design. The genetic counter [14] is a good example of a heterogeneous design combining circuitry operating in different domains. Two separate methodologies were used, resulting in counter devices that, while performing the same type of logic (being able to count to either two or three), were appropriate for different uses. The riboregulated transcriptional cascade (RTC) counter utilized a fast transcriptional cascade for counting. The DNA invertase cascade (DIC) counter used recombinases upstream of inverted promoters to count. Due to the dynamics of DNA recombination, these counters activate more slowly, and as such can be used only for the counting of low-frequency events. It is anticipated that future heterogeneous designs will also include non-biological elements, as well as biological circuits. One such example of this is an AND gate used to regulate protein folding, which utilized UV light and ATP as stimuli [23]. By including non-biological elements, the design space is increased, allowing for the design of more complex, and potentially better performing, constructs. In the AND gate example above, the wavelength and intensity of the UV light is an additional parameter that can be used to optimize the system. Similarly, one approach described here uses advanced detection to expand the list of viable reporters.

As synthetic biology matures, it becomes necessary to develop more sophisticated design strategies. The design of an industrial application calls for a systematic comparison of different possible designs meeting the user specification in order to identify an optimal design maximizing one or several figures of merit. Besides the design correctness (the design does what it is supposed to do), its performance, development cost, manufacturing cost, or reconfigurability are other criteria that an engineering team may need to optimize. When developing electronic systems, an important design decision is the partitioning of features implemented in hardware and those implemented in software, which is known as the hardware/software co-design problem [24,25]. Software ensures

† Corresponding author: peccoud@vt.edu
* These authors contributed equally to this work.

rapid time to market, design flexibility, and runs on inexpensive processors produced in large volumes. However, the development of application-specific circuits is needed when software running on generic processors cannot meet the required performance. The hardware/software co-design problem illustrates the issues arising when heterogeneous technologies are combined into a system. At a very high-level, the design of synthetic biology applications includes a wetware component (the living organisms), but also a hardware component represented by the instrument and software used to acquire and process signals generated by the biological component of the system. At a higher resolution, the wetware component is itself heterogeneous since the transcriptional, translational, and proteomic components of this machinery represent different design domains.. By leveraging methods acquired by electrical engineers to develop heterogeneous systems, the complexity, efficiency, and flexibility of synthetic biology applications will likely be dramatically increased, while reducing the production costs of these systems.

1.2. *An environmental sensor as test case for co-design analysis*

In the field of biosecurity there is an increasing recognition of the need for systems that can rapidly detect pollutants, contaminants, and biothreat agents (pathogenic bacteria, viruses and toxins) in food, agricultural products, pharmaceuticals, and environmental samples. For example, the safety of the US food supply is an ongoing concern because of potential impacts of contamination on both public health and the US economy. In addition to inadvertent contamination, concerns about a bioterrorism event such as the intentional introduction of pathogens and/or toxins, brings a new dimension to this problem.

Cells possess innate abilities that make them ideal for environmental sensing applications. Specifically, they are inherently able to detect small concentrations (parts per billion) of chemicals (or combinations of chemicals) in their environment and respond to it, usually with an amplified signal. Cells can be programmed by identifying three functional layers: an input layer, an information processing layer, and an output layer [26]. This abstract representation of the environmental sensing chain can help design environmental sensing devices relying on biological systems for transforming chemical information into electrical signals that can be recorded and processed by computer systems. For instance, a situational awareness monitoring system will rely on a network of geographically dispersed sensing units communicating chemical data to a central server or to personnel operating in their vicinity. In this scenario, the sensing units should be capable of some basic processing of chemical data in order to communicate informative data.

In many cases, the presence of individual molecules in the environment is not informative while the simultaneous presence of two molecules can provide valuable information worthy communicating. In the field of defense, many chemical agents and explosives are made from combinations of commonly available industrial chemicals. For example, mustard gas can be created with thiodiglycol, an industrial solvent used in dyes and other applications, and phosphorus trichloride, a common industrial chemical used to manufacture a wide range of organic phosphorous compounds. Rapid field detection of such combinations could be an important application of sensing devices [27]. In ecology, it is well established that levels of heavy metals can be below the safe threshold individually, but combine to be lethal in fish [28]. In human health, it was recently shown that the common herbicide Roundup is more toxic in the presence of its supposedly inert adjuvants, and has negative effects in pregnant women even at “safe” levels of the active ingredient [29]. These findings may lead to the elucidation of further chemicals that are safe alone, but unsafe together. Detection of such combinations could become important to assess environmental or security threats.

Table 1. Logic table of the environmental sensor

Input 1	Input 2	Input 3	Output 1	Output 2	Output 3
-	-	-	-	-	-
-	-	+	-	-	-
-	+	-	-	-	-
-	+	+	-	+	-
+	-	-	-	-	-
+	-	+	-	-	+
+	+	-	+	-	-
+	+	+	+	+	+

The following sections describe three possible methods for implementing a cell based system designed to be able to detect the presence of each pair of three different chemical inputs and produce three different electronic outputs in response. Formally, the system can be specified by a truth table (Table 1). These designs differ in where the logic is implemented in the system (Figure 1). In the first option, hybrid promoters that contain binding sites for transcription factors responsive to the inputs are used to control the expression of fluorescent proteins. Only when the proper inputs are present will reporter genes be expressed, thus the logic occurs at the transcriptional level. The second option is to implement the logic at the protein level. This is accomplished by coupling each input to the expression of a non-fluorescent fragment of a fluorescent protein. Only when the two proper inputs are present will the fragments associate to generate a fluorescent signal. A final option is to embed the logic in the electronic layer. In this case each input directly activates the expression of one of three different fluorescent proteins and the inputs present are determined by processing the pattern of fluorescence that is obtained. For all three scenarios, the fluorescent proteins used were cyan (CFP), yellow (YFP), and red (RFP), which are all easily separated from each other.

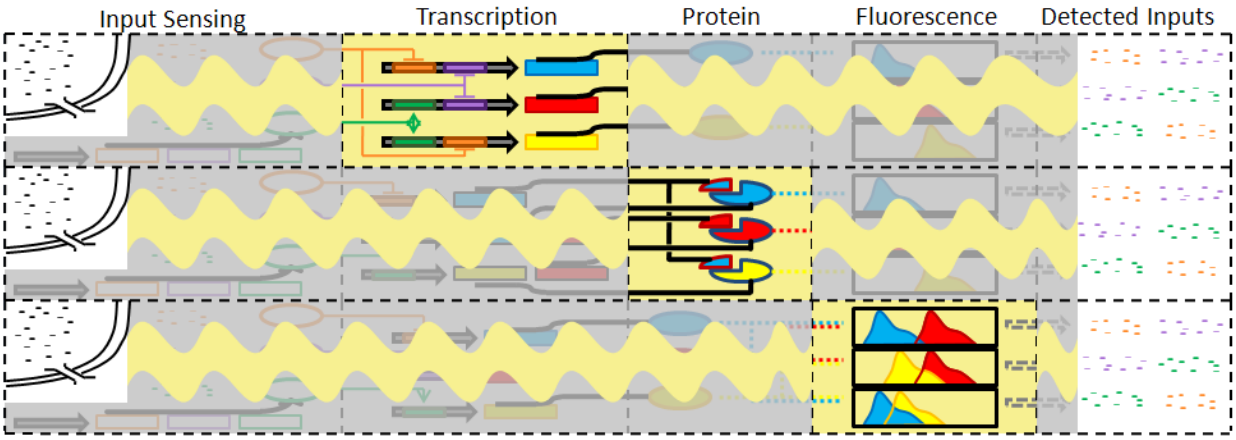


Figure 1. Implementation of logic in different design domains. The figure gives an overview of how each approach processes the environmental inputs. Wavy yellow lines indicate signal transduction, and yellow boxes highlight where the logic occurs. The details of each design are presented in Figure 2, Figure 3, and Figure 4

2. Solution 1: Hybrid Promoters

2.1. *Theoretical foundation*

One approach is to embed the design logic into transcriptional control with hybrid promoters. Since the input to the system is a set of small molecules, the first step is to sense the presence of the small molecule in the environment. In the described situation, these small molecules are ligands capable of binding to specific transcription factors. The sensor function is therefore accomplished by constitutive expression of the corresponding transcription factors. This sensing mechanism is conserved through each approach. Once bound by its ligand, the behavior of each transcription factor is altered. For example, some ligand-bound repressors can no longer bind to their corresponding promoters and exert their repressive properties. Thus, each transcription factor can be thought to switch on or off depending on the presence or absence of ligand.

In order to implement the design logic, the on/off state of these transcription factors must be processed. The hybrid promoter approach accomplishes the logic by controlling expression of a reporter gene through a promoter that responds to the state of pairs of transcription factors. That is, promoter A responds to the on/off state of transcription factors 1 and 2, promoter B to transcription factors 2 and 3, and promoter C to transcription factors 1 and 3. Each promoter responds only when both transcription factors have been toggled by the presence of the ligand molecule. The name “hybrid promoters” derives from the fact that they are engineered to respond to multiple transcription factors.

As a more detailed explanation, let us assume that inputs 1 and 2 are repressor proteins that repress only in the absence of their respective ligand. With no ligand present, both transcription factors effectively repress transcription of the reporter gene controlled by promoter A. In the presence of ligand 1 only, transcription factor 1 loses its ability to bind to promoter A, but transcription is still blocked by transcription factor 2. Likewise, if only ligand 2 is

present, transcription factor 1 continues to block transcription of the reporter gene. However, in the presence of both ligand 1 and ligand 2, neither transcription factor can bind to promoter A, and the reporter gene is freely expressed.

The transcription factor does not have to be an inducible repressor to accomplish the appropriate logic. There are 4 possible transcription factor responses: (1) the transcription factor *represses* only in the *absence* of its ligand, (2) the transcription factor *represses* only in the *presence* of its ligand, (3) the transcription factor *activates* only in the *absence* of its ligand, and (4) the transcription factor *activates* only in the *presence* of its ligand. Only options (1) and (4) are viable for implementing the design logic because they do not invert the signal. For example, a positive signal from the presence of a ligand would become a negative signal if the induced transcription factor transitioned from an inactive repressor (therefore allowing transcription) to an active repressor (therefore blocking transcription). Put another way, the positive (+) input signal would be inverted by *activation of the repressor* (-) to block expression from the promoter (+/- = -). On the other hand, the positive (+) input signal would be conserved by *inactivation of the repressor* (+) to allow expression from the promoter (+/+ = +), which preserves the signal.

Figure 2 illustrates an implementation of the logic using specific transcription factors and ligands. The rationale behind the choice of specific parts is described below.

2.2. *Proposed design*

A number of features need to be considered in the selection of appropriate inducible transcription factors. These features include: high range of control, compatibility with other transcription factors, and prior use in other applications. High range of control is necessary to ensure that the final signal is detected over the cellular noise. Compatibility with other transcription factors refers to the design of the actual promoter. For example, activators frequently must bind to specific promoter regions and thus may prevent the use of a second transcription factor that must bind an overlapping site. Last, well characterized transcription factors that have been widely used in other designs are preferred.

In synthetic biology, the list of commonly used genes is small and thus selection of appropriate parts is restricted. The first two appropriate transcription factors that match the criteria are LacI, which is inducible by isopropyl β-D-1-thiogalactopyranoside (IPTG), and TetR which is inducible by anhydrotetracycline (aTc). Both have operator sites that can be effectively placed in multiple locations to prevent interference with other transcription factors and both have been shown to have a high range of control [30,31]. Furthermore, hybrid promoters under the simultaneous control of LacI and TetR were described previously [31] which matched our logic criteria.

The selection of the third transcription factor is not as straightforward. LuxR is an attractive candidate. While LacI and TetR induce transcription by derepression in the presence of their respective ligand, LuxR activates transcription in the presence of its ligand acyl-homoserine lactone (AHL). LuxR has a high range of control for the wild type promoter [32]; it is described extensively in the literature, and it is commonly used in synthetic biology applications [26,33-35]. Although previous attempts to design hybrid promoter responding to LuxR/LacI or LuxR/TetR proved unsuccessful [31], a careful investigation of these LuxR hybrid promoters shows that none of them had spacing between the -10 box and the LuxR binding site that was identical to the wild type promoter. Given that this spacing has been shown to be important to ensure proper regulation of gene expression [32], we predict LuxR hybrid promoters can be redesigned if the suitable spacing is used.

The first promoter for implementing the environmental sensing device uses the sequence of the promoter A90 responding to LacI/TetR [31]. For the promoters responding to LuxR, we modified the wild type promoter for *luxI* and added *lac* and *tet* operators, respectively, downstream of the -10 box. Specifically, we replaced the sequence downstream of the -10 sequence in the wild type promoter with sequence downstream of the -10 box taken from promoters successfully responding to LacI and TetR [31]. Since the designed promoters already contain sufficient spacing downstream of +1, the 3 full sequences were assembled by simply adding a ribosome binding site sequence, a coding sequence for three different fluorescent proteins (CFP, YFP, RFP), and a terminator. Figure 1 shows the specific combinations of hybrid promoter and fluorescent reporter gene.

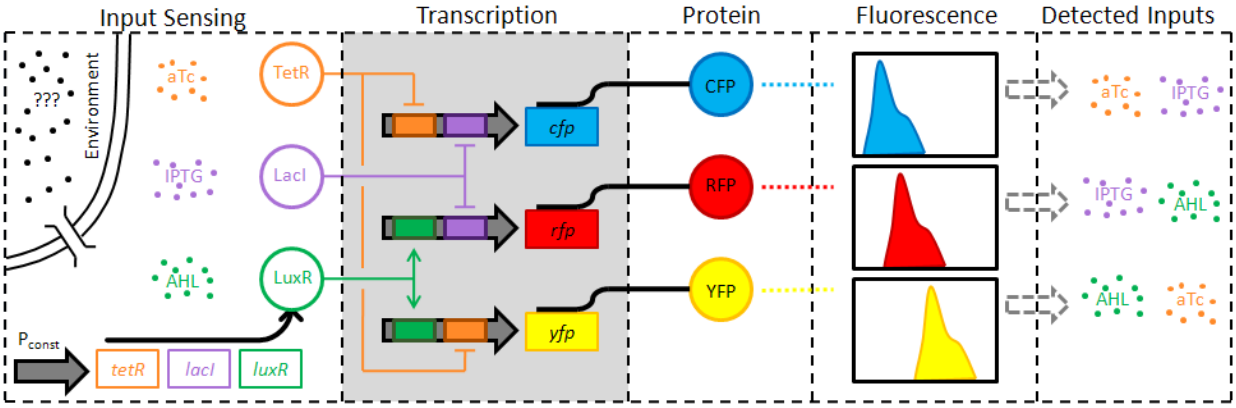


Figure 2. Hybrid Promoter Approach. The system logic is implemented in the control of transcription of 3 reporter genes. See text for detailed explanation of how the logic is processed. Orange indicates the aTc signal, purple the IPTG signal, green the AHL signal, blue the CFP signal, red the RFP signal, and yellow the YFP signal. Dots indicate small molecule inducers, thick solid arrows indicate promoters with color-coded operator boxes inside and adjacent boxes indicating genes, thick black lines indicate production of protein from a gene, circles indicate proteins, dotted lines indicate fluorescence measurement, and the gray box indicates where the logic occurs.

3. Solution 2: Fluorescence complementation

3.1. *Theoretical foundation*

While transcriptional logic is prevalent in the majority of synthetic biology constructs, logic can also be performed in the protein domain. As previously reported, the anti-parallel Leucine zipper mediated direction of protein reassembly allows for the reconstitution of intact and functional GFP [36]. Further research has shown the ability to adapt protein reassembly of fluorescent proteins to visualize protein-protein interactions *in vivo* [37,38]. These methods describe "Bimolecular Fluorescence Complementation" (BiFC) which uses non-fluorescent fragments of fluorescent proteins bound to separate functional proteins; when interaction between these proteins takes place the non-fluorescent pairs combine to produce a fluorescent complex. In this second solution, inputs are sensed in the same manner as in Solution 1. However, here the promoters respond to single transcription factors and therefore a single environmental input. Hence, they simply transmit the signal to the protein domain. The information is processed by coupling non-fluorescent halves of fluorescent proteins. This makes an "AND" gate from the pairing of non-fluorescent halves to produce a final product of fluorescence dependent on the promoters engaged. The logic comes not only from which fragments are produced but from the fact that only certain combinations of fragments will produce a detectable fluorescence output. Fragment 1 (N-terminal) of GFP combines with fragment 2 (C-terminal) of GFP to form functional GFP, in contrast fragment 1 (N-terminal) of GFP cannot combine with fragment 1 (N-terminal) of CFP.

3.2. *Design*

In order to implement this method of logic processing, several factors must be considered. Dissection sites of CFP, YFP, and a monomeric form of DsRed, a Red Fluorescent Protein variant that yield two non-fluorescent halves capable of reassembly into fluorescent proteins have been previously reported [37]. CFP is split into two fragments at amino acid 155 yielding a CFP 1-155 (N-terminal) fragment and a CFP 155-239 (C-terminal) fragment, subsequently referred to as CFP-N and CFP-C respectively. Similarly DsRed and YFP are split into the following fragments: RFP-N (residues 1-168), RFP-C (residues 169-225), and YFP-N (residues 1-154). The C-terminal of YFP is not required, because YFP-N can combine with CFP-C to form a species that produces yellow fluorescence [37,38].

These fragments will be cloned downstream of three inducible promoters. Fragments CFP-N and RFP-C are placed under the control of LacI (inducible by IPTG), Fragments YFP-N and RFP-N are placed under the control of LuxR (inducible by AHL), and the final fragment CFP-C is placed under TetR (inducible by aTc) control. As shown in Figure 3, the expected outputs from this system are dependent on which fragments are expressed. Some factors which play a role in total fluorescence are: the concentration of the inducer, the relative strength of promoter, and the fragment complementation.

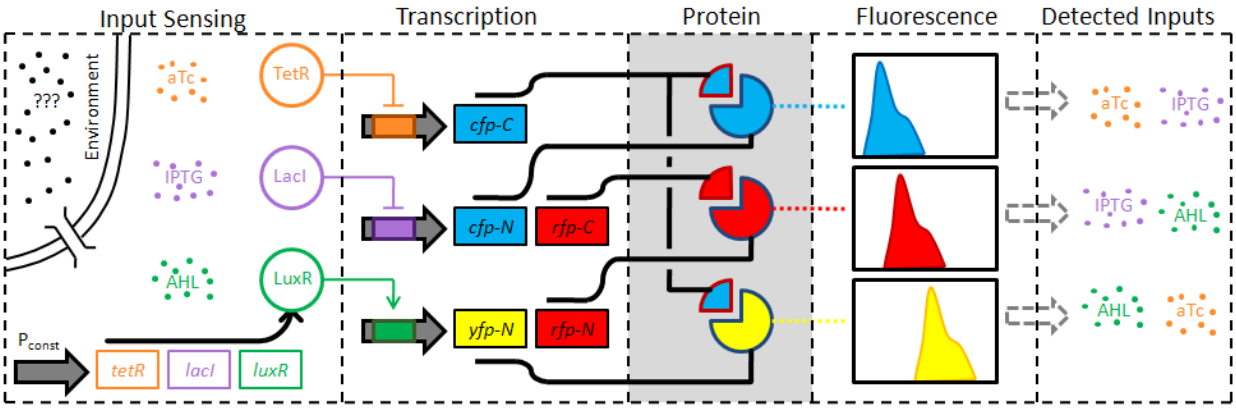


Figure 3. Fluorescence Complementation Approach. The system logic is implemented in the complementation of 3 fluorescent proteins. See text for detailed explanation of how the logic is processed. Orange indicates the aTc signal, purple the IPTG signal, green the AHL signal, blue the CFP signal, red the RFP signal, and yellow the YFP signal. Dots indicate small molecule inducers, thick solid arrows indicate promoters with color-coded operator boxes inside and adjacent boxes indicating genes, thick black lines indicate production of protein from a gene, circles indicate proteins, dotted lines indicate fluorescence measurement, and the gray box indicates where the logic occurs.

Logic circuit processing may be carried out by protein reassembly mediated by inducible promoters. For future experimentation, synthetically designed proteins for binding novel small and macro molecules could be incorporated into the system outlined here for logical processing.

Table 2. Previously reported fluorescent outputs generated by complementation of the different non-fluorescent protein fragments resulting in stable fluorescent protein complexes (given in color and name) or unstable/incompatible fragments (given by N/C "no color") [37].

Fragment	CFP-N	CFP-C	RFP-N	RFP-C	YFP-N
CFP-N	N/C	BLUE	N/C	N/C	N/C
CFP-C	BLUE	N/C	N/C	N/C	YELLOW
RFP-N	N/C	N/C	N/C	RED	N/C
RFP-C	N/C	N/C	RED	N/C	N/C
YFP-N	N/C	YELLOW	N/C	N/C	N/C

4. Solution 3: Unmixing of fluorescence spectra

4.1. *Theoretical foundation*

For the simultaneous detection of 3 (or fewer) fluorescent proteins it is possible to find fluorescent proteins with suitably separated excitation and emission spectra, such that the 3 colors can be distinguished by the use of optical band-pass filters [39]. However, this becomes difficult, if not impossible, as the number of fluorescent proteins increases. Therefore, a spectral unmixing approach was developed for the detection of multiple fluorescent signals [40-42]. The use of spectral unmixing also allows the use of a wider range of fluorescent proteins, which is of great value when it is necessary to use fluorophores with similar properties such as maturation and degradation times.

Spectral unmixing relies on the *a priori* collected emission spectra of the individual fluorescent proteins in the system to determine which fluorophores have contributed to the observed signal. The experimental output spectrum, F, can be described by the system of linear equations:

$$F = \mathbf{X}A, \quad (1)$$

where the m data points in the output spectrum, F, and the weights of the n individual fluorophores, A are column vectors and $\mathbf{X_{ij}}$ is the **i**th point in the spectrum of fluorophore **j**,

$$F = \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_m \end{pmatrix}, \quad A = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_n \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m1} & X_{m2} & \cdots & X_{mn} \end{pmatrix}. \quad (2)$$

If $m > n$, Eq. 1 describes an overdetermined system, which can be solved for A by a least squares fitting algorithm to minimize $F - \mathbf{X}A$.

4.2. *Design*

Biologically, spectral unmixing offers the simplest approach for implementing the logic required to detect combinations of environmental species, as it does not require any interactions between the various promoters or protein products. As illustrated in Figure 4, each input ligand triggers the production of a single fluorescent protein. The sensing mechanism is again the same as Solution 1. In this case, the input signal is transmitted all the way to the spectral detection via single-operator promoters controlling production of single fluorescent proteins. The various fluorescence components can be extracted by use of Eq, (1), and the concentrations of each chemical can be determined.

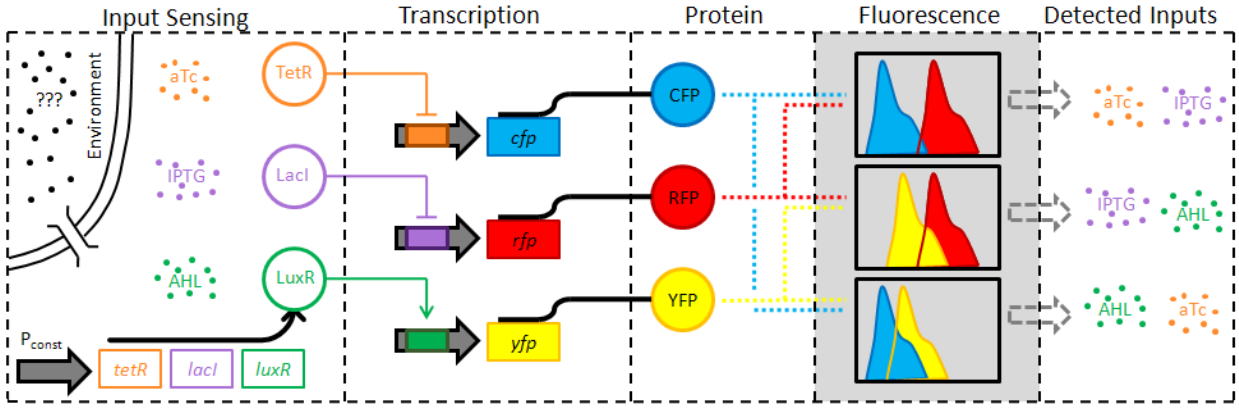


Figure 4. Spectral Detection Approach. The system logic is implemented in the unmixing of the measured spectra. See text for detailed explanation of how the logic is processed. Orange indicates the aTc signal, purple the IPTG signal, green the AHL signal, blue the CFP signal, red the RFP signal, and yellow the YFP signal. Dots indicate small molecule inducers, thick solid arrows indicate promoters with color-coded operator boxes inside and adjacent boxes indicating genes, thick black lines indicate production of protein from a gene, circles indicate proteins, dotted lines indicate fluorescence measurement, and the gray box indicates where the logic occurs.

4.3. *Preliminary data*

A spectrofluorimeter is required to collect the needed data to unmix the contributions of multiple fluorescent proteins. For the experiments reported here, a NanoDrop 3300 (Thermo Scientific, Wilmington DE) was used. The NanoDrop System uses a small sample volume of 2 µL. It contains three light-emitting diodes (LEDs) for fluorescence excitation: UV (peak emission at 365 nm), blue (470 nm) and white (460-650 nm). The collected fluorescence is dispersed over a 1024-pixel linear CCD, and allows the collection of wavelengths from 400-750 nm with a 4 nm resolution. Because the system does not include any optical bandpass filters, it is necessary to obtain spectra for blank samples, cells similar to those being used in the experiment but lacking any fluorophores.

Preliminary experiments were performed with four constitutively expressed fluorescent proteins: EGFP [43], acGFP [44], vYFP [45], and Citrine [46], each in a separate cell-line, to test the ability of the spectral unmixing algorithm to separate fluorescent proteins with similar emission spectra. Figure 5 presents the reference spectra for these four fluorophores. The spectra show that this combination of fluorescent proteins can be regarded as a worst-case scenario, as none of the four could be distinguished with the use of optical band-pass filters.

To account for differences in expression levels of the four different fluorescent proteins, the fluorescence spectra of the four cell-lines were normalized by their optical absorbance at 600 nm. The normalized spectra were

then used as the inputs to the unmixing algorithm, so that the extracted coefficients, such as those recorded in Table 3, are also in units of absorbance.
The four cell cultures were then mixed at known concentrations, and the resulting spectra shown in Figure 5b-f were analyzed with the spectral unmixing algorithm in order to extract the components of the mixtures. The measured optical densities at 600 nm for the existing cells in each mixture along with the extracted values are given Table 3. In all cases, there were no false-negatives. There are some false positive results, however, but their coefficients remained small (value less than 0.006).

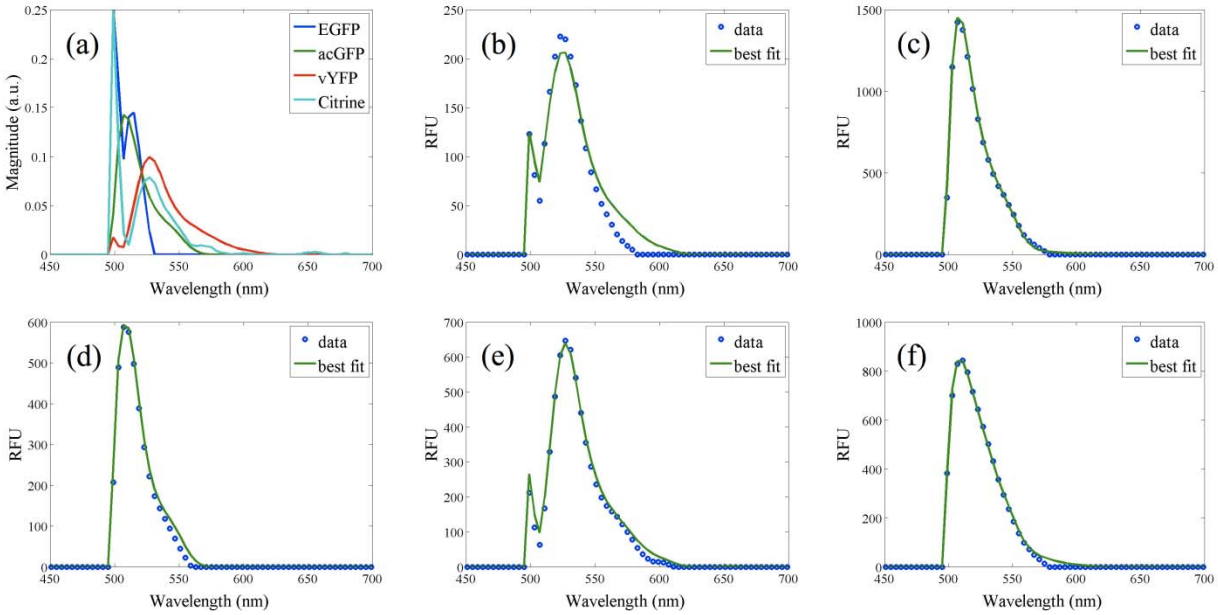


Figure 5: Experimentally measured emission spectra. (a) Emission profiles of cell cultures expressing 1 of the 4 fluorescent proteins. (b-f) Collected fluorescence spectra for mixtures of cell cultures expressing (b) EGFP and vYFP, (c) acGFP and Citrine, (d) acGFP and EGFP, (e) Citrine and vYFP, and (f) acGFP, Citrine, EGFP, and vYFP.

There are several possible explanations for the observed discrepancies between the measured and extracted optical densities of the true positives listed in Table 3. First, the small sample size used (2 μL) could mean that the cell concentrations in the measured sample did not completely reflect the concentrations of the stock solution. Also, as can be seen in Figure 5a, there is a sharp peak visible in the fluorescence spectrum of all of the fluorophores except acGFP at ~500 nm. This peak appears to be part of the spectrum of the blue LED used for exciting all of the fluorescent proteins, and most likely contributes to some of the errors in the extracted contributions.

Table 3: Measured and Extracted optical densities of mixtures of cells expressing a single fluorescent protein. Gray boxes indicate measured values of 0.

OD_{600}	Mix1		Mix 2		Mix 3		Mix 4		Mix 5	
	Meas.	Fit	Meas.	Fit	Meas.	Fit	Meas.	Fit	Meas.	Fit
acGFP	0	0.003	0.050	0.099	0.046	0.040	0	0.004	0.030	0.062
Citrine	0	0.006	0.070	0.039	0	0.002	0.081	0.014	0.027	0.017
EGFP	0.053	0.040	0	0	0.048	0.060	0	0	0.032	0.011
vYFP	0.048	0.029	0	0.006	0	0	0.082	0.100	0.027	0.037

5. Discussion

5.1. *Heterogeneous solutions to a synthetic biology design problem*

We have proposed three distinct solutions to a specification, namely the detection of distinct combinations of chemical signals. The first proposed solution uses a set of three hybrid promoters, performing the logic at the transcriptional level. This transcriptional solution would quickly become intractable if the number of input chemical

signals is increased. While we have shown a potential solution to the problem for three input signals, the number of parts needed to detect combinations from four possible inputs would require more parts than are currently available.

The second proposed solution still uses biology for the entirety of the logic, but does so in the protein domain. By moving the logic into the protein domain, the genetic circuitry is simplified. This allows the circuit to more easily be combined into a larger system, since the potential for cross-talk is decreased. Even though strategies to split fluorescent proteins still remain to be explored in a more systematic way, there are likely a limited number of fluorescent proteins that can be used in this context. So, the issue of the limited number of parts available faced in the transcriptional domain is also relevant in the protein domain. The assembly of the fluorescent proteins will increase the activation time of the system which can be an advantage or an inconvenience depending on the specific application of the environmental sensing device. In some cases, a fast response will be needed. In other cases, a longer maturation time can be used to time average the device response.

Finally, a third solution relies on the unmixing of fluorescence spectra for identification of molecules. In this case, biology is being used to generate the output signals, but the logic is being done outside of the biological domain. By implementing the logic outside of the biological system, the number of molecules possible to distinguish between is greatly increased, limited only by the number of transduction mechanisms and reporters. Also, the use of simpler biological circuits in this implementation circumvents any possible difficulties that may arise from incompatibilities between biological parts in the other two design schemes.

As we proceed with this project by physically implementing and characterizing each of these approaches, we will meticulously observe the differences in the difficulty of implementation and the performance of each design. As a result of unbalanced component behavior, each approach is likely to be improved by tweaking such elements as promoter strength, translational efficiency, degradation, etc. Thus, the cost and benefit of iterative designs will be evaluated as well. By further considering the difficulty of the design process for each solution, we will be able to holistically compare the strengths and weaknesses of the different approaches. Some comparative measures are discussed below. At this stage, it is important for the design team to acknowledge that there are multiple solutions to design problems and that the solutions can be implemented in different design domains. Just like the design of electronic systems is often a heterogeneous combination of hardware and software solutions, the design of a synthetic biology device can include multiple domains for the wetware component of the design as well as the hardware and software used to integrate the information originating from the design wetware component into a larger system.

A possibility that has not been considered in this paper is the combination of solutions implemented in different design domains. Are there solutions that could combine hybrid promoters and fluorescence complementation? Or could spectral unmixing be combined with fluorescence complementation to achieve better performance? Even though this manuscript proposes three distinct solutions, the universe of possible solutions is large and difficult to explore manually.

5.2. *Enabling co-design of synthetic biology application by design automation*

In order to compare different solutions to a design problem, it is necessary to define various figures of merits that can be used to quantitatively compare different solutions. Sensitivity, dynamic range, response time, robustness, or noise can be used to characterize the design performance. The development cost could be estimated by a function of the number of previously characterized components that can be reused in a new design. For instance, a solution requiring the development of a new promoter is expected to be slower and more expensive to implement than a solution relying on well characterized genetic parts such as the fluorescence unmixing approach. The manufacturing or production cost may also be a factor. The development of Solution 3 is the simplest but it relies on a more refined optical components that would increase the size and manufacturing cost of the device. This option may not be practical if millions of sensing devices needed to be distributed over large geographic regions to detect facilities manufacturing chemical weapons. Each specific application will require optimizing these metrics using multi-objective optimization algorithms [47,48].

Formalizing the representation of the design space is necessary to automate its exploration while searching for optimal designs. Fortunately, the wetware component of the system can be represented by the sequence of the synthetic DNA molecule implementing the design. Our group recently proposed to use formal languages to represent the structure of synthetic DNA sequences [49]. More recently, this original syntactic model was augmented with a semantic model used to predict the behavior encoded in a DNA sequence. By implementing this formalism in a logic programming language like Prolog [50], we were able to systematically explore a design space by generating structurally correct DNA sequences, compiling them into SBML files describing their behavior, and

simulating these files to identify solutions meeting a set of specification. Defining a distance in the design space, would make it possible to use optimization algorithms instead of a systematic exploration of all possible designs. In addition, it would be necessary to represent the non-DNA part of the designs by augmenting the language to represent detection systems and inputs.

The field of Synthetic Biology is growing by systematically adapting engineering practices to the design of biologically-inspired systems. The development of practical synthetic biology devices will require a system-level analysis and a co-design approach that have yet to be explored. In this paper, we have shown that the design space of a real-world device is large and may combine components developed in heterogeneous design domains. Finding optimal designs will require the use of design automation tools [51] like GenoCAD [52]. By adapting co-design methods used in more mature engineering fields [25,53,54], synthetic biology will fulfill its promise in the form of large, "interesting" circuits that were called for in the Big DNA contest .

Acknowledgments

This work was supported by the National Institutes of Health Grant 1 R01 GM078989, the National Science Foundation grant EF-0850100, the MITRE Innovation Program, and a generous donation from Science Applications International Corporation (SAIC) to the Virginia Bioinformatics Institute. We are indebted to James J. Valdes for indicating the possible application of this environmental sensor to the enforcement of the Chemical Weapons Convention and other nonproliferation treaties.

References

1. Endy D (2005) Foundations for engineering biology. Nature 438: 449-453.
2. Seelig G, Soloveichik D, Zhang DY, Winfree E (2006) Enzyme-free nucleic acid logic circuits. Science 314: 1585-1588.
3. Win MN, Smolke CD (2008) Higher-order cellular information processing with synthetic RNA devices. Science 322: 456-460.
4. Buchler NE, Gerland U, Hwa T (2003) On schemes of combinatorial transcription logic. Proc Natl Acad Sci U S A 100: 5136-5141.
5. Nguyen NPD, Kuwahara H, Myers CJ, Keener JP (2007) The design of a genetic muller C-element. ASYNC 2007: 13th IEEE International Symposium on Asynchronous Circuits and Systems: 95-104.
6. Gardner TS, Cantor CR, Collins JJ (2000) Construction of a genetic toggle switch in Escherichia coli. Nature 403: 339-342.
7. Hasty J, Pradines J, Dolnik M, Collins JJ (2000) Noise-based switches and amplifiers for gene expression. Proceedings of the National Academy of Sciences of the United States of America 97: 2075-2080.
8. Atkinson MR, Savageau MA, Myers JT, Ninfa AJ (2003) Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in Escherichia coli. Cell 113: 597-607.
9. Tigges M, Marquez-Lago TT, Stelling J, Fussenegger M (2009) A tunable synthetic mammalian oscillator. Nature 457: 309-312.
10. Elowitz MB, Leibler S (2000) A synthetic oscillatory network of transcriptional regulators. Nature 403: 335-338.
11. Fung E, Wong WW, Suen JK, Bulter T, Lee SG, et al. (2005) A synthetic gene-metabolic oscillator. Nature 435: 118-122.
12. Chilov D, Fussenegger M (2004) Toward construction of a self-sustained clock-like expression system based on the mammalian circadian clock. Biotechnology And Bioengineering 87: 234-242.
13. Ajo-Franklin CM, Drubin DA, Eskin JA, Gee EP, Landgraf D, et al. (2007) Rational design of memory in eukaryotic cells. Genes Dev 21: 2271-2276.
14. Friedland AE, Lu TK, Wang X, Shi D, Church G, et al. (2009) Synthetic Gene Networks That Count. Science 324: 1199-1202.
15. Goler JA, Bramlett BW, Peccoud J (2008) Genetic design: rising above the sequence. Trends Biotechnol 26: 538-544.
16. Guet CC, Elowitz MB, Hsing W, Leibler S (2002) Combinatorial synthesis of genetic networks. Science 296: 1466-1470.
17. Bayer TS, Smolke CD (2005) Programmable ligand-controlled riboregulators of eukaryotic gene expression. Nature Biotechnology 23: 337-343.

18. Win MN, Smolke CD (2007) A modular and extensible RNA-based gene-regulatory platform for engineering cellular function. Proc Natl Acad Sci U S A 104: 14283-14288.
19. Isaacs FJ, Dwyer DJ, Collins JJ (2006) RNA synthetic biology. NatBiotechnol 24: 545-554.
20. Isaacs FJ, Collins JJ (2005) Plug-and-play with RNA. Nature Biotechnology 23: 306-307.
21. Kim J, White KS, Winfree E (2006) Construction of an in vitro bistable circuit from synthetic transcriptional switches. Molecular Systems Biology: -.
22. Okamoto A, Tanaka K, Saito I (2004) DNA logic gates. Journal of the American Chemical Society 126: 9458-9463.
23. Muramatsu S, Kinbara K, Taguchi H, Ishii N, Aida T (2006) Semibiological molecular machine with an implemented "AND" logic gate for regulation of protein folding. J Am Chem Soc 128: 3764-3769.
24. DeMicheli G, Gupta RK (1997) Hardware/software co-design. Proceedings of the Ieee 85: 349-365.
25. Wolf WH (1994) Hardware-software co-design of embedded systems. Proceedings of the IEEE 82: 967-989.
26. Kobayashi H, Kaern M, Araki M, Chung K, Gardner TS, et al. (2004) Programmable cells: interfacing natural and engineered gene networks. Proceedings of the National Academy of Sciences of the United States of America 101: 8414-8419.
27. Institute of Medicine (U.S.). Committee to Survey the Health Effects of Mustard Gas and Lewisite., Pechura CM, Rall DP (1993) Veterans at Risk : the health effects of mustard gas and Lewisite. Washington, D.C.: National Academy Press. xviii, 427 p. p.
28. Witeska M, Jezierska B. The effects of environmental factors on metal toxicity to fish; 2002 Oct 14-16; Brno, Czech Republic. Parlar Scientific Publications (P S P). pp. 824-829.
29. Richard S, Moslemi S, Sipahutar H, Benachour N, Seralini GE (2005) Differential effects of glyphosate and roundup on human placental cells and aromatase. Environ Health Perspect 113: 716-720.
30. Lutz R, Bujard H (1997) Independent and tight regulation of transcriptional units in Escherichia coli via the LacR/O, the TetR/O and AraC/I1-I2 regulatory elements. Nucleic Acids Res 25: 1203-1210.
31. Cox RS, 3rd, Surette MG, Elowitz MB (2007) Programming gene expression with combinatorial promoters. Mol Syst Biol 3: 145.
32. Egland KA, Greenberg EP (1999) Quorum sensing in Vibrio fischeri: elements of the luxl promoter. Molecular Microbiology 31: 1197-1204.
33. Basu S, Gerchman Y, Collins CH, Arnold FH, Weiss R (2005) A synthetic multicellular system for programmed pattern formation. Nature 434: 1130-1134.
34. Karig D, Weiss R (2005) Signal-amplifying genetic enables in vivo observation circuit of weak promoter activation in the RhI quorum sensing system. Biotechnology And Bioengineering 89: 709-718.
35. Basu S, Mehreja R, Thiberge S, Chen MT, Weiss R (2004) Spatiotemporal control of gene expression with pulse-generating networks. Proceedings of the National Academy of Sciences of the United States of America 101: 6355-6360.
36. Ghosh I, Hamilton AD, Regan L (2000) Antiparallel leucine zipper-directed protein reassembly: Application to the green fluorescent protein. Journal of the American Chemical Society 122: 5658-5659.
37. Kodama Y, Wada M (2009) Simultaneous visualization of two protein complexes in a single plant cell using multicolor fluorescence complementation analysis. Plant Mol Biol 70: 211-217.
38. Hu CD, Kerppola TK (2003) Simultaneous visualization of multiple protein interactions in living cells using multicolor fluorescence complementation analysis. Nature Biotechnology 21: 539-545.
39. Shaner NC, Steinbach PA, Tsien RY (2005) A guide to choosing fluorescent proteins. Nature Methods 2: 905-909.
40. Zimmermann T (2005) Spectral imaging and linear unmixing in light microscopy. Microscopy Techniques. pp. 245-265.
41. Dickinson ME, Bearman G, Tille S, Lansford R, Fraser SE (2001) Multi-spectral imaging and linear unmixing add a whole new dimension to laser scanning fluorescence microscopy. Biotechniques 31: 1272-+.
42. Lansford R, Bearman G, Fraser SE (2001) Resolution of multiple green fluorescent protein color variants and dyes using two-photon microscopy and imaging spectroscopy. Journal of Biomedical Optics 6: 311-318.
43. Huh WK, Falvo JV, Gerke LC, Carroll AS, Howson RW, et al. (2003) Global analysis of protein localization in budding yeast. Nature 425: 686-691.
44. Gurskaya NG, Fradkov AF, Pounkova NI, Staroverov DB, Bulina ME, et al. (2003) Colourless green fluorescent protein homologue from the non-fluorescent hydromedusa Aequorea coerulescens and its fluorescent mutants. Biochemical Journal 373: 403-408.

45. Raser JM, O'Shea EK (2005) Noise in gene expression: origins, consequences, and control. Science 309: 2010-2013.
46. Heikal AA, Hess ST, Baird GS, Tsien RY, Webb WW (2000) Molecular spectroscopy and dynamics of intrinsically fluorescent proteins: Coral red (dsRed) and yellow (Citrine). Proceedings of the National Academy of Sciences of the United States of America 97: 11996-12001.
47. Goh C-K (2009) Evolutionary multi-objective optimization in uncertain environments : issues and algorithms. New York: Springer.
48. Deb K (2001) Multi-objective optimization using evolutionary algorithms. Chichester ; New York: John Wiley & Sons. xix, 497 p. p.
49. Cai Y, Hartnett B, Gustafsson C, Peccoud J (2007) A syntactic model to design and verify synthetic genetic constructs derived from standard biological parts. Bioinformatics 23: 2760-2767.
50. Colmerauer A (1990) An Introduction to Prolog-Iii. Communications of the Acm 33: 69-90.
51. Sangiovanni-Vincentelli A (2003) The tides of EDA. Ieee Design & Test of Computers 20: 59-75.
52. Czar MJ, Cai Y, Peccoud J (2009) Writing DNA with GenoCAD. Nucleic Acids Res (in press).
53. Benini L, De Micheli G (2000) System-level power optimization: Techniques and tools. Acm Transactions on Design Automation of Electronic Systems 5: 115-192.
54. Bolsens I, DeMan HJ, Lin B, VanRompaey K, Vercauteren S, et al. (1997) Hardware/software co-design of digital telecommunication systems. Proceedings of the Ieee 85: 391-418.

CRITICAL ANALYSIS OF TRANSCRIPTIONAL AND POST-TRANSCRIPTIONAL REGULATORY NETWORKS IN MULTIPLE MYELOMA

MARTA BIASIOLO[1*], MATTIA FORCATO[2*], LINO POSSAMAI[3], FRANCESCO FERRARI[1], LUCA AGNELLI[4], MARTA LIONETTI[4], KATIA TODOERTI[4], ANTONINO NERI[4], MASSIMO MARCHIORI[3], STEFANIA BORTOLUZZI[1], SILVIO BICCIATO[2]

[1]*Dipartimento di Biologia, Università di Padova, via U. Bassi 58/ B, 35121, Padova, Italy*
[2]*Dipartimento di Scienze Biomediche, Università di Modena, via G. Campi 287, 41125, Modena, Italy*
[3]*Dipartimento di Matematica Pura ed Applicata, Università di Padova, via Trieste 63,35121, Padova, Italy*
[4]*Dipartimento di Scienze Mediche, Università di Milano, Ematologia 1, CTMO, Fondazione IRCCS Ospedale Policlinico, Via F. Sforza, 35, 20122 Milano*

Network analysis has emerged as a powerful approach to understand complex phenomena and organization in social, technological and biological systems. In particular, it is increasingly recognized the role played by the topology of cellular networks, the intricate web of interactions among genes, proteins and other molecules regulating cell activity, in unveiling the biological mechanisms underlying the physiological states of living organisms. In this study, critical analysis of network components has been applied to inspect the transcriptional and post-transcriptional regulatory networks reconstructed from mRNA and microRNA expression data of multiple myeloma (MM) samples. Specifically, the importance of a gene as a putative regulatory element has been assessed calculating the drop in the network performance caused by its deactivation instead of quantifying its degree of connectivity. The application of critical analysis to transcriptional and post-transcriptional regulatory networks allowed inferring novel regulatory relations potentially functional in multiple myeloma.

1. Introduction

Systems biology elevates the study from the single entity level (e.g., genes, proteins) to higher hierarchies, such as entire genomic regions, groups of co-expressed genes, functional modules, and networks of interactions. The functioning and development of a living organism is controlled by the networks of relations among its genes (as well as proteins and small molecules) and the signals regulating each gene (or set of genes), therefore understanding how elementary biological objects act together and interact in the general context of a genome is fundamental to the advancement of science. As such, the scientific attention is focusing more and more on the critical levels of biological organization and their emerging properties rather than on the single components of the system [1]. However, despite the significant advances in genome sequencing and in transcription, protein and metabolite profiling, still significant limitations hamper the global understanding of regulatory phenomena. The control of biochemical processes is hierarchical and originates at the level of transcription (induction-repression mechanism and mRNA degradation), moving on to translation (protein activation and proteolysis) and enzyme activity through signaling cascades. The presence of several feedback loops among these regulatory processes makes their organization and functioning very complex. This level of complexity can, at least in part, be addressed using methods that allow the reverse engineering and the reconstruction of regulatory networks. In this context, the availability of high-throughput genomic data, coupled with bioinformatics tools for their analysis, represents a promising starting point in the identification of molecular interaction networks which will allow turning genomic researches into accurate biological hypotheses.

Microarray experiments have been extensively used to detect patterns in gene expression that stem from regulatory interactions and lately have been applied to analyze the transcriptional activity of microRNAs (miRNAs), i.e. small non-coding RNAs that regulate the post-transcriptional mRNA stability by binding 3' target sites. The availability of a sufficient number of matched miRNA-mRNA expression profiles represents a further opportunity to deepen the study of transcriptional regulation. However, standard methodologies for the analysis of gene expression profiles, which aim at identifying relevant genes from the statistical analysis of the microarray signals, seem to be severely limited in unveiling the mechanisms governing the transcriptional cascade. Although proving their effectiveness e.g., in identifying expression signatures for cancer diagnosis, most computational tools have fallen short of representing a systematic method for understanding how the transcriptional regulation process takes place.

* These authors equally contributed.

Bioinformatics and computational biology need to overcome this limitation and develop approaches to identify regulatory networks, through the integration of multiple types of data. These computational methods should help deciphering how the transcribed elements of genomes impact the molecular mechanisms of functional utilization.

Network analysis has emerged as a powerful approach to understand complex phenomena and organization in social, technological and biological systems [2-4]. In particular, it is increasingly recognized the role played by the topology of cellular networks, the intricate web of interactions among genes, proteins and other molecules regulating cell activity, in unveiling the function and the evolution of living organisms [5-9]. Gene networks, in this respect, present a unique opportunity to employ this new type of approach [10-11]. In general, transcriptional regulatory networks are structures where the nodes are the genes and the edges are the interactions. A gene can be considered the source of a direct regulatory edge if it encodes a molecule with known regulatory function. Algorithms to infer the structure of gene-gene relationships take as primary input the data from a set of microarrays measuring the mRNA expression levels in different physiological states and use either classical statistics (e.g., Pearson correlation), concepts from the information theory (i.e., the mutual information as in ARACNe (Algorithm for the Reconstruction of Accurate Cellular Networks [12]) and CLR algorithms [13]) or probabilistic models (as in the Bayesian networks [14]) to reconstruct the network of transcriptional interactions. Once reconstructed, a gene regulatory network can be inspected and analyzed at different levels, ranging from the single gene to groups of genes (sub-networks) and to the whole network. Putative regulatory targets of a gene of interest can be searched at the single gene level with the goal of identifying previously unknown targets that can be the focus of subsequent experimental validation. At a higher level, the presence of groups of genes organized in sub-networks may suggest novel interactions and shed light on regulatory modules involving these genes or their common targets. Nevertheless, both strategies rely on prior knowledge to select the genes of interest, thus hampering the capacity of extracting *de-novo* knowledge from the network. To overcome this limitation, the inference algorithms used to reconstruct gene regulatory networks are often supplemented with methods derived from the network theory, i.e. borrowing analytical schemas typically used to explore communication or infrastructure networks. For instance, network theory recently focused on the resilience of complex networks to the malfunctioning of its component and to external disturbances. A key aspect of this analysis is the identification of the most *critical components* of the networks, i.e. those nodes/edges that are really crucial for the functioning of the system [15]. Specifically, the importance of an element is assessed considering the drop in the network performance caused by its deactivation and is quantified calculating the performance of the perturbed network as compared with the original one. The very same approach can be adopted to dissect gene regulatory networks for the identification of critical genes and interactions.

Here, the analysis of the network *critical components* was applied on the nodes of transcriptional and post-transcriptional regulatory networks reconstructed from mRNA and miRNA expression data of multiple myeloma samples with the aim of i) identifying genes that are critical for the structure of the networks irrespectively of the number of their ingoing or outgoing links, i.e. of the fact that they are hubs or not, and ii) inferring novel regulatory relationships.

2. Materials and Methods

2.1. *mRNA and miRNA expression data*

Gene (mRNA) and miRNA expression data were obtained from multiple myeloma (MM) specimens. In details, the gene expression dataset (hereafter denoted as MM158GE) comprises 5 normal, 11 monoclonal gammopathies of unknown significance (MGUS), 133 MM, and 9 plasma cell leukemia (PCL) for a total of 158 samples [16]. Matched mRNA and miRNA expression data (hereafter denoted as MM40MGE) have been obtained for a subset of samples included in the MM158GE dataset. In particular, 40 MM samples representative of five translocation groups (9 TC1, 10 TC2, 9 TC3, 7 TC4 and 5 TC5) have been analyzed using both Affymetrix HG-U133A and Agilent Human miRNA Microarray V2 arrays. The Agilent Human miRNA Microarray V2 consists of 60-mer DNA probes synthesized in situ, which represent 723 human and 76 human viral miRNAs derived from the Sanger database v10.1. Gene expression signals have been quantified using RMA (*affy* Bioconductor package) and the

GeneAnnot custom Chip Definition Files [17]. Genes with low signal variability across samples were eliminated using an entropy-based filter. Briefly, given the expression levels $w_{g,}$ of gene g in sample t ($t=1,...,N$), the entropy of the expression distribution was defined as:

$$H_g = \sum_{1 \le t \le N} -p_{g|t} \log_2\left(p_{g|t}\right) \quad (1)$$

where

$$p_{t|g} = w_{g,t} \Big/ \sum_{1 \le t \le N} w_{g,t} \quad (2)$$

is the relative expression of gene g in sample t. The entropy threshold was selected in order to eliminate the 20% of less variable data [18]. Expression signals of the miRNAs arrays have been normalized using *aroma.light* Bioconductor package.

2.2. *Network critical components analysis*

The topological structure of a network can be used to identify the components (nodes or links) that are critical for the functioning of the system (*critical components*). *Network critical components analysis* has been successfully applied in different fields as communication or transportation. For instance, critical analysis is used to identify nodes that must be protected from terrorist attacks in communication networks, in social networks finding critical nodes can be fundamental to reduce the spreading of viruses, and in biological systems, this analysis can be extremely helpful to understand complex phenomena and to find more powerful ways to defend the system from a disease. Nodes and links can be removed using various techniques and different networks exhibit different levels of resilience to such disturbances. Networks can be perturbed simulating the deletion of node/links chosen at random (*error* removal or *failure*) or targeting a specific class of nodes/links (removal through intentional *attacks*). Attacks can be addressed sorting and removing progressively the nodes in descending order of degree or betweenness or the links in descending order of betweenness or range [19-21]. The network robustness is usually measured by the size of the largest connected component and by the average node-node distance as a function of the percentage of nodes/links removed.

The method used here to identify the critical components of gene regulatory networks is based on an *ad-hoc* definition of network performance, rather than on local node information such as the number of ingoing or outgoing links. Specifically, the importance of a node is measured by the drop in the network efficiency caused by the removal of that node, where the network efficiency *E(G)* quantifies how efficiently the nodes of the network exchange information [22]. The definition of *E(G)* requires recalling some formalism from the graph theory.

A network can be modeled by a graph G of nodes that are tied by one or more specific type of interdependency. Formally, an *undirected* graph $G=(N, L)$ consists of two sets N and L such that $N \neq \varnothing$ and L is a set of unordered pairs of element of N. The elements of $N=\{n_1,n_2,...n_M\}$ are the nodes of the graph G while the elements of $L=\{l_1,l_2,...,l_K\}$ are the edges. Two nodes joined by an edge are referred to as *adjacent* or *neighboring*. A graph is *weighted* when there exists a function $w : L \to \Re$ from edges to real numbers, such that each edge has associated a number that represents the strength of the connection. A graph is called *m-partite* if N admits a partition into m classes such that every edge has its ends in different classes: vertices in the same partition class must not be adjacent. When m=2, the graph is called *bipartite*. A *walk* from node i to node j is an alternating sequence of nodes and edges that begins with i and ends with j. If no node is visited more than once, the walk is called a *path*. A graph G is said to be *connected* if, for every pair of distinct nodes i and j, there is a path from i to j in G. The degree or connectivity k_i of a node i is the number of edges incident with the node, i.e. the number of neighbors of that node. One of the most relevant topological characterizations of a graph G can be obtained from the degree distribution *P(k)*, which is normally represented plotting the number of nodes having degree of connectivity k against k in a log-log scale. A decreasing linear dependency in this plot indicates that the network has a *scale-free* structure, associated with a corresponding power-law $n(k) \propto k^{-\gamma}$ (Figure 1). Graphs can be further classified as *assortative* if $k_{nn}(k)$, i.e., the average degree of the neighbors of degree k, is an increasing function of k; otherwise they are referred to as

disassortative. In *assortative* networks the nodes tend to connect to their connectivity peers, while in *disassortative* networks nodes with low degree are more likely connected with highly connected ones.

The efficiency of G relies on the calculation of the shortest path lengths d_{ij} between two generic nodes *i* and *j*. In a weighted graph d_{ij} is defined as the smallest sum of the physical distances throughout all the possible paths in the graph from *i* to *j*, while in an un-weighted graph d_{ij} reduces to the minimum number of edges traversed to get from *i* to *j*. The maximum value of d_{ij} is called the *diameter* of the graph and the average shortest path length *L* is quantified as follows:

$$L = \frac{1}{M(M-1)} \sum_{i,j \in N, i \neq j} d_{ij} \tag{3}$$

Supposing that every node sends information along the network, through its links, the efficiency ε_{ij} in the communication between node *i* and node *j* is assumed to be inversely proportional to their shortest distance, i.e. $\varepsilon_{ij} = 1/d_{ij} \ \forall i, j$. It's worthwhile noting that the assumption that efficiency and distance are inversely proportional is a reasonable approximation although sometimes other relationships might be used, especially if justified by a more specific knowledge of the system. By assuming $\varepsilon_{ij} = 1/d_{ij}$, when there is no path in the graph between *i* and *j*, $d_{ij} = +\infty$ and consistently $\varepsilon_{ij} = 0$. Consequently, the average efficiency *E(G)* of the graph *G* can be defined as:

$$E(G) = \frac{1}{M(M-1)} \sum_{i,j \in N, i \neq j} \varepsilon_{ij} = \frac{1}{M(M-1)} \sum_{i,j \in N, i \neq j} \frac{1}{d_{ij}} \tag{4}$$

The definition of *E(G)* according to Eq.(4) avoids the divergence of *L* in case of disconnected components thus allowing the analysis of the entire network and not only of the biggest connected sub-graph. Since *E(G)* varies in the range $[0,\infty]$, it would be more practical to normalize *E(G)* in the interval [0,1]. The most natural way to normalize *E(G)* is with respect to the efficiency of a network G^{ideal} composed of all the $M(M-1)/2$ possible edges:

$$E_{glob} = \frac{E(G)}{E(G^{ideal})} \tag{5}$$

Though the maximum value *E(G)*=1 is reached only when there is a link between each pair of nodes, real networks can nevertheless assume high values of E. This definition is valid for both un-weighted and weighted graphs and can also be applied to disconnected graphs.

The efficiency can be evaluated on any *sub-graph G'=(N',L')* of *G=(N,L)*, where *G'* of *G* is a graph such that $N' \subseteq N$ and $L' \subseteq L$. The sub-graph of the neighbors of a given node *i*, denoted as G_i, is the sub-graph induced by N_i, i.e., the set of nodes adjacent to *i*. Given c_i the node cardinality of G_i, the local efficiency E_{loc} is defined as the average of the sub-graph efficiencies $E(G_i)$ normalized with respect to the ideal sub-graphs in which all the $c_i(c_i-1)/2$ edges are present:

$$E_{loc} = \frac{1}{M} \sum_{i \in G} \frac{E(G_i)}{E(G_i^{ideal})} \tag{6}$$

Since $i \notin G_i$, the local efficiency E_{loc} quantifies the efficiency of the system in tolerating faults, i.e., how efficient is the communication between the first neighbors of *i* when *i* is removed. Graphs that have high value of E_{glob} and E_{loc}, i.e., that are very efficient both in their global and local communication, are defined as *small-words networks*.

Given the definition of *E(G)* and assuming that the efficiency is an appropriate quantity to characterize the average properties of a network, critical components can be identified considering the efficiency drop, caused by the deactivation of a component, as a measure of the centrality of that component. Therefore, the topological importance of a node α in a graph is quantified by the *network relevance* r_α:

$$r_\alpha = \frac{E(G) - E(G_\alpha)}{E(G)} = \frac{\Delta E_\alpha}{E} \tag{7}$$

where G_α is the graph obtained by removing node α from G, for each $\alpha = 1,\ldots,M$. The most critical nodes are those whose removal causes the largest drop in efficiency, i.e., those with the highest r_α (Figure 1). Although here the focus is on the determination of the critical nodes, the method is of general applicability to any subset (nodes, links and combination of nodes and links) of G [23].

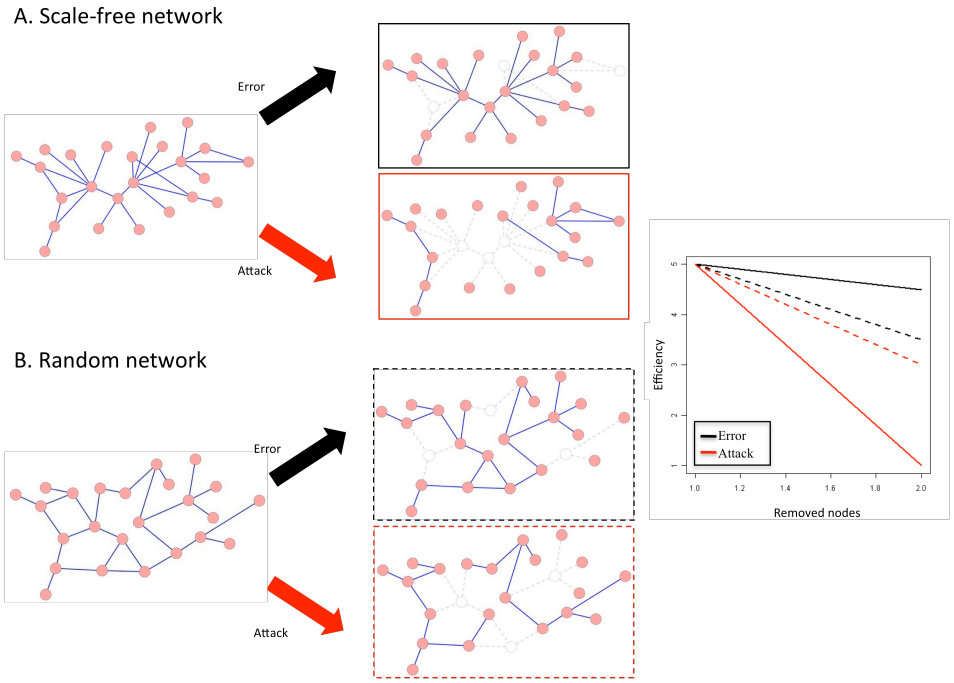


Figure 1. Effect of node removal through *error* and *attack* in scale free (A., solid lines) and in random (B., dotted lines) networks. In a scale-free network (solid lines), the random removal (*error*) of even a large fraction of vertices impacts the overall connectedness of the network very little (black line), while targeted *attack* (red line) destroys the connectedness very quickly, causing a rapid drop in efficiency. On the contrary, in random graphs, removal of nodes through either *error* or *attack* has the same effect on the network performance.

2.3. *Transcriptional regulatory network*

The transcriptional regulatory network was reconstructed using ARACNe and the MM158GE dataset of gene expression signals. ARACNe utilizes information and data transmission concepts (i.e., mutual information and data processing inequality) to identify statistically significant co-regulations among genes from microarray expression profiles. Mutual information and data processing inequality allow reconstructing gene-gene relationships which most likely represent either direct regulatory interactions or interactions mediated by post-transcriptional modifiers. Briefly, the algorithm first uses the expression data to calculate pair wise Mutual Information (MI) through a computationally efficient Gaussian kernel estimator. ARACNe calculates the kernel width depending on the size and statistics of the dataset. The second step is the elimination of the interactions that are not statistically significant according to a p-value or a MI threshold and returns a series of irreducible statistical dependencies. The post-processing step eliminates interactions that are likely to be indirect. The Data Processing Inequality (DPI) theorem removes indirect regulatory influences that appear as direct because of a high MI score due to the presence of a common neighbor. An additional parameter, called DPI tolerance, can be used to compensate for errors in the MI estimate that might affect DPI application [12].

The parameters of the kernel width and the Mutual Information threshold were calculated using MATLAB scripts. The p-value to determine the MI threshold was set at $1e^{-7}$, while the DPI tolerance was set equal to 10%. A list of Transcription Factors (TF) for the platform HG-U133A was also imputed as a parameter to prevent the DPI from removing transcriptional interactions in favor of non-transcriptional ones (interactions between two non-TFs).

2.4. *Post-transcriptional regulatory network*

The post-transcriptional regulatory network was reconstructed calculating the Pearson correlation coefficient of the expression vectors of miRNA target genes in the MM40MGE dataset. In details, the procedure required to i) identify the miRNA target genes basing on computational predictions; ii) select the target relationships supported by miRNA and mRNA expression data; iii) compute the Pearson correlation coefficient of the expression levels of target genes sharing at least one supported miRNA-target relationship; iv) reconstruct the post-transcriptional regulatory network from the adjacency matrix **S** of regulatory relations supported by miRNA and mRNA expression levels. Computational prediction of miRNA targets presents significant challenges due to the lack of a sufficiently large group of known miRNA targets to be used as training set. As such, most computational algorithms for target prediction (miRanda, TargetScan, PicTar, PITA, RNAhybrid) result in a significant proportion of false positives, i.e. in the prediction of not-functional miRNA-mRNA interactions. Given the increasing experimental evidences supporting the miRNA mechanism of target degradation, the integration of *in-silico* predictions with miRNA and target gene expression profiles has been proposed as a method to select functional miRNA-mRNA relationships. Since miRNAs tend to down-regulate target mRNAs, the expression profiles of genuinely interacting pairs are expected to be anti-correlated. This integrative analysis can be performed using a variational Bayesian model [24] or, as in this case, through a non-heuristic methodology based on the anti-correlation between miRNA and mRNA matched expression profiles [25-26].

Specifically, miRanda algorithm [27] was applied to predict miRNA targets from the human miRNA sequences and transcripts of miRBase Release 12.0 and ENSEMBL Release 52, respectively. Targeting predictions were retained if the miRanda score was higher than 160. The Pearson correlation coefficient of expression vectors was calculated for each miRNA-gene pair scored as potentially interacting according to the prediction of miRanda and used as an estimator of the functional activity of miRNAs on predicted target genes. Genes were considered genuine miRNA targets only if included within the top 3% of all anti-correlated pairs [25]. This selection gave rise to a final adjacency matrix **S** of regulatory relations supported by expression levels. The adjacency matrix **S** defined a bipartite directed network with two types of nodes (miRNAs and mRNAs) connected by directed edges, each representing a probably functional regulatory effect of a miRNA on a target gene. The same matrix **S** was used to derive a gene-only network in which genes (nodes) are connected by undirected weighted links and the edge weight quantifies the number of shared miRNAs regulating each gene pair.

3. Results

ARACNe inferred a transcriptional network with 9666 nodes (i.e. genes) and 86846 edges (i.e. interactions) from the MM158GE dataset. The topological characteristics of the network are reported in Table 1, in terms of number of nodes, number of edges, maximum k_{max} and average k_{mean} connectivity (k being the degree of a node, i.e. the number of its interactions), diameter (representing the maximum value of d_{ij}) and global and local efficiencies.

Table 1. Metrics of transcriptional and post-transcriptional networks.

Network type	Nodes (M)	Edges (K)	k_{max}	k_{mean}	Diameter	E_{glob}	E_{loc}
Transcriptional	9666	86846	219	17.96	8	0.279	0.150
Post-transcriptional	6435	909324	1811	282.62	8	0.611	0.866

The connectivity distribution shows a power-law tail suggesting that the underlying structure of the network is scale-free (Figure 2A). At low connectivity values (k<11), the degree distribution loses its linear progression probably as a consequence of the limited number of genes. The relationship between the average connectivity k_{nn} of the neighbors of a node and the node connectivity suggests an assortative behavior of the network, i.e., the nodes tend to connect with nodes with a similar connectivity thus partly implying a hierarchical structure of the network (Figures 2B). Ranking the nodes according to their connectivity allowed indentifying 27 *hubs*, i.e. genes with more than 100 interactions (data not shown).

The miRNA-mRNA integrated analysis resulted in a post-transcriptional gene network with 6435 nodes (genes) and 909324 weighted edges. The network was reconstructed first refining the predicted targeting relationships of

MiRanda through the selection of those predictions more supported by miRNA-mRNA expression data (the 3% most highly anti-correlated miRNA-gene pairs). This corresponded to 23729 regulatory relations involving 692 miRNAs and 6,435 target genes. It's worth noting that about 48% of genes associated to an expression profile resulted not to be real target of any considered miRNA and 9 miRNAs were not detected as sufficiently active on any target gene. Then, the remaining 692 miRNAs and 6,435 target genes were employed to reconstruct a bipartite directed miRNAs-mRNAs regulatory network, representing the probably functional regulatory effects of all these miRNA to their targets in MM. The number of target genes per miRNA ranges from 1 to 440 (average 33.3 with a mean value of 3.7 miRNAs per gene). Finally, a weighted post-transcriptional network of 6435 genes was extracted from the bipartite miRNA-mRNA regulatory network with the weight of an edge representing the number of functional interactions with microRNAs shared by the couple of connected genes. The topological characteristics of the network are reported in Table 1. Similarly to the transcriptional network, the connectivity distribution and the relationship between average connectivity k_{nn} of the neighbors of a node and the node connectivity suggest a scale-free, assortative structure (Figures 2C and 2D).

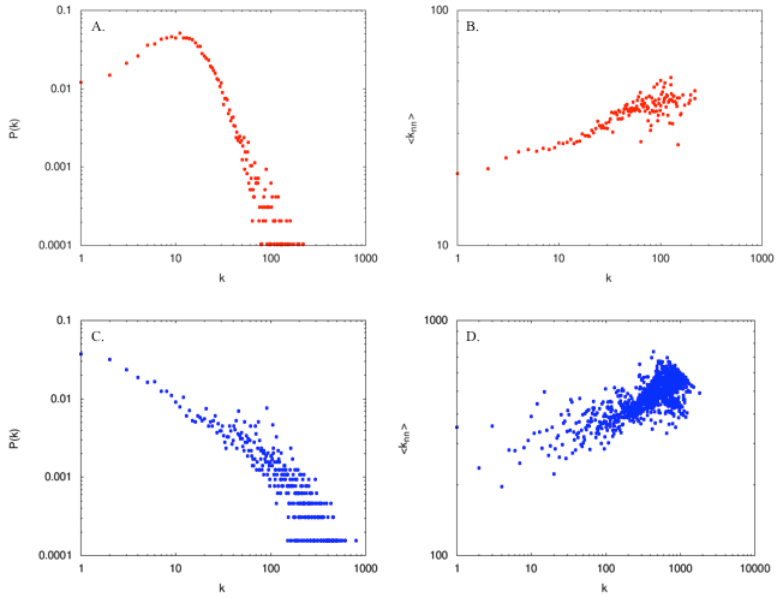


Figure 2. Connectivity properties of transcriptional (A. and B.) and post-transcriptional (C. and D.) networks. A. Connectivity distribution *P(k)* of nodes with a specific number of incident edges (degree of connectivity *k*) in the transcriptional network. The connectivity distribution shows a power-law tail suggesting a scale-free structure of the network. B. Relationship between the average connectivity k_{nn} of the neighbors of a node and the node connectivity *k*. The linear trend suggests an assortative behavior of the transcriptional network. C. Same as A. for the post-transcriptional network. D. Same as B. for the post-transcriptional network.

The critical nodes of both the transcriptional and post-transcriptional regulatory networks have been determined by the static analysis of *error* and *attack* tolerance. The drop in the network efficiency caused by the node removal (i.e., the node relevance r_α as defined in Eq. (7)) has been used as the criteria to determine the importance of a node. The critical analysis has been applied to the transcriptional network, to the post-transcriptional one, and, for testing the robustness of results, to a random graph [28]. The random network has been constructed starting from an initial condition of *M* nodes and no edges and then adding *K* edges between pairs of randomly selected nodes, where *M* and *K* were the same as in the transcriptional network. Figures 3A and 2B show the global efficiency for the transcriptional scale-free network and for the random graph (both with *M=9666* nodes and *K=86846* edges) as functions of the number of removed nodes through efficiency-based attacks (i.e., attacks performed removing nodes with the highest efficiency; red line) and random removals (errors; black line). The true transcriptional network shows a different behaviour with respect to attacks and errors (Figure 3A). The removal of ~30% of nodes in a

targeted way (*attack*) reduces the network efficiency to about half the initial value and removing ~60% of the nodes destroys completely the system. Instead, when removing nodes randomly (*error*), the drop of the network global efficiency shows a linear dependency with the number of removed nodes and even for high value of removals (>60%) the system maintains a considerable efficiency (Figure 3C). The fact that removing specific nodes causes a rapid drop in the capability of the system to communicate further supports the scale-free structure of the regulatory graphs and proves the existence of a discrete number of *critical components*, i.e. of nodes responsible for the specific structure of the network. As far as the random graph is concerned (Figures 3B and 3C), differences of tolerance to *attacks* and to *errors* are much less pronounced. In this case, in fact, there is no substantial variability in the efficiency and the removal of a node in a targeted or in a random way produces similar, though not equal, behaviours.

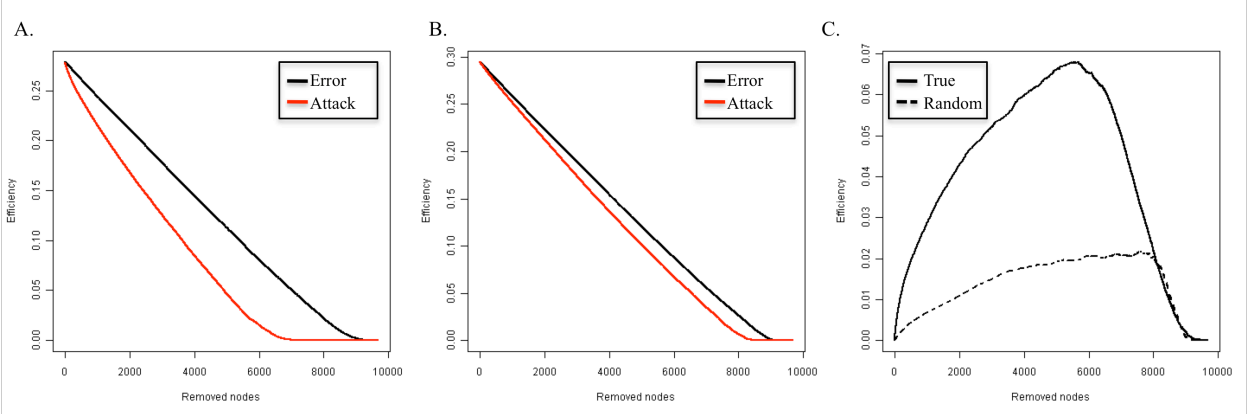


Figure 3. Global efficiency E(G) as a function of the number of removed nodes. In both the cases, the graphs were composed of M=9666 nodes and K=86846 edges and the node removal simulated by errors (black line) and efficiency-based attacks (red line). A. Transcriptional network generated by ARACNe using gene expression data from the MM158GE dataset. B, Random graph. C. Difference of tolerance to attacks and to errors (i.e., difference between the drop in efficiency caused by efficiency-based attacks and error node removals) for true transcriptional (unbroken line) and the randomly generated networks (broken line).

The analysis of critical components revealed that, in the transcriptional and post-transcriptional networks, critical nodes are not limited to hub genes and that also genes with a limited number of connections can be *critical* for the structure of the network. Figures 4A, 4B and 4C report the comparison between the node rankings calculated according to node degree (k) and node *criticality* (r_α) in the random, transcriptional, and post-transcriptional networks.

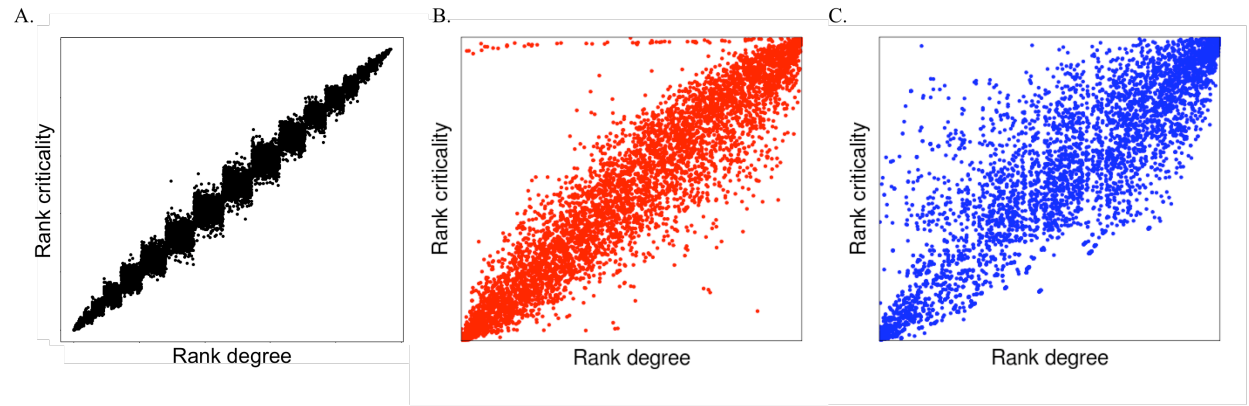


Figure 4. Comparison between the rank according to node degree k (rank degree) and the rank given by criticality r_α (rank criticality) for the nodes of A. random, B. transcriptional, and C. post-transcriptional networks.

As expected, this comparison indicates that i) in random network there is no significant difference between node degree and node criticality (Figure 4A) and ii) the vast majority of hub genes are critical nodes, i.e., the nodes whose removal causes a large drop in global efficiency correspond to the most connected genes (Figures 4B and 4C). Nevertheless, there exist a fraction of nodes that are characterized by a higher criticality value than expectable according to their degree. These nodes, even if characterized by a low node degree, are indeed critical and would have been disregarded as putative regulatory targets due to their limited number of connections. For instance, in the MM transcriptional network, the B cell linker gene (*BLNK*) emerged as one of the most critical genes although being connected to only 34 other nodes (as compared to hubs characterized by >100 links). *BLNK* is known to be involved in normal B-cell development and deficiency in this protein has been shown in some cases of pre-B acute lymphoblastic leukemia, suggesting its putative role as a tumor suppressor gene. A search in the network for putative targets of *BLNK* allowed identifying 8 genes that, once ranked according to the mutual information, indicated *CDKN1B* (cyclin dependent kinase inhibitor, p27 kip1) as the most strongly connected target. This interaction was confirmed by the same analysis conducted on another MM dataset ([29]; data not shown) and by recent experimental evidences, which reported p27 kip1 induction by *BLNK* trough *JAK3* [30]. Unfortunately, *JAK3* was not present in the datasets used for the network reconstruction and thus this evidence could not be further confirmed.

To integrate the results of the critical analyses, a list of 5145 non redundant genes/nodes represented in both transcriptional and post-transcriptional networks was compiled and used to select, in each network, the top 1% critical nodes and the top 1% most connected nodes (hubs). The intersections of such lists could help clarifying the role of nodes, which are critical in terms of network efficiency, although being not highly connected (Figure 5).

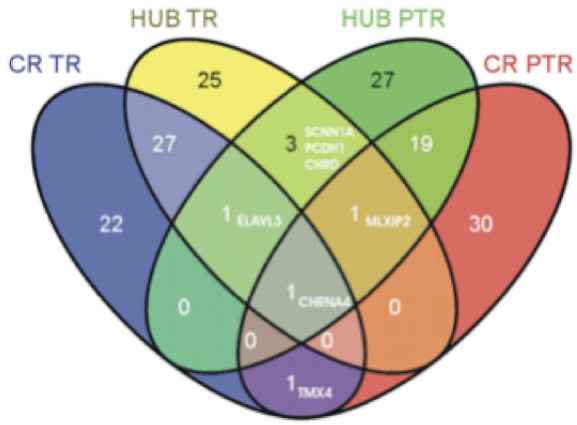


Figure 5. Venn diagram showing the intersections of four sets: CR TR (top 1% critical nodes in the transcriptional regulatory network), HUB TR (1% most connected nodes, hubs, in the transcriptional regulatory network), CR PTR (top 1% critical nodes in the post-transcriptional regulatory network) and HUB PTR (1% most connected nodes, in the post-transcriptional regulatory network).

In particular, when comparing the lists of top 1% critical genes for the two networks, 2 genes emerged as critical for both transcriptional and post-transcriptional regulation, i.e. *CHRNA4* (cholinergic receptor, nicotinic, alpha 4) and *TMX4* (thioredoxin-related transmembrane protein 4), both known to be expressed in B cells and the latter involved in B cells activation and in cancer. Interestingly, *CHRNA4* shares, as post-transcriptional regulators supported by expression data, two miRNAs (i.e., hsa-miR-15a* and hsa-miR-30c-1*) that regulate also *ELAVL3* and *MLXIPL*, two genes which are hubs and critical in the transcriptional and post-transcriptional network, respectively. Both miR-15a and miR-30c are differentially expressed in peripheral blood cells [31] and the presence of a *MYB*-miR-15a auto-regulatory feedback loop is of potential importance in human hematopoiesis [32]. In particular, miR-15a expression inversely correlates with *MYB* expression in cells undergoing erythroid differentiation and the over-expression of miR-15a blocks both erythroid and myeloid colony formation *in-vitro*.

4. Discussion

Reconstructing regulatory network from expression data is a crucial step to understand the mechanisms underlying biological systems. However, the high number of genes and interactions still represents a challenging issue for the extraction of relevant targets and relationships from such large systems. A standard approach is searching targets among the most connected genes (hubs) or among sub groups of genes known to be relevant in the analyzed phenotype. The goal of this type of analysis is to identify previously unknown relationships that can be the object of a subsequent experimental validation. An alternative approach is studying the network characteristics to identify groups of genes organized in sub-networks, which may suggest novel interactions and shed light on regulatory modules involving these genes or their common targets. Although effective, both strategies rely on prior knowledge to select the genes of interest, thus hampering the capacity of extracting *de-novo* knowledge from the network. A way to overcome this limitation could be adapting techniques commonly used in the analysis of communication and infrastructure networks. In these fields, a key analysis is the resilience of the network to external disturbances and to malfunctioning. Network robustness strongly relies on the network structure and, in particular, on the existence of paths between the nodes. When nodes or links are removed, the lengths of these paths can increase and some nodes will become disconnected. It is therefore interesting to find the critical component of the network, i.e. the nodes or edges that are really important for the functioning of the network. In Latora and Marchiori, the authors proposed a method to evaluate the importance of a network element (that can be a node or an edge) by considering the drop in the network performance caused by its deactivation [15]. The performance of the perturbed network is compared with the original one. Iturria-Medina and colleagues applied a similar approach to investigate the human brain anatomical network [33].

Different criteria can be used to measure the performance, such as efficiency or mean flow rate of information. Efficiency measures how efficiently the nodes of the network exchange information. Applying this concept to regulatory network, critical nodes and edges are critical genes and critical regulatory interaction, respectively. Usually the most important nodes are considered the most connected ones (hubs), but this is not always the case [23]. In genetic networks a gene can be connected to many genes simply because is a transcription factor that normally controls many targets or a gene that is controlled by many other genes. For instance, in the analysis of the B cell networks, the largest hub with more than 300 interactions was a poorly characterized gene, *BYSL*. Instead, a much more interesting gene was *MYC* that, with only 56 neighbours, ranked 410th in terms of connectivity. *MYC*, a well-known proto-oncogene, had neighbours that were themselves genetic hubs (including *BYSL*), such that *MYC* could modulate a substantial percentage of all genes in the cell trough a relatively small number of neighbours [12].

Recently, some approaches exploited the topological features of large gene regulatory networks to identify individual components that are biologically relevant or to elucidate the role of each particular element in regulation. Patapov and co-workers introduced the pair-wise disconnectivity index to quantitatively evaluate the topological significance of each element (i.e., nodes and edges) in the context of all other elements of the regulatory network [34]. The application of this approach to the analysis of the TLR4 signal transduction network allowed identifying a number of key signalling and transcription regulators among the nodes top-ranking in terms of disconnectivity index. Differently, Emmert-Streib and Dehmer used the concept of functional robustness, originally introduced by Li et al. [35], to study the functional robustness of the transcriptional regulatory network in yeast [36]. The definition of an information theoretic measure to estimate the influence of single node perturbations on the global network topology allowed identifying nodes which are fragile with respect to single node knockouts and revealed significant differences between fragile nodes and hubs. Interestingly, the set of fragile nodes was statistically enriched in essential genes, i.e. in genes required to sustain vital yeast.

Here, the critical analysis of network components has been applied to inspect the transcriptional and post-transcriptional regulatory networks reconstructed from mRNA and miRNA expression data of multiple myeloma samples. The transcriptional and post-transcriptional networks were reconstructed using ARACNe and the Pearson correlation coefficient of the expression vectors of miRNA target genes, respectively. Both networks showed a scale free structure, i.e. a type of structure reported with evidence in lower organisms, but still argument of debate in eukaryotes. The connectivity plots of Figure 2 strengthen the hypothesis that the structure of human interaction

networks has a scale free nature with a saturation effect also reported for other scale-free networks, when the maximum connectivity range is below 1000 [12, 37-39]. Both networks are also slightly assortative, meaning that they tend to have an aristocratic behaviour where nodes with high degree tend to connect with nodes with similar degree. This suggests a hierarchical control mechanism, as also reported in [12]. The analysis of critical components revealed that genes with a limited number of connections could be critical for the structure of the network and that hubs are not necessarily critical nodes. Indeed, about one half of most connected nodes in each considered network were not included in the corresponding list of most critical nodes and genes like *BLNK*, characterized by a low node degree, were instead critical. These *non-hub* critical nodes would have been disregarded as putative regulatory targets due to their limited number of connections although they may provide clues to the detection of key regulatory circuits. Finally, the integration of the transcriptional and post-transcriptional levels allowed identifying critical genes for both types of regulatory interactions and dissecting direct critical relationships at transcriptional level from interaction that are instead indirect since mediated by post-transcriptional regulation.

5. Acknowledgments

This work was supported by grants from Fondazione CARIPARO (Progetti Eccellenza 2006); MIUR (PRIN 2007Y84HTJ and PRIN 2007CHSMEB); University of Padova (CPDA065788/06 and CPDR074285/07); University of Modena (Finanziamento Linee Strategiche di Sviluppo dell'Ateneo, Medicina Molecolare e Rigenerativa, 2008); Fondazione Cassa di Risparmio di Modena (Bando ricerca 2007), and Associazione Italiana Ricerca sul Cancro (AIRC).

6. References

1. T. Ideker, *Nat Biotechnol*. **22**(4), 473 (2004)
2. S. H. Strogatz, *Nature.* **410**, 268 (2001)
3. S. Wasserman and K. Faust, *Social Networks Analysis*, Cambridge University Press, Cambridge (1994)
4. S.N. Dorogovtesev and J. F. F. Mendes, *Evolution of networks*, Oxford University Press, (2003)
5. H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai ZN and A. L. Barabasi, *Nature*. **407**(6804), 651 (2000)
6. H. Jeong, S. P. Mason, A. L. Barabasi and Z. N. Oltvai, *Nature*. **411**(6833), 41 (2001)
7. S. A. Wagner and D. A. Fell, *Proc. R. Soc. London*. **B268**, 1803 (2001)
8. S. Maslov and K. Sneppen, *Science*. **296**(5569), 910 (2002)
9. R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii and U. Alon, *Science*. **298**(5594), 824 (2002)
10. A. Vazquez, R. Dobrin, D. Sergi, J. P. Eckmann, Z. N. Oltvai, A. L. Barabási, *Proc Natl Acad Sci U S A*. **101**(52), 17940 (2004)
11. R. Sharan and T. Ideker, *Nat Biotechnol*. **24**(4), 427 (2006)
12. K. Basso, A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera and A. Califano, *Nat Genet*. **37**(4), 382 (2005)
13. J. J. Faith, B. Hayete, J. T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. J. Collins and T. S. Gardner, *PLoS Biology*. **5**(1) (2007).
14. J. Pearl, *Probabilistic reasoning in intelligent system: networks of plausible inference*, San Francisco, CA, Morgan Kaufmann Publishers, Inc. (1988)
15. V. Latora and M. Marchiori, *Physical Review E Statistical, Nonlinear and Soft Matter Physics*. **71**(1Pt2), 015103 (2005)
16. L. Agnelli, S. Bicciato, M. Mattioli, S. Fabris, D. Intini, D. Verdelli, L. Baldini, F. Morabito, V. Callea, L. Lombardi and A. Neri, *Journal of Clinical Oncology*. **23**(29), 7296 (2005)
17. F. Ferrari, S. Bortoluzzi, A. Coppe, A. Sirota, M. Safran, M. Shmoish, S. Ferrari, D. Lancet, G. A. Danieli and S. Bicciato, *BMC Bioinformatics*. **8**, 446 (2007)
18. J. Schug, W. P. Schuller, C. Kappen, J. M. Salbaum, M. Bucan and C. J. Stoeckert Jr, *Genome Biol*. **6**(4), R33 (2005)
19. P. Holme, B. J. Kim, C. N. Yoon and S. K. Han, *Phys Rev E*. **65**, 056109 (2002)
20. R. Albert, I. Albert, and G. L. Nakarado, *Phys. Rev. E*. **69**, 025103 (2004)
21. A.E. Motter and Y. Lai, *Phys. Rev. E*. **66**, 065102 (2002)
22. V. Latora and M. Marchiori, *Phys. Rev. Lett.* **87**, 198701 (2001)
23. V. Latora and M. Marchiori, *Chaos, Solitons and Fractals*. **20**, 69 (2004)

24. J. C. Huang, Q. D. Morris and B. J. Frey, *J Comput Biol*. **14**(5), 550 (2007)
25. V. A. Gennarino, M. Sardiello, R. Avellino, N. Meola, V. Maselli, S. Anand, L. Cutillo, A. Ballabio and S. Banfi, *Genome Res*. **19**(3), 481 (2009)
26. F. Xin, M. Li, C. Balch, M. Thomson, M. Fan, Y. Liu, S. M. Hammond, S. Kim and K. P. Nephew, *Bioinformatics*. **25**(4), 430 (2009)
27. B. John, A. J. Enright, A. Aravin, T. Tuschl, C. Sander and D. S. Marks, *PLoS Biol*. **2**(11), e363 (2004)
28. P. Crucitti, V. Latora, M. Marchiori and A. Rapisarda, *Physica A*. **340**, 388 (2004)
29. W. J. Chng, S. Kumar, S. Vanwier, G. Ahmann, T. Price-Troska, K. Henderson, T. H. Chung, S. Kim, G. Mulligan, B. Bryant, J. Carpten, M. Gertz, S. V. Rajkumar, M. Lacy, A. Dispenzieri, R. Kyle, P. Greipp, P. L. Bergsagel and R. Fonseca, *Cancer Research*. **67**(7), 2982 (2007)
30. J. Nakayama, M. Yamamoto, K. Hayashi, H. Satoh, K. Bundo, M. Kubo, R. Goitsuka, M. A. Farrar and D. Kitamura, *Blood*. **113**(7), 1483 (2009)
31. M. Merkerova, M. Belickova and H. Bruchova, *H. Eur J Haematol*. **81**(4), 304 (2008)
32. H. Zhao, A. Kalota, S. Jin and A. M. Gewirtz, *Blood*. **113**(3), 505 (2009)
33. Y. Iturria-Medina, R. C. Sotero, E. J. Canales-Rodríguez, Y. Alemán-Gómez and L. Melie-García, *Neuroimage*. **40**(3), 1064 (2008)
34. A. P. Potapov, B. Goemann, E. Wingender, *BMC Bioinformatics*. **9**:227 (2008)
35. F. Li, T. Long, Y. Lu, Q. Ouyang, C. Tang, *Proc Natl Acad Sci U S A*. **101**(14), 4781 (2004)
36. F. Emmert-Streib, M. Dehmer, *BMC Systems Biology*. **3**, 35 (2009)
37. A. L. Barabasi and R. Albert, *Science*. **286**(5439), 509 (1999)
38. R. Albert, *J Cell Sci*. **118**, 4947 (2005)
39. E. Almaas, *J Exp Biol*. **210**(Pt 9), 1548 (2007)

A COMPUTATIONAL MODEL OF GENE EXPRESSION IN AN INDUCIBLE SYNTHETIC CIRCUIT

FRANCESCA CERONI

Department of Electronics, Computer Sciences and Systems (DEIS)
Cellular and Molecular Engineering Laboratory, University of Bologna, via Venezia 52 Cesena, 47023, ITALY

SIMONE FURINI

Department of Medical Surgery and Bioengineering
University of Siena, viale Mario Bracci 12, Siena, 53100, Italy

SILVIO CAVALCANTI

Department of Electronics, Computer Sciences and Systems (DEIS)
Cellular and Molecular Engineering Laboratory, University of Bologna, via Venezia 52 Cesena, 47023, ITALY

Abstract. Synthetic biology aims to the rational design of gene circuits with predictable behaviours. Great efforts have been done so far to introduce in the field mathematical models that could facilitate the design of synthetic networks. Here we present a mathematical model of a synthetic gene-circuit with a negative feedback. The closed loop configuration allows the control of transcription by an inducer molecule (IPTG). *Escherichia coli* bacterial cells were transformed and expression of a fluorescent reporter (GFP) was measured for different inducer levels. Computer model simulations well reproduced the experimental induction data, using a single fitting parameter. Independent genetic components were used to assemble the synthetic circuit. The mathematical model here presented could be useful to predict how changes in these genetic components affect the behaviour of the synthetic circuit.

1. Introduction

Synthetic Biology is a novel discipline defined as the engineering of biology, i.e. the synthesis of systems based on biological material, which display prefixed functions. This engineering perspective may be applied at all hierarchical levels of biology, from individual molecules to whole cell, tissues and organisms (1). A major focus of the discipline is the synthesis of genetic components and gene circuits with predictable behaviours (2), either to endow cells with novel functions, or to study analogous natural systems (3). In the last years many gene circuits have been developed to achieve a fine regulation of gene expression and protein synthesis (4-6). Among them, auto-regulated networks with negative feedback loops have been invoked as a way to control and decrease transcriptional noise (7), conferring stability to gene expression, and high sensitivity to induction by an extracellular stimulus (8). A crucial element in these networks is the presence of promoter sequences with one or more operator sites that can be recognized by regulatory molecules, the transcription factors (TF). The position of the operator sequences inside the promoter region has a strong impact on how transcription factors control gene-transcription. Notably, the presence of an operator downstream of the TATA box region is responsible for transcription repression (9). This property can be used to assemble synthetic promoters that are repressed by the same TF, but which have different constitutive transcriptional strengths. Here we present a computational study of a synthetic device where gene expression is controlled by regulated promoters. The device includes two parts (Fig. 1): (i) an auto-regulated

generator of the LacI repressor protein (LacI-supplier); and (ii) a LacI-inverter, which uses a GFP protein as reporter. The mathematical model describes the dynamical interactions between these parts and accurately reproduces the device response to IPTG induction.

2. Materials and Methods

2.1.*Gene circuit scheme*

The LacI and GFP genes were placed both under the control of a synthetic promoter repressed by the LacI protein (Fig. 1). The promoters were designed by assembling the natural Lac operator sequence O2 downstream of two constitutive promoters (P_{LacI} and P_{GFP}), with the P_{GFP} transcriptional strength greater than the P_{LacI} one. The LacI repressor protein can bind to the operator site O_2 preventing the binding of DNA-polymerase to the P_{LacI} promoter. As a result, the LacI transcription is auto-regulated by a negative feedback. The LacI protein amount produced by the LacI-supplier also controls the GFP protein transcription by binding to the operator site O_2 placed downstream of the constitutive promoter P_{GFP}. When the LacI molecule concecentraion increases, the GFP reporter decreases, thus this part was called the LacI-inverter. The LacI-supplier was cloned in a medium copy number plasmid, while the LacI-inverter was cloned in a high copy number plasmid. This allows the amplification of the GFP reporter. The inducer molecule, IPTG (Isopropyl β-D-1-tiogalattopiranoside), inhibits the repressor activity for the operator site O_2 enhancing the transcription rate of the GFP protein. The inducer is considered as the external input of the device and regulates GFP expression.

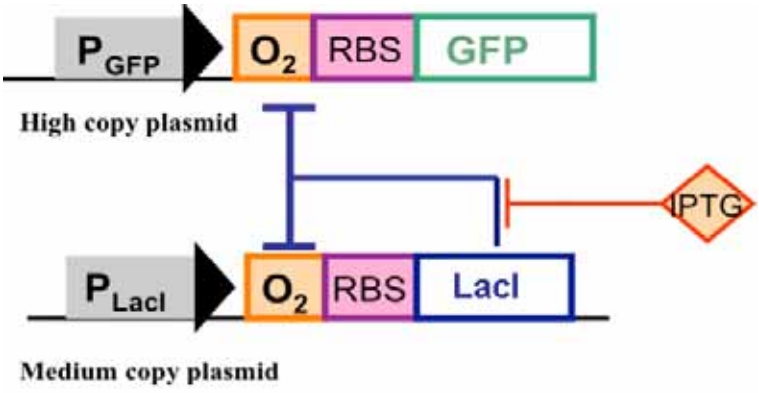


Figure 1 Gene network scheme

2.2.*BioBricks and Bacterial strain*

The constitutive promoters P_{GFP} (BBa_J23100) and P_{LacI} (BBa_J23118) were taken from the Registry of standard biological parts (10, 11) and were both cloned upstream of the same natural Lac operator sequence O2 (aaatgtgagcgagtaacaacc). The GFP gene with a degradation tag was

placed under the control of the P_{GFP} regulated promoter on a high copy number plasmid (pSB1A2) with Ampicillin resistance and a pUC19-derived pMB1 replication origin. The Lac repressor coding sequence with a degradation tag was placed under the control of the P_{LacI} regulated promoter on a medium copy number plasmid (pSB3K3) with Kanamycin resistance and a p15A pMR101-derived replication origin. The two vectors were co-transformed in *Escherichia coli* Dh5α cells (closed loop configuration) and grown under a two antibiotic selection. The same cells were also transformed with the LacI-inverter plasmid only (open loop configuration).

2.3.*Growth conditions and media*

Cells were grown in flasks at 37 °C in M9 minimal medium with glucose as the main carbon source, supplemented with casamino acids, thiamine and the proper antibiotics. Cells were induced for GFP expression by overnight growth with different IPTG concentrations (1 μM, 2 μM, 6 μM, 10 μM, 60 μM, 75 μM, 100 μM), while cultures with uninduced cells were prepared by overnight growth in the absence of IPTG. During each experimental run, after overnight growth, 100 μl from each cell culture were transferred into a multiwell plate for fluorescence analysis with the Victor 2 plate reader (Perkin Elmer). Wells on the plate edges were filled with fresh M9 medium in order to close off the samples from the external environment and to limit the thermal dispersion. For each well both fluorescence and optical density were measured. The green fluorescence protein values were estimated by normalizing the total fluorescence on the corresponding optical density.

2.4.*Mathematical Model*

The mathematical model describes gene expression from two independent co-transformed plasmids. The subscript R and G will be used for variables related to the LacI repressor and the the GFP reporter, respectively. Each plasmid can switch among three functional states: (i) bound to RNA-polymerase; (ii) bound to repressor protein; (iii) and free. The conservation of plasmid number in a single cell gives:

$$\begin{aligned} N_R &= L_R + R_R + P_R \\ N_G &= L_G + R_G + P_G \end{aligned}, \qquad (1)$$

where N_R and N_G are the plasmid numbers in each cell for the medium-copy and low-copy respectively; P_R, R_R, L_R are the number of plasmids bound to RNA polymerase, to repressor protein or free. When a free plasmid binds to a RNA polymerase molecule (P), an activated complex (P_R, P_G) forms that can initiate transcription and mRNA generation. The following differential equations describe the process:

$$\begin{aligned}\frac{dP_R}{dt} &= k_{RP}^f L_R P - \left(k_{RP}^b + \alpha_{RNA}\right)P_R = \frac{1}{\tau_{P_R}}\left(\frac{L_R P}{K_{RP}} - P_R\right)\\ \frac{dP_G}{dt} &= k_{GP}^f L_G P - \left(k_{GP}^b + \alpha_{RNA}\right)P_G = \frac{1}{\tau_{P_G}}\left(\frac{L_G P}{K_{GP}} - P_G\right)\end{aligned}. \quad (2)$$

Tables 1 and 2 summarize the variables and the parameters used in the mathematical model. The second expression of the differential equations was preferred because it highlights the time-constant of the process. We hypothesized that the number of polymerase molecules available for heterologous transcription is constant, which gives:

$$P_0 = P + P_R + P_G, \quad (3)$$

where P_0 is the total number of RNA-polymerase molecules. mRNA molecule synthesis and degradation were described by first order equations:

$$\begin{aligned}\frac{dM_R}{dt} &= \alpha_{RNA\_R}P_R - \beta_{RNA\_R}M_R = \frac{1}{\tau_{RNA\_R}}\left(M_{R,\infty}P_R - M_R\right)\\ \frac{dM_G}{dt} &= \alpha_{RNA\_G}P_G - \beta_{RNA\_G}M_G = \frac{1}{\tau_{RNA\_G}}\left(M_{G,\infty}P_G - M_G\right)\end{aligned}. \quad (4)$$

Equations analogous to (2), (3) and (4) were used to describe the binding of the mRNA molecules M_R and M_G, to ribosomes and the synthesis of the two proteins R and G:

$$\begin{aligned}\frac{dB_R}{dt} &= k_{RB}^f M_R B - \left(k_{RB}^b + \alpha_{PROT}\right)B_R = \frac{1}{\tau_{B_R}}\left(\frac{M_R B}{K_{RB}} - B_R\right)\\ \frac{dB_G}{dt} &= k_{GB}^f M_G B - \left(k_{GB}^b + \alpha_{PROT}\right)B_G = \frac{1}{\tau_{B_G}}\left(\frac{M_G B}{K_{GB}} - B_G\right)\end{aligned}, \quad (5)$$

$$\begin{aligned}\frac{dR_T}{dt} &= \alpha_{PROT\_R}B_R - \beta_{PROT\_R}R_T = \frac{1}{\tau_{PROT\_R}}\left(P_{R,\infty}B_R - R_T\right)\\ \frac{dG}{dt} &= \alpha_{PROT\_G}B_G - \beta_{PROT\_G}G = \frac{1}{\tau_{PROT\_G}}\left(P_{G,\infty}B_G - G\right)\end{aligned}. \quad (6)$$

$$B_0 = B + B_R + B_G. \quad (7)$$

Note that in equations (6) the symbol R_T is used for the repressor in place of R. R_T is the number of all the repressor molecules, while R is the number of free repressors, R_G is the number of repressors bound to the high-copy number plasmids, R_R the repressors bound to the low-copy number plasmids and R_I the repressors bound to inducer molecules. Thus:

$$R_T = R + R_I + R_R + R_G. \quad (8)$$

The binding process between the free repressor, R, and the inducer, I, was described by the equation:

$$\frac{dR_I}{d} = k_{RI}^f RI^n - k_{RI}^b R_I = \frac{1}{\tau_I}\left(\frac{RI^n}{K_I} - R_I\right), \quad (9)$$

where n is the binding cooperativity.

Finally, the following equations complete the mathematical model describing the free repressor binding to the operator site O_2 on the plasmids:

$$\begin{aligned}\frac{dR_R}{dt} &= k_{RR}^f L_R R - k_{RR}^b R_R = \frac{1}{\tau_{R_R}}\left(\frac{L_R R}{K_{RR}} - R_R\right)\\ \frac{dR_G}{dt} &= k_{GR}^f L_G R - k_{GR}^b R_G = \frac{1}{\tau_{R_G}}\left(\frac{L_G R}{K_{GR}} - R_G\right)\end{aligned} \quad (10)$$

Model equations were implemented in Simulink (Mathworks), and were numerically integrated starting from null initial conditions for the state variables. Model parameters describe well-characterized physical processes involving standard biological molecules, thus parameter values were taken from the literature (see Table 2 for details). The only parameter used to fit the experimental data was the repressor-operator binding constant. This was identified by fitting experimental data on GFP production, measured as fluorescence in the cell population. Parameter identification was performed by the Matlab routine fminsearch.

Table 1. Model variables

Symbol	Definition
$L_{R/G}$	Free plasmids, i.e. not bound to Dna-polymerase or repressor
$P_{R/G}$	Plasmids bound to Dna-polymerase
P	Free polymerase, i.e. not bound to any plasmid
$M_{R/G}$	mRNA
$B_{R/G}$	Ribosomes bound to $M_{R/G}$
B	Free ribosomes, i.e. not bound to any mRNA molecule
$R_{R/G}$	Plasmids bound to the repressor
R	Free repressor, LacI
R_I	IPTG bound repressor
R_T	Total repressor molecules, free and bound
G	Reporter protein, GFP (green fluorescence protein)

Table 2. Model parameters

Symbols	Definition and Comments	Value
N_R	Number of medium-copy plasmids	15 (11)
N_G	Number of high-copy plasmids	150 (10)

P_0	Number of polymerase molecules		1500 (12)
B_0	Number of ribosomes		6800 (12)
α_{PROT_R}	LacI synthesis rate. The parameter was calculated as the peptide chain elongation rate multiplied by the number of LacI residues.		2.85 min^{-1} (12)
α_{PROT_G}	GFP synthesis rate. The parameter was calculated as the peptide chain elongation rate multiplied by the number of GFP residues.		1.92 min^{-1} (12)
α_{RNA_R}	LacI mRNA molecule synthesis rate. The parameter was calculated as the mRNA elongation rate multiplied by the number of nucleotides in LacI.		3.17 min-1 (13)
α_{RNA_G}	GFP mRNA molecule synthesis rate. The parameter was calculated as the mRNA elongation rate multiplied by the number of nucleotides in GFP.		4.63 min-1 (13)
$\beta_{PROT\_R/G}$	Protein degradation rate. This parameter was estimated through experimental measurements of GFP degradation rate. Analogous degradation rate was assumed for the LacI protein.		$2.13*10^{-2}$ min^{-1}
$\beta_{RNA\_R/G}$	mRNA degradation rate		0.19 min^{-1} (14)
$k_{GB}^{f/b}$	Forward and backward kinetic rates for the reaction between the reporter gene mRNA molecules the and ribosomes		
$k_{RB}^{f/b}$	Forward and backward kinetic rates of the reaction between the repressor protein mRNA molecules and the ribosomes		
$k_{GR}^{f/b}$	Forward and backward kinetic rates of the reaction between free repressor molecules and the operator O_2 on the high-copy number plasmid		
$k_{RR}^{f/b}$	Forward and backward kinetic rates of the reaction between free repressor molecules and the operator O_2 on the low-copy number plasmid		
$k_{RP}^{f/b}$	Forward and backward kinetic rates of the reaction between DNA polymerase molecules and the promoter on the low-copy number plasmids		
$k_{GP}^{f/b}$	Forward and backward kinetic rates of the reaction between DNA polymerase molecules and the promoter on the high-copy number plasmids		
$k_{RI}^{f/b}$	Forward and backward kinetic rates of the reaction between free repressor and inducer molecules		
$\tau_{P_{R/G}}$	$\frac{1}{k_{R/GP}^{b} + \alpha_{RNA\_R/G}}$		$1.7*10^{-3}$ min (8)
K_{RP}	$\frac{k_{RP}^{b} + \alpha_{RNA\_R}}{k_{RP}^{f}}$		$1.5*10^{4}$ (15)
K_{GP}	$\frac{k_{GP}^{b} + \alpha_{RNA\_G}}{k_{GP}^{f}}$	The ratio between GFP expression with the two promoters, P_{GFP} and P_{LacI}, was determined experimentally as 1.4. The same ratio was assumed between K_{RP} and K_{GP}	$0.7*10^{4}$
$\tau_{RNA\_R/G}$	$\frac{1}{\beta_{RNA\_R/G}}$		5.32 min

$M_{R,\infty}$	$\frac{\alpha_{RNA\_R}}{\beta_{RNA\_R}}$	16.9
$M_{G,\infty}$	$\frac{\alpha_{RNA\_G}}{\beta_{RNA\_G}}$	24.6
$\tau_{B_{R/G}}$	$\frac{1}{k^b_{R/GB} + \alpha_{PROT\_R/G}}$	$7.4*10^{-3}$ min (13)
$K_{R/GB}$	$\frac{k^b_{R/GB} + \alpha_{PROT\_R/G}}{k^f_{R/GB}}$	$4.4*10^{-3}$ (13)
$\tau_{PROT\_R/G}$	$\frac{1}{\beta_{PROT\_R/G}}$	47 min
$P_{R,\infty}$	$\frac{\alpha_{PROT\_R}}{\beta_{PROT\_R}}$	90
$P_{G,\infty}$	$\frac{\alpha_{PROT\_G}}{\beta_{PROT\_G}}$	134
n	IPTG inducer binding cooperativity	2
τ_I	$\frac{1}{k^f_{RI}}$	22.4 min (16)
K_I	$\frac{k^b_{RI}}{k^f_{RI}}$	1680 (17)
$\tau_{R_{R/G}}$	$\frac{1}{k^b_{G/RR}}$	$1*10^{-3}$ min
$K_{R/GR}$	$\frac{k^b_{RR}}{k^f_{RR}}$ This is the only parameter of the mathematical model adjusted to fit the experimental data.	0.3

3. Results

Figure 2 shows how GFP transcription can be modulated using different IPTG concentrations. A good agreement between the mathematical model and the experimental data was obtained by tuning the parameter K_{RR}, which models the binding affinity of the repressor for the operator on the medium-copy number plasmid. Since the same operator sequence was used both in the high copy number and low copy number plasmid, parameters K_{RR} and K_{GR} were assumed equal. The

fluorescence observed at different IPTG concentrations was divided by the fluorescence measured in the open loop configuration (bacteria transformed only with the LacI-inverter circuit), thus defining a normalized measurement of GFP expression levels.

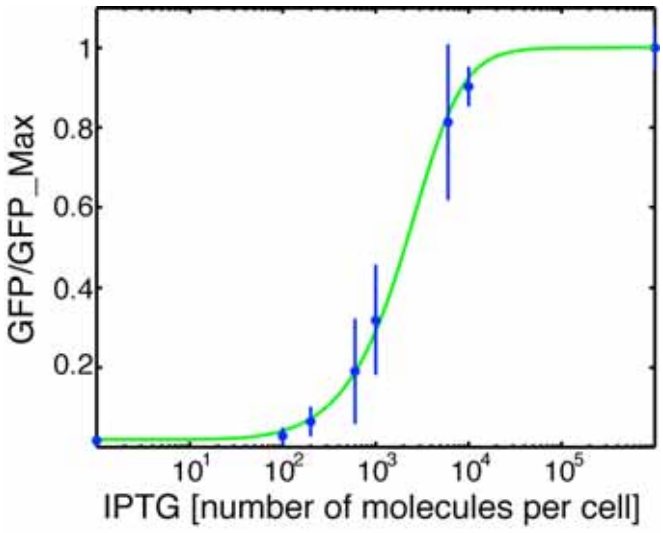


Figure 2 Dose Response curve. GFP expression was normalized to the value in the open-loop gene circuit, both for the experimental data (blue points, bars for standard deviation) and for the results of the numerical simulations (green line).

To simulate the regulated promoter response to changes in the LacI repressor level (Fig. 3), we forced the model by opening the negative feedback and using R as an input. R was slowly increased from 0 to 1000 molecules and the quasi-static level of the reporter protein G was computed. Simulation shows that the regulated promoter is repressed at 50% by ~100 molecules of LacI (see Fig. 3). This value closely agrees with the experimental determination of LacI repressor binding constant on the P_{Lac} promoter (70 molecules).

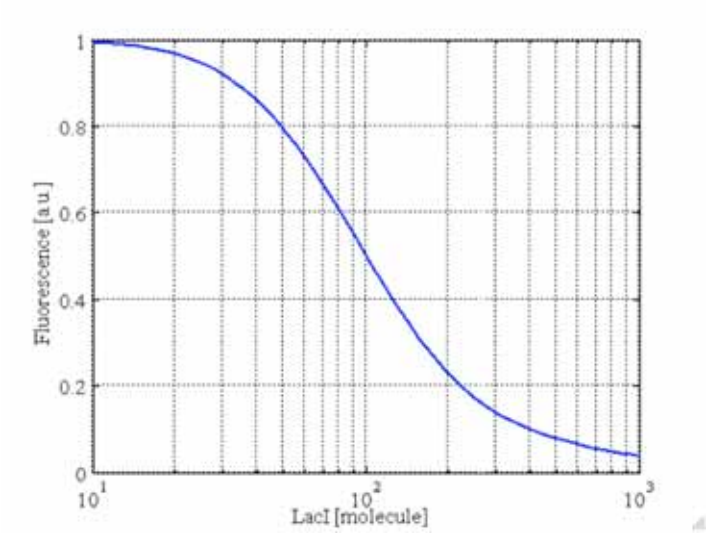


Figure 3. Fluorescence decresing in presence of the LacI repressor.The 50% of repression is achieved with 100 repressor molecule.

To characterize the dynamic response of the device we considered the rise-time, (i.e. the delay between the initial and half maximal production). The rise-time was estimated by simulating the transient of the gene-circuit from null initial conditions ($M_R = R = 0$ and $M_G = G = 0$), in both open and close loop configurations (Fig. 4). In open-loop, when the GFP promoter is unregulated, the circuit reaches the steady state after 180 min, while in close-loop there is an initial over-shoot at 20 min due to the latency in the R protein synthesis. After 80 min the transient is extinct.

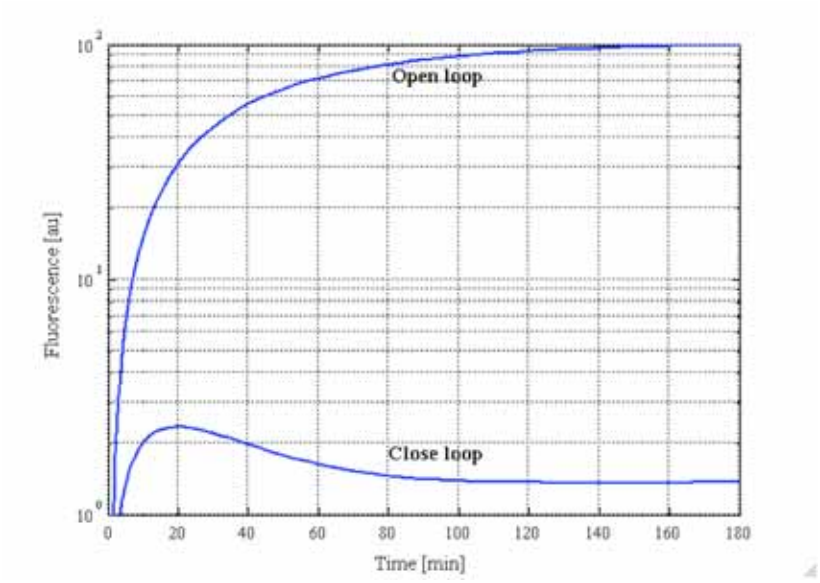


Figure 4. Dynamic response of close- and open-loop configurations. Open loop refers to the cells transformed with the Lac-inverter plasmid only, while closed loop configuration refers to cells co-transformed with both LacI-supply and the LacI-inverter plasmids

4. Discussion

In recent years an increasing number of studies in synthetic biology have been focused on constructing simple synthetic gene circuits that exhibit desired properties. In this work, we used a gene network with a negative feedback, where the Lac repressor protein (LacI) is responsible for the repression of GFP transcription from a strong promoter on a high copy number plasmid and, at the same time, it prevents its own transcription from a weak promoter on a medium copy number plasmid, auto-regulating itself. Since the usage of the same biological parts in another synthetic gene network would be greatly facilitated by their mathematical modelling, we performed an analytical analysis of the gene circuit. After the model formulation, we have cloned the device in *Escherichia coli* to compare the experimental measurements with the computer model prediction. A good fitting of the experimental data was obtained (see Fig. 2) only by tuning the LacI repressor affinity to the operator site, whereas the other model parameters were assigned according to the literature. Notably, variation of the circuit genetic components can be simulated *in-silico* by changing the appropriate model parameter. As an example, after model identification we computed the LacI-inverter response to LacI repressor (Fig. 3). This curve can be used to establish the sensitivity of the LacI-inverter to LacI changes in the region with negative slope (-4 % for each 10 LacI molecules) as well as the amount of LacI molecules (10^3) needed to get a 95% GFP repression. Moreover, the mathematical model can be used to identify which modifications should be introduced in the circuit to obtain a specific sensitivity to the inducer molecules or to change the parameter values to predict the protein expression levels from promoters with different transcriptional strengths.

We used the model to compute the dynamic behaviour of the device through time. To characterize the dynamics we considered the rise-time, (i.e. the delay from the initiation of production until half maximal product concentration is reached). According to previous observation (18), the rise-time in negatively auto-regulated transcription circuits was shorter than in non-regulated transcription. Negative feedback (also termed autogenous control) reduces the rise-time. This may help in understanding the function of negative auto-regulation, which appears in over 40% of known transcription factors in E. coli.

Mathematical modelling could prove to be useful in the design of synthetic gene networks. The behaviour of a gene network is highly affected by the transcription rates of the involved genes. The optimization of these transcription rates is usually a time-consuming task, which is a severe obstacle for a further evolution of synthetic biology. The gene circuit presented here does not only offer the possibility to tune gene expression in a predictable way by controlling the concentration of an extracellular inducer molecule, but it also exhibits a modular architecture, which is particularly useful for the control of gene transcription in synthetic biology. Different plasmids types, operator sequences, promoters, can be combined to obtain a set of circuits, whose characteristics can be described, and eventually predicted, with the same mathematical model, with a typically engineering approach.

5. References

1. Serrano, L., Mol Syst Biol, **3**:158 (2007).
2. Guido, N.J., et al., Nature, **439**(7078):856-860 (2006).
3. Ellis, T., et al., Nat Biotechnol, **27**(5):465-71 (2009).
4. Elowitz, M.B., et al., Nature, **403**(6767):335-338 (2000).
5. Gardner, T.S., et al., Nature, **403**(6767):339-342 (2000).
6. Atkinson, M.R., et al., **113**(5):597-607 (2003).
7. Dublanche, Y., et al., Mol Syst Biol, **2**:41 (2006).
8. Stricker, J., et al., Nature, **456**(7221):516-519 (2008).
9. Cox, R., et al., Mol Syst Biol, **3** (2007).
10. Knight, T. *http://partsregistry.org/Part:pSB1A2*. (2004).
11. Shetty, R. *http://partsregistry.org/Part:pSB3K3*. (2004).
12. Bremer, H., et al., *Modulation of chemical composition and other parameters of the cell by growth rate*, in *Escherichia coli and Salmonella typhimurium: Cellular and Molecular Biology*, Neidhardt, Editor. (1996)
13. Epshtein, V., et al., Science, **300**(5620):801-805 (2003).
14. Bernstein, J.A., et al., Proc Natl Acad Sci U S A, **101**(9):2758-63 (2004).
15. Bintu, L., et al., Current Opinion in Genetics & Development, **15**(2):116-124 (2005).
16. Xie, X.S., et al., Annual Review of Biophysics, **37**(1):417-444 (2008).
17. Wilson, C.J., et al., Biophysical Chemistry, **126**(1-3):94-105 (2007).
18. Rosenfeld, N., et al., J Mol Biol, **323**(5):785-93 (2002).

RETROVIRUS HTLV-1 GENE CIRCUIT: A POTENTIAL OSCILLATOR FOR EUKARYOTES

ALBERTO CORRADIN[1], BARBARA DI CAMILLO[1], FRANCESCA RENDE[2],
VINCENZO CIMINALE[2], GIANNA MARIA TOFFOLO[1], CLAUDIO COBELLI[1]
[1]*Department of Information Engineering, University of Padua*
via Gradenigo 6, Padua, 35131, Italy
[2]*Department of Oncology and Surgical Sciences, University of Padua*
Via Gattamelata 64, Padua, 35128, Italy

Retrovirus HTLV-1 gene circuit is characterized by positive and negative feedback phenomena, thus candidating it as a potential relaxation oscillator deliverable into eukaryotes. Here we describe a model of HTLV-1 which, by providing predictions of genes and proteins kinetics, can be helpful for designing gene circuits for eukaryotes, or for optimizing gene therapy approaches which are currently carried out by means of lentiviral vectors or re-engineered adenoviruses. Oscillatory patterns of HTLV-1 gene circuit are predicted when positive feedback is faster than negative feedback. Techniques to mutate the retroviral genome in order to implement practically the above conditions are discussed. Finally, the effect of stochasticity on the system behavior is tested by means of Gillespie algorithm. Simulations show the difficulties to preserve synchronization in viral expression for a multiplicity of cells, while the long tail of the density probability function of the master regulator gene *tax/rex*, due to its steady state fluctuations, suggests an activation mechanism of HTLV-1 similar to that recently proposed for HIV[1]: the virus tends to latency but under certain circumstances, the master regulator gene reaches high values of expression, whose persistence induces the viral replication.

1. Introduction

Synthetic gene circuits have already been delivered into bacterial cells, proving that it is possible to design and implement synthetic biological systems. A genetic toggle switch, designed by Gardner et al.[2] in 2000, was soon followed by the gene oscillator of Elowitz and Leibler[3]. An essential ingredient of this last work was the preliminary circuit characterization by modeling gene and protein expression; in particular, bifurcation analysis has allowed identifying the ranges of parameter values corresponding to periodic patterns. Nevertheless, the modeling strategy must be sound; otherwise misleading results can be obtained, which can heavily affect the subsequent circuit design. Recently, Kaern et al.[4] supported the use of systems of differential equations based on mass action, i.e. the approach previously adopted by Hasty et al.[5,6] for modeling the λ phage. In moving from bacteria to eukaryotes, the design of gene circuits becomes more difficult because of the more complex regulatory mechanisms. However, significant contributions have become available, e.g. Ramachandra et al.[7] re-engineered adenoviruses to hit tumor cells selectively, i.e. without impairing the healthy cells; Bainbridge et al.[8] addressed the Leber's Congenital Amaurosis by means of recombinant adeno-associated virus vectors. Also, of note is that mathematical models of viral kinetics are potentially valuable for optimizing gene therapy approaches, in particular by improving the design of retroviral vectors by predicting the gene expression following their delivery. In this paper we propose a novel model of HTLV-1 viral kinetics[9]. This model is characterized by positive and negative feedback phenomena, similarly to synthetic relaxation oscillators delivered into prokaryotes and able to exhibit limit cycles. In order to investigate the potential use of HTLV-1 circuit as a novel oscillator for eukaryotes, we analyze the periodic behavior of HTLV-1 model. This represents a preliminary step that can be instrumental for designing gene circuits to be delivered into eukaryotic cells, or for optimizing retroviral vector design in gene therapy.

Results previously obtained with bacteria suggest that a deterministic model may be not adequate to predict the true behavior of a biological system, e.g. a remarkable variability was observed in the repressilator period of oscillation by Elowitz and Leibler[3], Thattai et al.[10] described the noise in transcription and translation whereas Elowitz et al.[11] experimentally highlighted the effects induced by variable quantities of metabolites in the single cells of the same sample. On the other hand, Gillespie[12] pointed out that if a system is small enough that the molecular populations of some reactant species are not too many orders of magnitude larger than one, discreteness and stochasticity may play important roles, so that the predictions coming from deterministic differential equations do not accurately describe the system's true behavior. This is the case for HTLV-1 kinetics since one of the basic mechanisms is transactivation, i.e. the enhancement of transcription caused by the interaction between the viral promoter and a viral protein, and the number of promoters in a cell corresponds to the number of viral genomes integrated in the host cell, which is small. Thus, stochastic simulations are important for an adequate understanding of the system behavior.

In the following, after presenting an HTLV-1 model, which provides deterministic predictions about viral kinetics, we will discuss first the periodic patterns revealed by the bifurcation analysis, and then, the results of stochastic simulations performed by the Gillespie algorithm. Finally, the practical feasibility of an oscillator for eukaryotes will be considered, together with some recent biotechnological techniques potentially helpful to mutate the HTLV-1 genome so as to obtain periodic oscillations of gene and protein expression.

2. The model

The main mechanisms of the HTLV-1 gene circuit[13] are summarized in Figure 1:

1. The full-length genomic RNA of the single stranded retrovirus HTLV-1 encodes for the primary transcript *gag* (compartment 1) which undergoes either single or double splicing in the nucleus or, alternatively, remains unspliced.
2. The doubly spliced mRNA *tax/rex* (compartment 2) is considered the master regulator of viral gene expression since it encodes for two distinct regulatory proteins, p40*Tax* and p27*Rex* (compartments 3 and 4), from ORF III and IV, respectively; in the following we will refer to them simply as *Tax* and *Rex*.
3. *Tax* boosts the transcription of the primary transcript *gag*, generating a positive feedback phenomenon called transactivation.
4. *Rex* prevents the multiple splicing of *gag*, causing a decrease in the amount of *tax/rex* in favor of unspliced and singly spliced genes, and generating a negative feedback phenomenon with respect to *tax/rex*.
5. A variety of *Rex*-dependent viral genes deriving from the single splicing of *gag* were identified, e.g. *1-B*, *p13*, p21Rex (compartments 5.1, 5.2… 5.n), but the mechanisms of splitting up are still unclear.

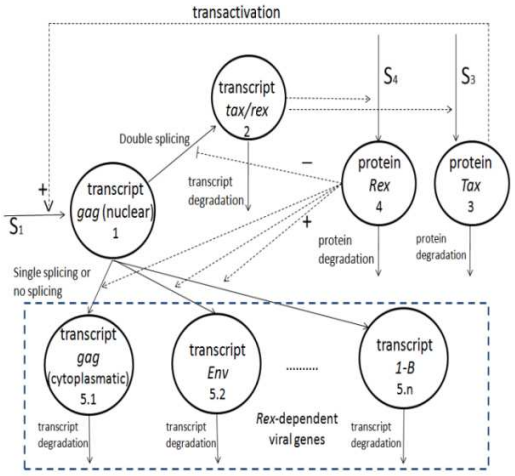


Figure 1. Background knowledge on the HTLV-1 gene circuit. Solid arrows represent fluxes, dashed arrows controls.

The system of differential equations which describe the above mechanisms, based on one-step reactions and mass action (detailed in the Appendix), is:

$$\begin{cases} \frac{dq_1}{dt} = \mathrm{m\,S} + \mathrm{m}\,S\,\beta'_1\,\frac{\mathrm{q_3(t)^2}}{\mathrm{h_1^2}+\mathrm{q_3(t)^2}} - \mathrm{k_s}q_1(\mathrm{t}) & \mathrm{q_1(0)=0} \quad (1) \\ \frac{dq_2}{dt} = \{1 - \mathrm{g(q_1(t)}, q_4(\mathrm{t}), z)\}\,\mathrm{k_s}q_1(\mathrm{t}) - k_{02}\,q_2(\mathrm{t}) & \mathrm{q_2(0)=0} \quad (2) \\ \frac{dq_3}{dt} = \beta'_3 q_2(\mathrm{t}) - k_{03}\,q_3(\mathrm{t}) & \mathrm{q_3(0)=0} \quad (3) \\ \frac{dq_4}{dt} = \beta'_4 q_2(\mathrm{t}) - k_{04}\,q_4(\mathrm{t}) & \mathrm{q_4(0)=0} \quad (4) \end{cases}$$

where the state variables are the concentrations [molecules/l] of nuclear *gag*, *tax/rex*, *Tax*, *Re* corresponding to the compartments 1-4 of Figure 1. Initial conditions were set to zero, i.e. we supposed to deliver the HTLV-1 gene circuit into eukaryotic cells which were not infected previously. System parameters are: the transcription rate S [1/h], the concentration of viral genomes integrated in the host cell m [molecules/l], the transactivation constant β'_1 (adimensional), Michaelis constant h_1 (which is the product of many equilibrium constants, as described in the Appendix), the nuclear export rate k_s [1/h], the order of *Rex* multimerization z (adimensional), degradation rates of the transcript *tax/rex* k_{02} [1/h], of proteins *Tax* k_{03} [1/h] and *Rex* k_{04} [1/h], and parameters β'_3 and β'_4 [protein molecules/(transcript molecules*h)], which are the products of the *Tax* and *Rex* gains in protein translation multiplied for the rate constants of the translation processes (as detailed in the Appendix). The function g(·) is defined in Eq. 5-7:

$$\mathrm{g(q_1(t)}, q_4(t), z) = \begin{cases} \frac{\mathrm{q_4^z}}{\mathrm{h_3^z}}\frac{1}{\mathrm{q_1}} & \text{If } \frac{\mathrm{q_4^z}}{\mathrm{h_3^z}} < \mathrm{q_1} \text{ and } \mathrm{q_1} > 0 \quad (5) \\ 1 & \text{If } \frac{\mathrm{q_4^z}}{\mathrm{h_3^z}} > \mathrm{q_1} \text{and } \mathrm{q_1} > 0 \quad (6) \\ 0 & \text{If } \mathrm{q_1} = 0 \quad (7) \end{cases}$$

where h_3 is the product of the some equilibrium constants, as described in the Appendix. Unfortunately, only k_{01} was measured by Grone et al.[14] equal to 0.069/h. As concerns other parameters, approximate values can be derived from the literature, based on measurements of similar biological processes, reported below with the corresponding model parameters between brackets:

1. Rosin-Arbesfeld et al.[15] and Lewis[16] reported the half-lives of about 10 and 4 minutes, respectively, for the nucleo-cytoplasmatic transport (k_s).
2. Weinberger and Shenk[17] estimated the transactivation constant (β'_1) in the HIV gene circuit by fitting an ODE model to normalized data of fluorescence intensities from single-cell measurements, obtaining a value of about 8.
3. As regards protein degradation rates, the *Tax* ubiquitination was confirmed by many laboratories[18] supporting the thesis of a ubiquitin-mediated degradation; moreover, Peloponese et al.[19] proved the *Tax* inactivation induced by ubiquitin. Jeong et al.[20] observed similar decays for *Tax* and β-Galactosidase, whose half-life was estimated in about 13 hours[21,22]. Therefore, we considered a half-life of about 10 hours for *Tax*, and also for *Rex* (k_{03}, k_{04}).
4. Kugel and Goodrich[23] measured the transcription rate induced by polymerase II in eukaryotes (S): 1.9e-3/s.

As regards Michaelis constant h_1, no information is available as well as for the parameter h_3, thus the amount of *Tax* and *Rex* were scaled by h_1 and h_3, respectively: $\widetilde{q_3} = \frac{q_3}{h_1}$ and $\widetilde{q_4} = \frac{q_4}{h_3}$; $\widetilde{q_3}$ and $\widetilde{q_4}$ can be viewed as the effective proteins, present in the nucleus and effectively acting for transactivation and RNA nuclear export, respectively.

Moreover, to preserve the validity of the study front of future HTLV-1-specific measurements resulting in different values of the transcription rate, the time t [h] was scaled by S as done in Ref. 4: $\tau = t*S$; τ is adimensional because the unit of measurements of S is 1/h. After scaling, the model of differential equations became:

$$\begin{cases} \frac{dq_1}{d\tau} = m + m\,\beta'_1 \frac{\widetilde{q_3}(\tau)^2}{1+\widetilde{q_3}(\tau)^2} - k'_s q_1(\tau) & q_1(0)=0 \quad (8) \\ \frac{dq_1}{d\tau} = \{1 - \tilde{g}(q_1(\tau), \widetilde{q_4}(\tau), z)\}\, k'_s q_1(\tau) - k'_{02}\, q_2(\tau) & q_2(0)=0 \quad (9) \\ \frac{d\widetilde{q_3}}{d\tau} = \widetilde{\beta'}_3 q_2(\tau) - k'_{03}\widetilde{q_3}(\tau) & \widetilde{q_3}(0)=0 \quad (10) \\ \frac{d\widetilde{q_4}}{d\tau} = \widetilde{\beta'}_4 q_2(\tau) - k'_{04}\widetilde{q_4}(\tau) & \widetilde{q_4}(0)=0 \quad (11) \end{cases}$$

where the function $\tilde{g}(\cdot)$ is defined in Eq. 12-14:

$$\tilde{g}(q_1(t), \widetilde{q_4}(t), z) = \begin{cases} \frac{\widetilde{q_4}^z}{q_1} & \text{If } \widetilde{q_4}^z < q_1 \text{ and } q_1 > 0 \quad (12) \\ 1 & \text{If } \widetilde{q_4}^z > q_1 \text{ and } q_1 > 0 \quad (13) \\ 0 & \text{If } q_1 = 0 \quad (14) \end{cases}$$

Model parameters were fixed to the following nominal values: $\beta'_1=10$, $k'_{01}=k_{01}/S=0.01$, $k'_s=k_s/S=1$, $k'_{02}=k_{02}/S=0.01$, $k'_{03}=k_{03}/S=0.01$, all of which are adimensional. Parameter m was initially set equal to 1 molecule/l, to reflect the hypothesis of low multiplicity of infection, which underlies the model development (see the Appendix). As regards the parameters $\widetilde{\beta'}_3$ and $\widetilde{\beta'}_4$, since no information was available in the literature, computational simulations of the deterministic system (Eq. 8-11) were performed, and the parameter values were fixed so as to obtain gene expression time course consistent with some experimental measurements[24,25]. This happened for $\widetilde{\beta'}_3$ and $\widetilde{\beta'}_4$ in the range [1e-3, 1e-1], thus we chose the median value $\widetilde{\beta'}_3 = \widetilde{\beta'}_4 = 0.01$ as the default value to be used for the subsequent analyses. Since it was not possible to establish if *Rex* forms dimers (z=2), pentamers (z=5), or something else, different values of z were considered.

3. Bifurcation analysis

We tested if the model of the HTLV-1 gene circuit (Eq.8-11), can exhibit periodic patterns by performing bifurcation analysis with the MatCont software package[26]. Among the system parameters, some were considered tunable on the basis of experimental observations, i.e. the protein degradation rates and the concentration m of viral genomes integrated in the host cell, whereas other parameters were set to their default values. The rational is that *Tax* undergoes ubiquitination[18] and experimental observations support the tunability of the proteins half-life when their degradation involves the ubiquitin pathway[27]. As regards m, it can be easily regulated at the time of virus delivery following an estimation of the titer of viral particles.

3.1 Periodic patterns

With the default parameter values the system falls into a stable steady state. By varying the ratio between the two protein degradation rates, RD=k'_{03}/k'_{04}, two Hopf bifurcations were detected confirming the possibility for the HTLV-1 gene circuit to oscillate[28]; moreover, the periodic patterns were stable because both bifurcations are supercritical. With z=2 the critical values of RD and the Lyapunov coefficients were: RD_{H1}=2.3, RD_{H2}=26.2 with L_{H1}=-4.6e-3 and L_{H2}=-7.6e-4, whereas with z=5 they were: RD_{H1}=7.0, RD_{H2}=13.7 with L_{H1}=-1.5e-3 and L_{H2}=-7.6e-4. In Figure 2a, the trajectories of the state variables *gag* and *tax/rex* for z=2 and RD=3 are shown.

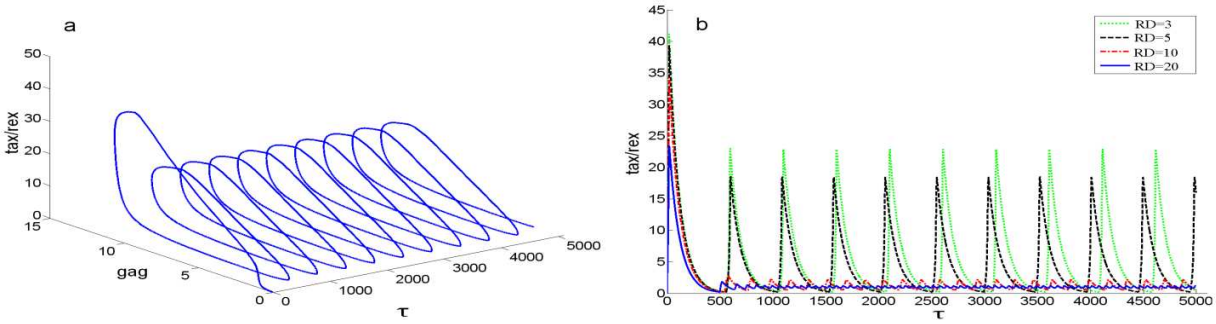


Figure 2. a) The trajectory of the system for z=2 and RD=3. b) Periodic patterns of *tax/rex* for z=2 and the following values of RD: 3,5,10,20.

Higher values of *Tax* degradation rate (with respect to *Rex*) result in smaller amplitudes and periods of oscillation, as shown in Figure 2b, where the periodic patterns of *tax/rex* corresponding to z=2 and RD=3,5,10,20, respectively, are plotted. If multiplicity of infection m is increased, periodic patterns arise if RD is within specific limits which depends on m, as shown in Figure 3a, for z=2 and z=5. This figure suggests the relevant role of z on system behavior. To have a better insight on the role of z, Hopf continuation was performed by varying RD and z, for specific m values. Results (Figure 3b) indicate that periodic oscillations of the model state variables are prevented for z>5.

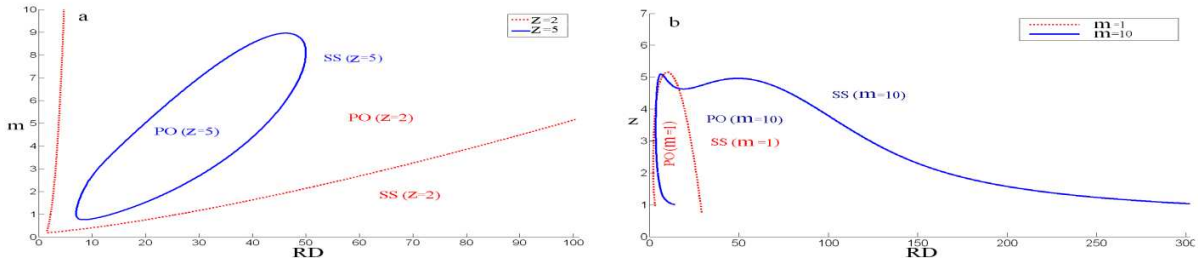


Figure 3. a) Hopf curve continuation with free parameters RD and m, for z=2 and 5. The areas corresponding to parameter settings allowing periodic oscillations are signaled with the symbol PO whereas the areas corresponding to parameter settings for which the system falls into steady states are signaled with the symbol SS. b) Hopf curve continuation with free parameters RD and z, for m=1 and 10.

4. Stochastic fluctuations

Since stochasticity and discreteness can cause deviations of the true system behavior from the predictions of deterministic differential equations when the molecular populations are small, as is the case of the viral promoter sites in our model, stochastic simulations were performed by Gillespie algorithm (direct method[29,30]), with the parameter settings corresponding to the periodic patterns and to the steady state solution of the system (Eq. 8-11). Gillespie algorithm describes the number X(t) of molecules of chemical species involved in the reactions R_j characterizing the system. The key to simulate trajectories of X(t) is the probability function $p(\tau, j \mid x,t)$[12], which is the probability, given X(t) = x, that the next reaction in the system will occur in the infinitesimal time interval $[t+\tau, t+\tau+d\tau)$, and will be an R_j reaction. This probability is related to the number of molecules composing the chemical species involved in the reaction as reactant. Thus, the variability concerns which reaction takes place and when this happens. Two software packages, providing distinct implementation of the algorithm, were considered: Dizzy[31] (ver 1.11.4) and Cain[32] (ver 0.12).

4.1 Periodic patterns

The effects of stochasticity on the periodic patterns of Figure 2b were verified by simulating 1000 trajectories of the state variables. Results evidenced different time points at which peaks of *tax/rex* rise in distinct realizations (see Figure 4a) causing a lack of synchronicity. Consequently, the *tax/rex* mean time course resulted to be leveled, as shown in panel b; moreover, a high variability in gene expression was observed (see panel c, where the standard deviations corresponding to each time point are reported). The leveling of the *tax/rex* mean time course appeared also for other values of z, as shown in panel d, where it is plotted the mean of 1000 realizations with parameter setting z=5 and RD=8, for which periodic oscillations arise in the deterministic system (Eq. 8-11). To verify if the leveling depended on m, the same simulations were repeated with m=10 and 100, instead of 1, but no better result was observed.

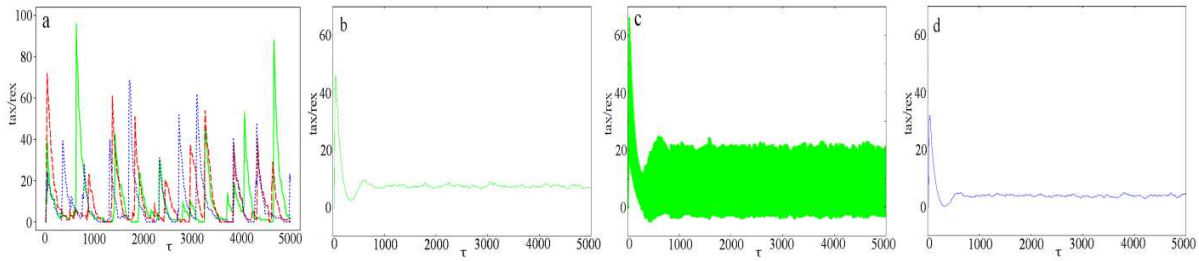


Figure 4. Stochasticity in chemical reactions causes the lack of synchronicity and the leveling of the *tax/rex* mean time course. a) 3 trajectories of the state variable *tax/rex*, b) the mean time course of 1000 realizations, and c) the standard deviations corresponding to each time point for z=2 and RD=3. d) The *tax/rex* mean time course of 1000 realizations for z=5 and RD=8.

4.2 Steady state fluctuations

Since the addition of stochastic phenomena on the periodic patterns provided surprising results, also stochastic fluctuations of the steady state solution were examined by means of 10000 stochastic simulations. All state variables resulted to be affected by remarkable variability. In particular, *tax/rex* values presented a coefficient of variation (CV) higher than 100%, as it is shown in Figure 4, panel a and b, where the results obtained with Dizzy and Cain are reported. To have a better insight, we investigated the density probability function of *tax/rex* that resulted to be characterized by a long tail (see panel c). With different values of z similar distributions appeared, as shown in panel d, where it is plotted the density probability function of *tax/rex* for z=5.

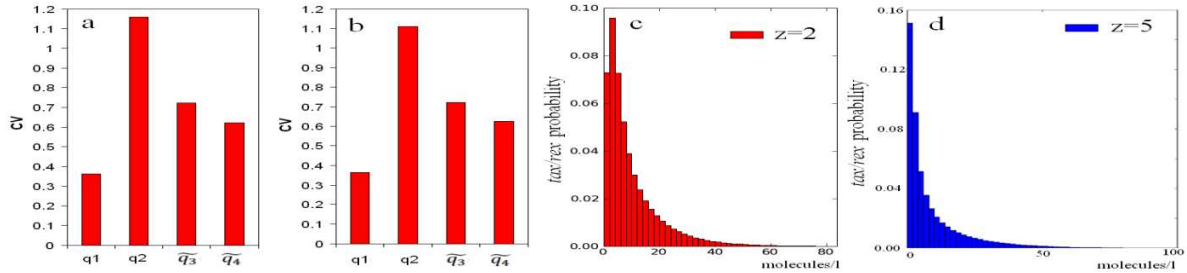


Figure 5. Steady state fluctuations of the steady state solution showed a high variability. a) The CVs obtained from stochastic simulations performed by Dizzy; b) The CVs obtained from stochastic simulations performed by Cain. For these simulations z was set to 2. Density probability functions of the values of *tax/rex* for : c) z=2, and d) z=5.

5. Discussion and conclusions

The gene circuit of the retrovirus HTLV-1 is characterized by positive and negative feedback phenomena, due to the regulatory proteins *Tax* and *Rex*, thus candidating it as a potential relaxation oscillator. To test this hypothesis, a model of the retroviral gene and protein kinetics was developed on the basis of well-established knowledge. The model incorporates the cascade of interactions involving the viral promoter and the biological processes of transcription, translation and degradation, assuming the former to be faster and in equilibrium with respect to the latter. Reasonable simplifications were introduced to limit the number of parameters, and approximate but reasonable numerical values were assigned to them based on information derived in the literature, with the only

exception of order of *Rex* multimerization z, which is unknown. Thus the analysis of the system behavior was for different values of z. To test *in silico* the possibility of observing periodic patterns, bifurcation analysis was performed on the deterministic system of differential equations, considering two model parameters as tunable on the basis of experimental observations: the degradation rate of regulatory proteins and the parameter m, which is related to the multiplicity of infection. Our results show that oscillatory behaviors take place if the kinetics of the positive feedback are faster than those of the negative feedback, as expected from a relaxation oscillator; moreover, higher values of *Tax* degradation rate result in smaller amplitudes and oscillation periods. Obviously, other parameters influence the system behavior: the values of RD compatible with oscillatory behavior depend on m as well as on z. In particular, periodic oscillations are prevented if z is greater than 5.

To reach periodic oscillations, the *Tax* degradation rate should be increased with respect to that of *Rex*. Three recent experimental techniques support the possibility of altering protein degradation rate in practice, two based on enhancing the ubiquitin degradation pathway and the latter on protein tagging. Bachmair et al.[33] pointed out the important role of the amino-terminus in stabilizing/destabilizing the proteins which undergo ubiquitination: in their experiments the β-Galactosidase half-life lowered from more than 20 hours to less than 3 minutes depending on the amino-terminus. Rogers and Rechsteiner[34,35] observed the correlation between the high presence of PEST sequences – where PEST is the nice abbreviation of proline (P), glutamic acid (E), serine (S), and threonine (T) - and the short half-lives of proteins degraded by the ubiquitin pathway. Consequently, the substitution of PEST amino acids with more stable ones should increase the protein half-life. McGinness et al.[36] suggested the import of the E.coli ClpXP protease into eukaryotic cells and the addition of an appropriate ssrA tag to the protein under exam to modulate its degradation rate. Briefly, the tag should have weak affinity for the protease so the introduction of the SspB adaptor protein can be used as a control lever to increase this affinity and, consequently, the proteolysis of the tagged proteins, induced by the ClpXP protease. However, application to the HTLV-1 gene circuit of these three methods is not immediate because *Tax* and *Rex* are translated from the same transcript *tax/rex* and, consequently, the mutation of one protein implies the mutation of the other. To address this problem and make *Tax* kinetics faster than *Rex* kinetics, a possible solution is supplied by the PEST hypothesis. Since the coding sequences of *Tax* and *Rex* are (4829..4832, 6951..8008) and (4773..4832, 6950..8008), respectively (data from the NCBI Reference Sequence NC_001436.1), the sequence (4773..4832) is present in *Rex* but absent in *Tax*. Moreover, it configures as a PEST region since it includes one glutamine, one serine, two threonines and five prolines out of 19 aa, which are all destabilizing amino acids. Therefore, their substitution with more stable amino acids, by site-directed mutagenesis[37,38], should decrease the *Rex* degradation rate, allowing to obtain periodic oscillations. Conversely, practical applicability of the methods proposed by Bachmair et al.[33] and McGinness et al.[36] to our system is still an open issue, due to the overlapping of *Rex* and *Tax* protein sequences.

Since stochasticity and discreteness can cause deviations of the true system behavior from the predictions of deterministic differential equations when the molecular populations are small, as is the case of the viral promoter sites in our model, the Gillespie algorithm was used to perform stochastic simulations. Simulations revealed the leveling of *tax/rex* time course essentially due to the lack of synchronization among the oscillators delivered in distinct cells. In particular, peaks occur at different times in the distinct realizations, as experimentally observed by Stricker et al.[39] by measuring single-cell fluorescence trajectories. The problem of cell synchronization can be addressed by electroporation, but only partially, since the recently developed methodology of transfection[40], which allows the delivery of genes of interest directly into the nuclear compartment in a time period of microseconds, does not guarantee the persistence of synchronization. A continuous synchronizing signal[41,42] may be needed to preserve the forced initial synchronization over time, but currently this is not available. As a consequence, experimental validation is not straightforward and will require single-cell measurements of out of phase oscillators, by time-lapse microscopy and using GFP reporters. Recent findings support the applicability of this technique to viral genes, since some lentiviral vectors with GFP as reporter of the transactivator gene *Tat* were described for the HIV gene circuit[1]. Particularly interesting is the wild type HIV-1 with the gene *Nef* substituted by the GFP; a fascinating testable hypothesis is the realization of a reporter version of the HTLV-1 genome with *tax/rex* substituted by the GFP, to be delivered in addition to the appropriately mutated virus. We expect the RNA *gag* transcribed from this reporter

construct, be either doubly spliced into the GFP, supplying fluorescence intensities proportional to the presence of *tax/rex*, or transferred to the cytoplasm and degraded.

Stochastic fluctuations of the steady state solution were also examined. All state variables are affected by remarkable variability. In particular, *tax/rex* values have a CV higher than 100%, with a long tail of their density probability function, indicating that *tax/rex* gene expression is likely to sometimes assume very high values because of stochastic fluctuations, suggesting mechanisms of retroviral activation similar to those recently proposed by Weinberger for HIV[1].

In conclusion, the bifurcation analysis of the proposed model of the HTLV-1 gene circuit revealed that periodic patterns are possible, provided that *Tax* kinetics is faster than *Rex* kinetics. The next step is the experimental validation of these predictions. However, the stochastic simulations pointed out the problem of cell synchronicity; consequently, single-cell measurements are necessary to observe oscillatory patterns of genes or proteins. Moreover, the high variability at steady state suggests mechanisms of retroviral activation similar to those proposed for HIV.

Appendix

Following Ref. 5,6, the HTLV-1 chemical reactions[13] were divided into two categories: fast and slow; in particular, protein multimerization and complex formation were assumed to be faster than the processes of transcription, translation and degradation of proteins and transcripts. It is reasonable to assume that fast reactions are of the order of seconds, similarly to λ phage's[6] and thus, although not exactly known, much faster than protein degradation in eukaryotes, that is of the order of hours or days[27]. Therefore, faster reactions can be safely assumed to be in equilibrium with respect to the slower ones. In the following paragraphs, we will introduce the fast reactions and then the slower ones.

A.1 Fast reactions

In this paragraph, the following HTLV-1 biological processes[13] will be described: (1) dimerization and complex formation, which lead to transactivation; (2) cooperative interactions, which are necessary for transcription; and (3) *Rex* multimerization.

Dimerization and complex formation: HTLV-1 transactivation is due to the binding of a complex, composed of dimers of *Tax* and dimers of the cellular transcription factor CREB, to the Tax Responsive Element[43] (TRE) in the viral Long Terminal Repeat, where the viral promoter is located. A set of chemical reactions are used to describe:

1. *Tax* dimers formation and their transfer to the nucleus (*Tax* is a shuttling protein[44], i.e. it transfers from the cytoplasm to the nucleus and viceversa, so dimers are present in the whole cell but only the nuclear fraction is involved in transactivation).
2. CREB dimers formation and their transfer to the nucleus.
3. The formation of a $CREB_2$-TAX_2 complex, which subsequently binds to the TRE inducing transactivation, and the alternative interaction $CREB_2$-TRE from which the basal transcription follows. In the following we will indicate the transactivated promoter sites and the nontransactivated ones with the abbreviations $TPrS_0$, and $NTPrS_0$, respectively, and with TRE the inactivated viral promoter sites. Table A1 summarizes the chemical reactions and the corresponding equilibria.

Table A.1. Chemical reactions and the corresponding equilibria concerning molecular dimerization and complex formation.

Tax dimerization	$2\,TAX \underset{}{\overset{K_1}{\rightleftharpoons}} TAX_2$	$[TAX_2] = K_1[TAX]^2$
Tax transfer to the nucleus	$TAX_2 \overset{K_2}{\rightleftharpoons} TAX_2^n$	$[TAX_2^n] = K_2[TAX_2]$
CREB dimerization	$2\,CREB \overset{K_3}{\rightleftharpoons} CREB_2$	$[CREB_2] = K_3[CREB]^2$
CREB transfer to the nucleus	$CREB_2 \overset{K_4}{\rightleftharpoons} CREB_2^n$	$[CREB_2^n] = K_4[CREB_2]$
$CREB_2$-TAX_2 complex formation in the nucleus	$CREB_2^n + TAX_2^n \overset{K_5}{\rightleftharpoons} Complex$	$[Complex] = K_5K_4K_3[CREB]^2\,K_2K_1[TAX]^2$

Binding of the complex to TRE (transactivated promoter)	$\underset{}{\overset{K_{6T}}{\text{Complex} + \text{TRE} \rightleftharpoons \text{TPrS}_0}}$	$[\text{TPrS}_0] = K_{6T}K_5K_4K_3[\text{CREB}]^2\, K_2K_1[\text{TAX}]^2\, [\text{TRE}]$
Binding of $CREB_2$ to TRE (non-transactivated promoter)	$\underset{}{\overset{K_{6NT}}{\text{CREB}_2^n + \text{TRE} \rightleftharpoons \text{NTPrS}_0}}$	$[\text{NTPrS}_0] = K_{6NT}K_4K_3[\text{CREB}]^2\, [\text{TRE}]$

Cooperative interactions: viral transcription involves a cascade of co-activators and general transcription factors, CBP/p300, PCAF, TFIIA, TFIIB,TFIID[13], whose binding reactions with the promoter regions are described in Table A.2. The transactivated and the non-transactivated promoter sites will be indicated with TPrS_i, and NTPrS_i, where the suffix i denotes the step in the cascade. Following the same line of reasoning which underlies the formulation of the well-known pseudo-first order rate equations[45], the concentrations of all co-activators and general transcription factors are assumed to be constant or in great excess with respect to the viral promoters they interact with, so that the effects of the variations of their concentrations on the viral kinetics are negligible. Consistently, a low multiplicity of infection is assumed, i.e. we suppose that few viral genomes are integrated in the host cells. Table A.2 summarizes the chemical reactions and the corresponding equilibria.

Table A.2. Chemical reactions and the corresponding equilibria concerning cooperative interactions.

Binding of CBP/p300 to the non-transactivated promoter site	$\underset{}{\overset{K_{7A}}{\text{NTPrS}_0 + \text{CBP/p300} \rightleftharpoons \text{NTPrS}_1}}$	$[\text{NTPrS}_1] = K_{7A}[\text{NTPrS}_0]\,[\text{CBP}] = K'_{7A}[\text{NTPrS}_0]$
Binding of PCAF to the non-transactivated promoter site	$\underset{}{\overset{K_{8A}}{\text{NTPrS}_1 + \text{PCAF} \rightleftharpoons \text{NTPrS}_2}}$	$[\text{NTPrS}_2] = K_{8A}[\text{NTPrS}_1]\,[\text{PCAF}] = K'_{8A}[\text{NTPrS}_1]$
Binding of TFIIA to the non-transactivated promoter site	$\underset{}{\overset{K_{9A}}{\text{NTPrS}_2 + \text{TFIIA} \rightleftharpoons \text{NTPrS}_3}}$	$[\text{NTPrS}_3] = K_{9A}[\text{NTPrS}_2]\,[\text{TFIIA}] = K'_{9A}[\text{NTPrS}_2]$
Binding of TFIIB to the non-transactivated promoter site	$\underset{}{\overset{K_{10A}}{\text{NTPrS}_3 + \text{TFIIB} \rightleftharpoons \text{NTPrS}_4}}$	$[\text{NTPrS}_4] = K_{10A}[\text{NTPrS}_3]\,[\text{TFIIB}] = K'_{10A}\,[\text{NTPrS}_3]$
Binding of TFIID to the non-transactivated promoter site	$\underset{}{\overset{K_{11A}}{\text{NTPrS}_4 + \text{TFIID} \rightleftharpoons \text{NTPrS}_5}}$	$[\text{NTPrS}_5] = K_{11A}[\text{NTPrS}_4]\,[\text{TFIID}] = K'_{11A}\,[\text{NTPrS}_4]$
Binding of CBP/p300 to the transactivated promoter site	$\underset{}{\overset{K_{7B}}{\text{TPrS}_0 + \text{CBP/p300} \rightleftharpoons \text{TPrS}_1}}$	$[\text{TPrS}_1] = K_{7B}[\text{TPrS}_0]\,[\text{CBP}] = K'_{7B}[\text{TPrS}_0]$
Binding of PCAF to the transactivated promoter site	$\underset{}{\overset{K_{8B}}{\text{TPrS}_1 + \text{PCAF} \rightleftharpoons \text{TPrS}_2}}$	$[\text{TPrS}_2] = K_{8B}[\text{TPrS}_1]\,[\text{PCAF}] = K'_{8B}[\text{TPrS}_1]$
Binding of TFIIA to the transactivated promoter site	$\underset{}{\overset{K_{9B}}{\text{TPrS}_2 + \text{TFIIA} \rightleftharpoons \text{TPrS}_3}}$	$[\text{TPrS}_3] = K_{9B}[\text{TPrS}_2]\,[\text{TFIIA}] = K'_{9B}[\text{TPrS}_2]$
Binding of TFIIB to the transactivated promoter site	$\underset{}{\overset{K_{10B}}{\text{TPrS}_3 + \text{TFIIB} \rightleftharpoons \text{TPrS}_4}}$	$[\text{TPrS}_4] = K_{10B}[\text{TPrS}_3]\,[\text{TFIIB}] = K'_{10B}[\text{TPrS}_3]$
Binding of TFIID to the transactivated promoter site	$\underset{}{\overset{K_{11B}}{\text{TPrS}_4 + \text{TFIID} \rightleftharpoons \text{TPrS}_5}}$	$[\text{TPrS}_5] = K_{11B}[\text{TPrS}_4]\,[\text{TFIID}] = K'_{11B}[\text{TPrS}_4]$

Rex multimerization: protein *Rex* multimerizes[46], but the exact kind of multimer it forms is not known; in particular, there is evidence that Rex at least dimerizes[47], but the formation of complexes of higher orders like pentamers or hexamers is likely as well. To describe *Rex* multimerization, the Helfferich procedure for multistep reactions[48] is applied, summarizing with K_{12} the ratio of the overall forward and backward kinetics constants. Like *Tax*, also *Rex* is a shuttling protein[49] and only its nuclear fraction is involved in the nuclear export of incompletely spliced transcripts. Table A.3 summarizes the chemical reactions and the corresponding equilibria; with the symbol z we indicate the number of *Rex* molecules involved in the multimer formation, e.g. z=2 for dimers and z=5 for pentamers.

Table A.3. Chemical reactions and the corresponding equilibria concerning *Rex* multimerization and its transfer to the nucleus.

Rex multimerization	$\underset{}{\overset{K_{12}}{z\ \text{REX} \rightleftharpoons \ldots \rightleftharpoons \text{REX}_z}}$	$[\text{REX}_z] = K_{12}[\text{REX}]^z$
Rex transfer to the nucleus	$\underset{}{\overset{K_{13}}{\text{REX}_z \rightleftharpoons \text{REX}_z^n}}$	$[\text{REX}_z^n] = K_{13}[\text{REX}_z] = K_{13}K_{12}[\text{REX}]^z$

A.2 Slow reactions

Transcription, translation and the degradation of transcripts and proteins are irreversible and slow reactions; in the following we will describe: (1) the transcription of the primary transcript *gag*; (2) the alternative splicing of *gag* and the nuclear export of mRNAs; (3) the kinetics of the transcript *tax/rex*; (4) the kinetics of the proteins *Tax* and *Rex*; and (5) the kinetics of the incompletely spliced transcripts.

The transcription of the primary transcript *gag*: *gag* synthesis $S_1(t)$ is due to the basal transcription $S_{11}(t)$ and to the transcription induced by transactivation $S_{12}(t)$. Chemical reactions of the transcription processes $S_{11}(t)$ and $S_{12}(t)$, with their mathematical formulations, are shown in Table A.4, where the concentration of RNA polymerase (RNAp), is assumed to be constant or in great excess with respect to the viral promoter sites it interacts with. In Table A.4, β_1 indicates the gain in transcription, i.e. the number of transcripts generated by the binding of a molecule of RNAp to DNA and the subsequent process of *gag* transcription, k_t is a reaction rate constant [1/h] and c_0 a multiplicative constant.

Table A.4. Chemical reactions of transcriptions and the corresponding syntheses of nuclear *gag*.

Basal transcription	$\mathrm{NTPrS_5 + RNAp \xrightarrow{k_t} NTPrS_5 + RNAp} + \beta_1$ nuclear *Gag*	$S_{11}(t) = \beta_1 k_t p_0 [\mathrm{NTPrS_5}](t) = \\ = \beta_1 k_t p_0 \mathrm{K'_{11A} K'_{10A} K'_{9A} K'_{8A} K'_{7A}} [\mathrm{NTPrS_0}](t)$
With transactivation	$\mathrm{TPrS_5 + RNAp \xrightarrow{c_0 k_t} TPrS_5 + RNAp} + \beta_1$ nuclear *Gag*	$S_{12}(t) = \beta_1 c_0 k_t p_0 [\mathrm{TPrS_5}](t) = \\ = \beta_1 c_0 k_t p_0 \mathrm{K'_{11B} K'_{10B} K'_{9B} K'_{8B} K'_{7B}} [\mathrm{TPrS_0}](t)$

From the transcription processes $S_{11}(t)$ and $S_{12}(t)$, in molecules/h, of Table A.4, we derive the transcription rates $S=\beta_1 k_t p_0 \mathrm{K'_{11A} K'_{10A} K'_{9A} K'_{8A} K'_{7A}}$ and $S'=\beta_1 c_0 k_t p_0 \mathrm{K'_{11B} K'_{10B} K'_{9B} K'_{8B} K'_{7B}}$, in 1/h, for the basal transcription and transcription following transactivation, and we call c_1 their ratio, i.e. $c_1 = S'/S$. To have a better insight on the effects of the *Tax*-induced transactivation on the total *gag* synthesis $S_1(t)$, we introduce the multiplicity of infection (MOI), which is the mean number of viral genomes integrated in the host genome per cell. Then, we indicate with [cells] the cell concentration in a sample, in number of cells/l. Consequently, the concentration of retroviral genomes integrated in the host cells, m, equals MOI*[cells], in number of viral molecules/l. Now, making the working hypothesis that all the promoters are demethilated, i.e. none of them is *a priori* prevented from being involved in transcription, we derive that:

$$\mathrm{m} = [\mathrm{TRE}](t) + [\mathrm{NTPrS_0}](t) + [\mathrm{TPrS_0}](t) \tag{A.1}$$

The total transcription $S_1(t)$ is the sum of $S_{11}(t)$ and $S_{12}(t)$. Therefore, by summing up we have:

$$S_1(t) = \mathrm{S}\,[\mathrm{NTPrS_0}](t) + S'\,[\mathrm{TPrS_0}](t) = S\{[\mathrm{NTPrS_0}](t) + c_1[\mathrm{TPrS_0}](t)\} \tag{A.2}$$

where the constant c_1 was introduced to obtain the right-hand side of the equation. Then, from Eq. A.1 and A.2, and by some algebraic passages we obtain:

$$S_1(t) = S\{\mathrm{m} - [\mathrm{TRE}](t) + (c_1 - 1)[\mathrm{TPrS_0}](t)\} = S\{\mathrm{m} - [\mathrm{TRE}](t)\}\left\{1 + \frac{[\mathrm{TPrS_0}](t)}{\mathrm{m} - [\mathrm{TRE}](t)}(c_1 - 1)\right\} \tag{A.3}$$

Now, we focus on the term $\frac{[\mathrm{TPrS_0}](t)}{\mathrm{m}-[\mathrm{TRE}](t)}$ of Eq. A.3 and call $q_3(t)$ the *Tax* concentration. From Eq A.1 and equilibria of Table A.1 we obtain:

$$\frac{[\mathrm{TPrS_0}](t)}{\mathrm{m} - [\mathrm{TRE}](t)} = \frac{[\mathrm{TPrS_0}](t)}{[\mathrm{NTPrS_0}](t) + [\mathrm{TPrS_0}](t)} = \frac{\mathrm{q_3}(t)^2}{\mathrm{h_1^2} + \mathrm{q_3}(t)^2} \qquad \text{with: } \mathrm{h_1^2} = \frac{K_{6NT}}{K_{6T}} \frac{1}{K_5 \mathrm{K_2 K_1}} \tag{A.4}$$

Eq. A.4 shows that an elevated *Tax* concentration increases the number of transactivated promoter sites among the promoters involved in transcription, which are $[\mathrm{TPrS_0}](t)+[\mathrm{NTPrS_0}](t)$. Then, by inserting the right term of Eq. A.4 in Eq. A.3 the synthesis of nuclear *gag* becomes:

$$S_1(t) = S\{\mathrm{m} - [\mathrm{TRE}](t)\} + \{\mathrm{m} - [\mathrm{TRE}](t)\}\, S\, \beta'_1 \frac{\mathrm{q_3}(t)^2}{\mathrm{h_1^2} + \mathrm{q_3}(t)^2} \tag{A.5}$$

where $\beta'_1 = c_1 - 1$ is the transactivation constant (adimensional). Eq. A.5 shows a saturative effect of *Tax* concentration on *gag* synthesis given by the term $\frac{\mathrm{q_3}(t)^2}{\mathrm{h_1^2} + \mathrm{q_3}(t)^2}$, which can be due to the limited number of integrated viral promoter sites; in other words once they have been all transactivated, the *gag* transcription $S_1(t)$ saturates and no further increase is possible. Now, let $p_1 = [\mathrm{CREB}]$, which is assumed to be constant or in great excess with respect to the viral promoter sites it interacts with, and focus on the term m-[TRE](t). From equilibria of Table A.1 and Eq. A.1, and by several algebraic passages we obtain:

$$\mathrm{m} - [\mathrm{TRE}](t) = \mathrm{m} - \frac{m}{\mathrm{f}(\mathrm{p_1}, \mathrm{q_3}(t))} \tag{A.6}$$

where:

$$f(p_1, q_3(t)) = 1 + \frac{{p_1}^2}{h_2^2} + \frac{{p_1}^2}{h_2^2}\frac{1}{h_1^2} q_3(t)^2 \qquad \text{with } h_2^2 = \frac{1}{K_{6NT} K_4 K_3} \tag{A.7}$$

Eq. A.6 implies that if there is a lot of *Tax* or CREB then $f(p_1, q_3(t)) \gg m$ and, consequently, $m - [TRE](t) \approx m$, i.e. the number of promoter sites which are not involved in transcription becomes negligible. Therefore, by inserting the right term of Eq. A.6 in Eq. A.5, the synthesis of *gag* becomes:

$$S_1(t) = S\left\{m - \frac{m}{f(p_1, q_3(t))}\right\} + \left\{m - \frac{m}{f(p_1, q_3(t))}\right\} S\,\beta_1' \frac{q_3(t)^2}{h_1^2 + q_3(t)^2} \tag{A.8}$$

The alternative splicing of *gag* and the transfer of mRNAs to the cytoplasm: in presence of nuclear *Rex*, the transcripts are transferred incompletely spliced to the cytoplasm whereas, in absence of *Rex*, *gag* is doubly spliced into *tax/rex*[13]. Quantitatively, the amount of incompletely spliced RNAs transferred to the cytoplasm depends on the fraction of *gag* molecules which interact with multimers of *Rex* in the nucleus. This fraction is represented by the function $g(q_1(t), q_4(t), z)$, see Eq. A.9-11.

$$g(q_1(t), q_4(t), z) = \begin{cases} \frac{K_{13} K_{12} q_4^z}{q_1} = \frac{q_4^z}{h_3^z}\frac{1}{q_1} & \text{with } h_3^z = \frac{1}{K_{13} K_{12}} \qquad \text{If } \frac{q_4^z}{h_3^z} < q_1 \text{ and } q_1 > 0 \quad \text{(A.9)} \\ 1 \qquad \text{If } \frac{q_4^z}{h_3^z} > q_1 \text{and } q_1 > 0 \quad \text{(A.10)} \\ 0 \qquad \text{If } q_1 = 0 \quad \text{(A.11)} \end{cases}$$

where the variable $q_4(t)$ represents the *Rex* concentration in the sample, and the algebraic passages derive from the equilibria of Table A.3. The transcripts that are not transferred to the cytoplasm are degraded by the nuclear enzymes. Therefore, the total decay of g*ag* in the nucleus is due to the nuclear export plus the nuclear degradation (see reactions in Table A.5, where k_s and k_{01} are reaction rate constants, and c_2 is a multiplicative constant; for simplicity, the splicing and the nuclear export are condensed into a unique reaction).

Table A.5. Nuclear *gag* decay is due to the nuclear export of incompletely spliced RNAs (with Rex case), the splicing and nuclear export of doubly spliced RNAs (without *Rex* case) and to the nuclear degradation of the transcripts.

with *Rex*	k_s Nuclear *gag* + Rex_z → → Cytoplasmatic incompletely spliced RNA + Rex_z	$L_{11}(\tau) = g(q_1(t), q_4(t), z)\, k_s q_1(t)$
without *Rex*	$c_2 k_s$ Nuclear *gag*+ $\sum_i splicing\ factor_i$ → → Cytoplasmatic *tax/rex* +$\sum_i splicing\ factor_i$	$L_{12}(\tau) = \{1 - g(q_1(t), q_4(t), z)\}\, c_2\, k_s q_1(t)$
Nuclear degradation	k_{01} Nuclear *gag*→	$D_1(\tau) = \{1 - g(q_1(t), q_4(t), z)\}\, k_{01} q_1(t)$

The sum of the three addends of Table A.5 is:

$$\text{Nuclear } gag \text{ decay}(t) = L_{11}(t) + L_{12}(t) + D_1(t) = \tag{A.12}$$
$$= g(q_1(t), q_4(t), z)\, k_s q_1(t) + \{1 - g(q_1(t), q_4(t), z)\}\, c_2\, k_s\, q_1(t) + \{1 - g(q_1(t), q_4(t), z)\}\, k_{01} q_1(t)$$

Therefore, by considering *gag* synthesis (Eq. A.8) and nuclear decay (Eq. A.12), the rate equation for nuclear g*ag* is:

$$\frac{dq_1}{dt} = S_1(t) - Nuclear\ gag\ decay(t) = \tag{A.13}$$
$$= S\left\{m - \frac{m}{f(p_1, q_3(t))}\right\} + \left\{m - \frac{m}{f(p_1, q_3(t))}\right\} S\,\beta_1' \frac{q_3(t)^2}{h_1^2 + q_3(t)^2} - g(q_1(t), q_4(t), z) k_s q_1(t) - \{1 - g(q_1(t), q_4(t))\}\{c_2\, k_s + k_{01}\} q_1(t)$$

where $q_1(t)$ represents the concentration of nuclear *gag*, in molecules/l.

The kinetics of the transcript *tax/rex*: *tax/rex* kinetics is given in Eq. A.14, where $S_2(t)$ is the synthesis following the double splicing of nuclear *gag*, i.e. $S_2(t) = L_{12}(t)$ (Table A.5), and $D_2(t)$ represents the degradation; the reactions are reported in Table A.6, where k_{02} is the reaction rate constant of the *tax/rex* degradation process. Therefore, the rate equation for *tax/rex* is:

$$\frac{dq_2}{dt} = S_2(t) - D_2(t) = \{1 - g(q_1(t), q_4(t), z)\} c_2\, k_s\, q_1(t) - k_{02}\, q_2(t) \tag{A.14}$$

where $q_2(t)$ represents the *tax/rex* concentration, in molecules/l.

Table A.6. Transcript *tax/rex* kinetics is due to two terms: synthesis, by double splicing of nuclear *gag*, and degradation.

tax/rex synthesis	$\xrightarrow{c_2 k_s}$ Nuclear *Gag*+ $\sum_i splicing\ factor_i \rightarrow$ $\rightarrow$ Cytoplasmatic *tax/rex* +$\sum_i splicing\ factor_i$	$S_2(t) = L_{12}(\tau) = \{1 - g(q_1(t), q_4(t), z)\}\ c_2\ k_s\ q_1(t)$
tax/rex degradation	$\xrightarrow{k_{02}}$ *tax/rex* $\rightarrow$	$D_2(t) = k_{02}\ q_2(t)$

The kinetics of the proteins *Tax* and *Rex*: *Tax* and *Rex* are translated from the same transcript *tax/rex*; their synthesis reactions are shown in Table A.7, where k_{T3} and k_{T4} are reaction rate constants [1/h], β_3 and β_4 the gains in protein translations [protein molecules/transcript molecules], and β'_3, β'_4 their products [molecules/(molecules*h)].

Table A.7. Reactions of translation of the regulatory proteins *Tax* and *Rex*.

Tax synthesis	$\xrightarrow{k_{T3}}$ *tax/rex* $\rightarrow$ *tax/rex* + β_3 *Tax*	$S_3(t) = k_{T3}\ \beta_3\ q_2(t) = \beta'_3 q_1(t)$
Rex synthesis	$\xrightarrow{k_{T4}}$ *tax/rex* $\rightarrow$ *tax/rex* + β_4 *Rex*	$S_4(t) = k_{T4}\ \beta_4\ q_2(t) = \beta'_4 q_1(t)$

Therefore, the rate equations for *Tax* and *Rex* are:

$$\frac{dq_3}{dt} = S_3(t) - D_3(t) = \beta'_3 q_2(t) - k_{03} q_3(t) \quad (A.15)$$

$$\frac{dq_4}{dt} = S_4(t) - D_4(t) = \beta'_4 q_2(t) - k_{04} q_4(t) \quad (A.16)$$

where $q_3(t)$ and $q_4(t)$ represent the concentrations [molecules/l] of *Tax* and *Rex*, respectively, and k_{03} and k_{04} are the rate constants of the corresponding degradation reactions.

The kinetics of the incompletely spliced transcripts: analogously to *tax/rex*, the rate equations of the incompletely spliced RNAs are:

$$\frac{dq_{5i}}{dt} = S_{5i}(t) - D_{5i}(t) = f_{5i} g(q_1(t), q_4(t), z)\ k_s\ q_1(t) - k_{05i}\ q_{5i}(t) \qquad \text{with i=1,..,n} \quad (A.17)$$

where $q_{5i}(t)$ represent the concentrations of incompletely spliced RNAs, k_{05i} are the rate constants of the corresponding degradation reactions, and f_i are the fractions of nuclear *gag* singly spliced into each transcript (or remained unspliced in the case of the cytoplasmatic *gag*) and transferred to the cytoplasm.

From the rate equations (Eq. 13-17), a system of differential equations was derived. To limit the complexity of the model, a number of assumptions were made:

Simplification 1: Grone et al.[14] observed that the total viral RNA in the sample is *Rex*-independent. From Eq. A.13, A.14 and A.17, total RNA is:

$$\int_0^t \left(\frac{dq_1}{d\tau} + \frac{dq_2}{d\tau} + \Sigma_i \frac{dq_{5i}}{d\tau}\right) d\tau = \int_0^t [\, S_1(\tau) - D_1(\tau) - D_2(\tau) - \Sigma_i D_{5i}(\tau)] d\tau \quad (A.18)$$

If the nuclear degradation $D_1(t)$ is negligible with respect to the other addends, the total viral RNA does not depend on *Rex*, thus leading to k_{01} in Eq. A.13 much smaller than the transcript degradation rates. For simplicity, k_{01} was set to 0 in Eq. A.13.

Simplification 2: we assume that the multiplicity of infection is so low that $\frac{m}{f(p_1, q_3(t))} \approx 0$ in Eq. A.13. Since the MOI can be regulated at the time of virus delivery by the titer of viral particles, this condition is attainable.

Simplification 3: we assume that the nuclear-cytoplasmatic transport take place with the same rate for every transcript, i.e. $k_s \approx c_2 k_s$ in Eq. A.13.

Simplification 4: since the kinetics of incompletely spliced RNAs do not affect the possible oscillatory behavior of *tax/rex* and *gag*, their kinetics were excluded from further considerations.

From Simplifications 1-4, we obtain the system of differential equations of Eq. A.19-22:

$$\begin{cases} \frac{dq_1}{dt} = m\,S + m\,S\,\beta'_1 \frac{q_3(t)^2}{h_1^2 + q_3(t)^2} - k_s q_1(t) & (A.19) \\ \frac{dq_2}{dt} = \{1 - g(q_1(t), q_4(t), z)\}\, k_s q_1(t) - k_{02}\, q_2(t) & (A.20) \\ \frac{dq_3}{dt} = \beta'_3 q_2(t) - k_{03} q_3(t) & (A.21) \\ \frac{dq_3}{dt} = \beta'_4 q_2(t) - k_{04} q_4(t) & (A.22) \end{cases}$$

References

1. L. S. Weinberger, R. D. Dar, M. L. Simpson, *Nat Genet* **40**, 466 (2008).
2. T. S. Gardner, C. R. Cantor, J. J. Collins, *Nature* **403**, 339 (2000).
3. M. B. Elowitz, S. Leibler, *Nature* **403**, 335 (2000).
4. M. Kaern, W. J. Blake, J. J. Collins, *Annu Rev Biomed Eng* **5**, 179 (2003).
5. J. Hasty, J. Pradines, M. Dolnik, J. J. Collins, *Proc Natl Acad Sci U S A* **97**, 2075 (2000).
6. J. Hasty, F. Isaacs, M. Dolnik, D. McMillen, J. J. J. Collins, *Chaos* **11**, 207 (2001).
7. M. Ramachandra, et al., *Nat Biotechnol* **19**, 1035 (2001).
8. J.W. Bainbridge, et al., *N Engl J Med*. **358**,2282 (2008)
9. A. Corradin, et al., *The 10th International Conference on Systems biology,* Aug 30-Sept 4, 2009. Stanford, California (accepted).
10. M. Thattai, A. van Oudenaarden, *Proc Natl Acad Sci U S A* **98**, 8614 (2001).
11. M. B. Elowitz, A. J. Levine, E. D. Siggia, P. S. Swain, *Science* **297**, 1183 (2002).
12. D. T. Gillespie, *Annu Rev Phys Chem* **58**, 35 (2007).
13. M.D. Lairmore, G. Franchini, *In Fields Virology, Fifth Edition. Ed. David M. Knipe and Peter M. Howley. Lippincott Williams and Wilkins, Philadelphia*, pp. 2071-2106 (2007).
14. M. Gröne, et al., *Virology* **218,**316 (1996).
15. R. Rosin-Arbesfeld, et al., *EMBO J*. **22,**1101 (2003).
16. J. Lewis, *Current biology : CB* **13**, 1398 (2003).
17. L. S. S. Weinberger, T. Shenk, *PLoS Biol* **5** (2006).
18. F. Kashanchi, J.N. Brady, *Oncogene* **24**,5938(2005).
19. J. M. Peloponese Jr, et al. *J Virol*. **78**,11686 (2004).
20. S. J. Jeong, et al., *Biochem Biophys Res Commun*. **381**,294(2009)
21. N. Yildirim, M. C. Mackey, *Biophys J* **84**, 2841 (2003).
22. K. D. Jacobsen, B. M.Willumsen. *J. Mol. Biol.* **252**, 289 (1995)
23. J. F. Kugel, J. A. Goodrich, *Proc Natl Acad Sci U S A* **95**,9232 (1998).
24. A. Corradin, et al., *Proceedings of the ENFIN Symposium at the Functional Genomics & Disease Conference*, Oct 2-4, 2008. Innsbruck, Austria
25. F. Rende, et al., *Proceedings of the HERN meeting*, June 1-2, 2008. Bruges, Belgium.
26. A. Dhooge, et al., *ACM TOMS.* **29**,141(2003).
27. T. S. Olson, J. F. Dice, *Curr Opin Cell Biol.* **1,**1194 (1989).
28. Y.A. Kuznetsov, *Elements of Applied Bifurcation Theory, Springer-Verlag, New York* (2004).
29. D. T. Gillespie, *Journal of Computational Physics* **22**, 403 (1976).
30. D. T. Gillespie, *The Journal of Physical Chemistry* **81**, 2340 (1977).
31. S. Ramsey, D. Orrell, H. Bolouri, *J Bioinform Comput Biol* **3**, 415 (2005).
32. S. Mauch, M. Stalzer, *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **99**, (5555).
33. A. Bachmair, D. Finley, A. Varshavsky, *Science (New York, N.Y.)* **234**, 179 (1986).
34. S. Rogers, et al., *Science* **234**,364 (1986).
35. M. Rechsteiner, S. W. Rogers, *Trends in biochemical sciences* **21**, 267 (1996).
36. K. E. McGinness, T. A. Baker, R. T. Sauer, *Mol Cell* **22**, 701 (2006).
37. T. A. Kunkel, *Proc Natl Acad Sci U S A* **82**,488 (1985).
38. J. W. Taylor, et al., *Nucleic Acids Res.* **13**,8765 (1985).
39. J. Stricker, et al., *Nature* (2008).
40. F. Rende, et al., *Proceedings of the 14th ICHR: HTLV and Related Viruses.* July 1-4, 2009. Salvador, Brazil.
41. J. Hasty, et al., *Phys Rev Lett*. **88**,148101 (2002).
42. T. Zhou, et al., *Chaos* **18**,037126 (2008).
43. F. Tie, et al., *J Virol*. **70,**8368 (1996).
44. M. Burton, et al., *J Virol.* **74**,2351 (2000).
45. K. A. Connors, *Chemical Kinetics, the study of reaction rates in solution*.VCH Publishers (1991)
46. L. Fu, et al., *FEBS Lett.* **396,**47 (1996).
47. R. E. Smith, et al., *Virolog.* **237**,397 (1997).
48. F.G. Helfferich, *J. Phis. Chem.* **93**,6676 (1989)
49. D. Palmeri, M.H. Malim, *J Virol.* **70**,6442.

EMULSION BASED SELECTION OF T7 PROMOTERS OF VARYING ACTIVITY

ERIC A. DAVIDSON
Institute for Cell and Molecular Biology, University of Texas at Austin, 1 University Station A4800
Austin, Texas 78712, United States

THOMAS VAN BLARCOM
Department of Chemical Engineering, University of Texas at Austin, 1 University Station C0400
Austin, Texas 78712, United States

MATTHEW LEVY
Department of Biochemistry, Albert Einstein College of Medicine, 1301 Morris Park Avenue
Bronx, NY 10461, United States

ANDREW D. ELLINGTON
Department of Chemistry and Biochemistry, University of Texas at Austin, 1 University Station A4800
Austin, Texas 78712, United States

The ability to build and control complex biological systems is greatly enhanced by the generation of related parts with varying strengths. In this way, various parts can be strung together and the connectivity and expression levels can be matched for the desired system performance. Engineered gene circuits, both *in vivo* and *in vitro*, often utilize the T7 RNA polymerase in tandem with the T7 promoter for transcription. In this work, we describe the selection of T7 promoter variants of varying strength by emulsifying *in vitro* transcription with subsequent fluorescence activated cell sorting (FACS) to enrich for active promoters. Such variant promoters should be of use to synthetic biologists for both *in vivo* and *in vitro* applications.

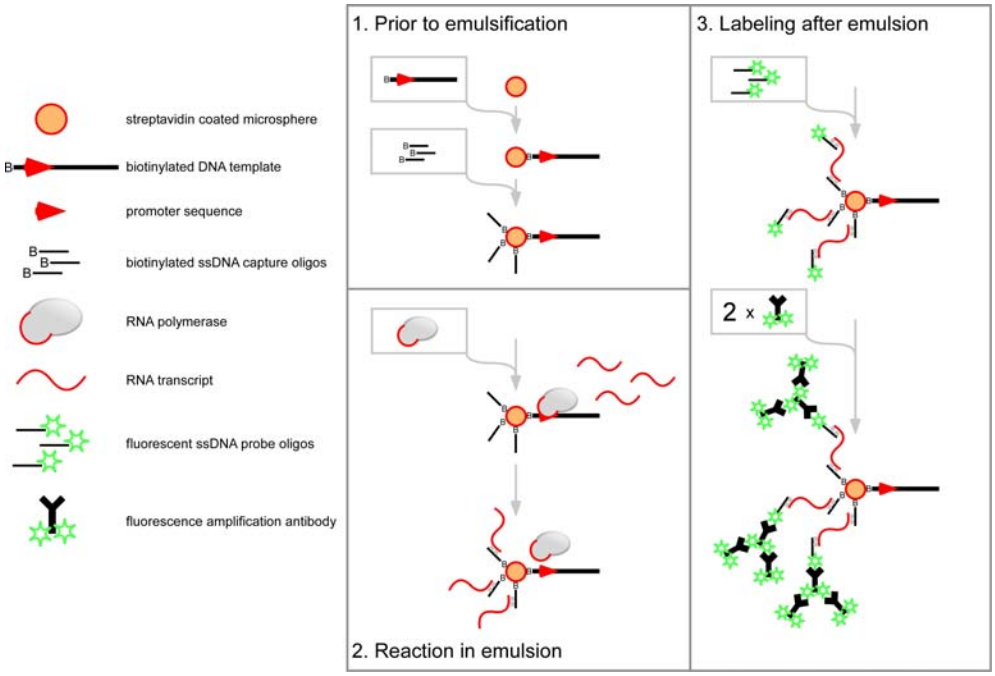
1. Introduction

T7 RNA polymerase (T7 RNAP) plays an important role in the production of RNA transcripts for biotechnology applications. For example, it is widely used for protein overproduction. It has also been routinely used for generating large quantities of functional RNAs *in vitro*, such as aptamers, ribozymes, and siRNAs. T7 RNAP has also played a key role in many engineered and synthetic genetic circuits[1-3], at least in part because a single, monomeric protein can act orthogonally on a short, 17 nt promoter[4]. Indeed, as scientists and engineers contemplate creation of an artificial, minimal cell, the T7 RNA polymerase is at the forefront of the discussion of how to power transcription[5].

Because of the difficulties inherent in modeling synthetic parts and circuits in organisms, the development of synthetic circuits in cell free settings, including artificial cell-like liposomes and water-in-oil emulsions, is an attractive alternative[6-8]. Indeed, cell-free systems are already being used for biomolecular engineering in order to bypass the systemic and evolutionary constraints inherent in living cells[6,9-12]. The shift from *in vivo* to *in vitro* circuitry may be particularly desirable for T7 RNA polymerase and its promoter, since in cells it is active to the point of toxicity when not exquisitely controlled (for example, see reference 13). Typically, synthetic biologists have chosen to control T7 driven gene expression by controlling the production of the T7 RNA polymerase[1].

However, just as the performance of parts *in vitro* is not necessarily predictive of their role *in vivo*, it will be necessary to determine how well individual parts function in emulsions and other cell-like environments. To generate parts for synthetic circuits and to better understand how such circuits can themselves adapt and evolve, we randomized the initially transcribed sequence (ITS) of the T7 promoter, which modulates transcription levels from the core phi10 T7 promoter, and selected for different T7 RNAP promoters with different transcription strengths in emulsions. These and additional selected promoters can now be used as a parts set to contribute to the building of future circuits requiring a range of transcription activities.

A.



B.

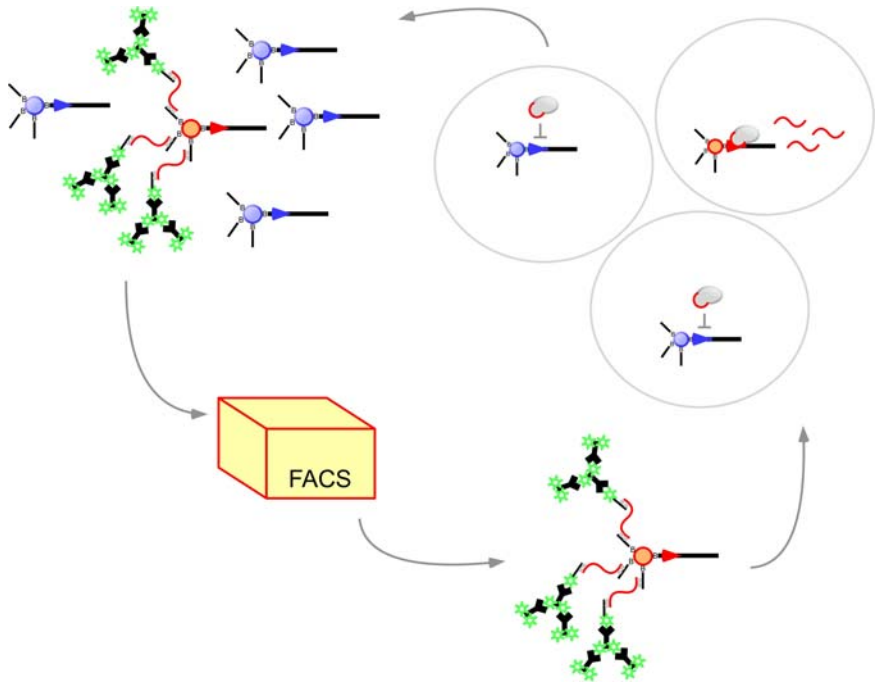


Figure 1. Selection scheme for transcription templates containing active promoter sequences. A. Biotinylated template DNA and biotinylated capture oligonucleotides are immobilized on a streptavidin coated bead. Each bead should one or zero templates and an excess of capture oligonucleotide. Beads are suspended in a transcription reaction and emulsified to encapsulate a single bead per emulsion compartment. RNA produced from the template DNA hybridizes to the bead through the capture oligonucleotide. After breaking the emulsion, the RNA is labeled by a fluorescent probe oligonucleotide and then an antibody based fluorescence amplification system. B. Individual beads are sorted on the basis of their fluorescent signal on an Aria FACS machine. Highly fluorescent beads are collected and reamplified for additional FACS or characterization.

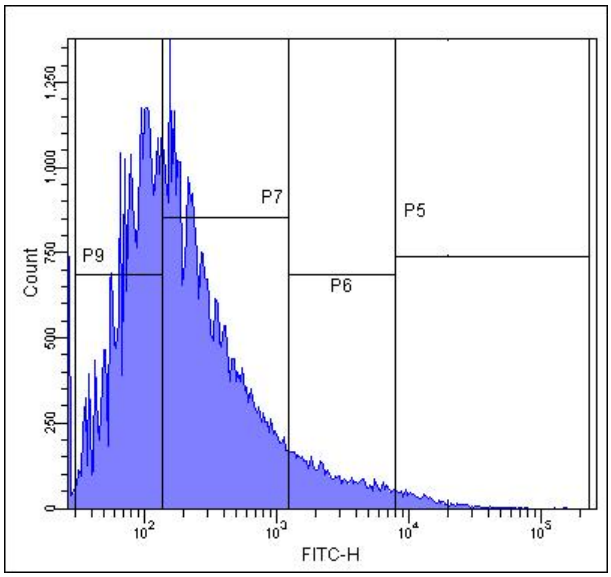
2. Results

2.1. *System design and testing*

We designed a scheme to select for active T7 promoters through emulsified transcription, fluorescence activated cell sorting (FACS) for active templates and then reamplification of collected templates. This process was mediated by attaching biotinylated template DNA to streptavidin coated beads at a ratio of no more than a single template per bead (in practice, no more than a single template for every two beads was added when sorting to minimize beads containing multiple templates). Transcripts produced in emulsion hybridize to the bead containing the template they were transcribed from through a capture oligonucleotide. The RNA remained hybridized to the bead through the process of breaking the emulsion and washing the beads (data not shown). The transcripts are finally fluorescently labeled and beads containing high amounts of fluorescence (and thus active templates) are collected by FACS. The process of preparing beads for emulsified transcription and subsequent fluorescence labeling is presented in Figure 1 and described in detail in the Methods section.

In order to test the viability of this scheme, we investigated its ability to enrich a highly active T7 promoter-containing template from a background of weakly active T7 promoter-containing templates (the weakly active sequence contains +1 C and +2 C instead of +1 G and +2 G in the highly active sequence). The highly active templates were added to a solution of weakly active templates at 1 part in 10 and added to beads at 1 template per 2 beads (thus, at least 50% of the beads should contain no template). During FACS sorting, four non-overlapping regions of the bead population representing nearly the entire fluorescence spectrum were sampled. In order of increasing fluorescence, region P9 contained ~40% of the population, P7 contained ~50% of the population, P6 contained ~7.6% of the population and P5 contained the most fluorescent ~1.6% of the population. The collected

A.

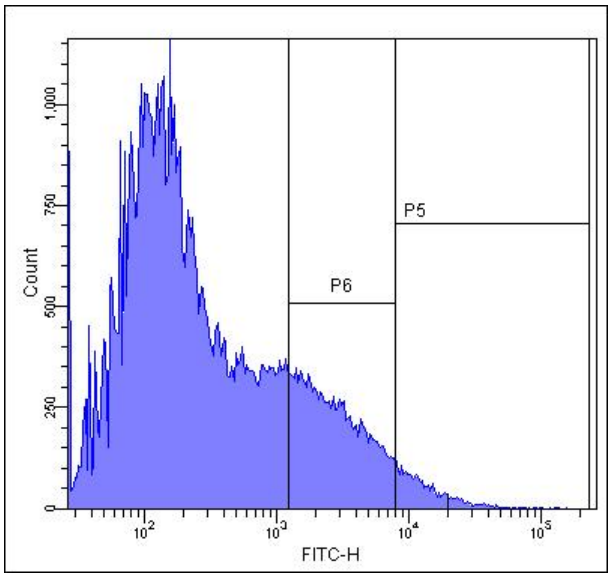


B.

	% of population	active template	weak template
P9	39.5	0	5
P7	50.1	0	5
P6	7.6	4	1
P5	1.6	5	0

Figure 2. One round test selection with a two member population. A. Histogram showing bead distribution by relative fluorescence intensity. The beads carry a two member mixed population of either highly active or weakly active transcription templates. The highly active templates were present at 10% of the total mixed population. Templates were added to the beads at a ratio of 1 template per 2 beads. The total bead population was gated into 4 non-overlapping subpopulations named P9, P7, P6 and P5. B. Population analysis and sequence results show that an increase in relative fluorescence corresponds to an increase in the fraction of highly active template sequences.

A.



B.

Round 3 (P5)	Sequence	Relative transcription efficiency
R3P5 #9	TAATACGACTCACTATA **GGCGG~** TTCCCCATCTTA	100%
R3P5 #8	TAATACGACTCACTATA **GGTAGC** TTCCCCATCTTA	29%
R3P5 #1	TAATACGACTCACTATA **GAGAAT** TTCCCCATCTTA	71%
R3P5#10	TAATACGACTCACTATA **GACTCC** TTCCCCATCTTA	36%
R3P5 #5	TAATACGACTCACTATA **GCTCA~** TTCCCCATCTTA	30%
R3P5 #6	TAATACGACTCACTATA **GTGGAA** TTCCCCATCTTA	89%
R3P5 #2	TAATACGACTCACTATA **AGGGGT** TTCCCCATCTTA	62%
R3P5 #7	TAATACGACTCACTATA **AGATAT** TTCCCCATCTTA	56%
Round 3 (P6)		
R3P6 #5	TAATACGACTCACTATA **TTCAAA** TTCCCCATCTTA	0.3%
R3P6 #2	TAATACGACTCACTATA **CTTCCC** TTCCCCATCTTA	0.5%

Figure 3. A. Histogram showing bead distribution by relative fluorescence intensity for the lib1 Round 2 population. Beads from P5 and P6 gates were collected and resulting sequences analyzed. B. Sequence and relative transcription data from select clones recovered from Round 3 of FACS. Deletions from the expected sequence are represented by a tilde (~). The region randomized in the starting sequence library is shown in bold (generally +1 to +6). Transcription efficiency is shown relative to the most transcribed clone. Sequences were selected for further characterization by *in vitro* transcription assay in order to test a variety of +1 and +2 sequence combinations. All sequences from P5 contained +1 purine +2 N (where N is any of the 4 standard nucleotides). Two clones from P6 that did not contain the +1 purine and were predicted to be inactive were tested and found to have only trace activity.

beads from these regions were amplified, cloned and 5 colonies from each region sequenced to determine the template distribution within the spectrum (Figure 2). Consistent with the addition of only 1 template for every two beads, region P9 amplified poorly compared to the other three regions. All 5 clones from region P5 and 4 of the 5 clones from region P6 were the highly active template sequence. All clones from regions P7 and P9 (together, the least fluorescent ~90% of the bead population) were weakly active templates. These results show that highly active promoter sequences can be differentially identified from a background of weakly active templates.

2.2. *Promoter selection from randomized ITS library*

After confirming that the selection scheme was viable, we wanted to select for active promoter sequences from a randomized promoter library. Starting with the core T7 phi10 promoter sequence (from -17 to -1) we completely randomized the first six bases of the initially transcribed region (ITS positions +1 to +6; lib1). These bases are known to influence promoter strength by affecting the ability of the T7 RNA polymerase to transition from the promoter recognizing initiation complex to the highly processive elongation complex[14]. This is a small library that

can be easily screened by FACS (a theoretical library size of 4^6 or 4096 possible unique members) but that we predicted would retain a broad range of activity due to the inclusion of the -17 to -1 core promoter sequence.

Similar to the initial two member test pool, the lib1 pool was attached to beads at a ratio of 1 template per 2 beads with excess capture oligonucleotide. The library was emulsified for transcription and the resulting RNA labeled beads were fluorescently labeled. Two rounds of FACS enrichment for the highest ~1.5% (round 1) and ~5% (round 2) of the population yielded the Round 2 population seen in Figure 3A. A third round of FACS was performed and two non-overlapping regions were collected (P6 and P5; Figure 3A). The templates from these regions were cloned and 9 sequences for P5 and 10 sequences for P6 were determined. Figure 3B contains a select list of sequences and relative transcription activities. For a full list of sequences, see Appendix Table A1.

The templates which were tested by *in vitro* transcription from Round 3 region P5 (R3P5) contained a broad spectrum of activities down to 30% of the maximum transcription level. The tested promoter sequences were chosen out of the 20 clones sequenced in order to cover clones containing the entire spectrum of nucleotides found at the +1 and +2 positions of the library under the theory that the +1 sequence was the most important for initiation followed closely by the +2 sequence. This expectation appears to have been borne out by the presence of a purine at +1 in 16 out of the 20 total sequences from R3P5 and R3P6 combined. Two of the four sequences that did not contain a purine at +1 were also tested by *in vitro* transcription and found to be inactive, suggesting that further enrichment of the selected pool would likely have been possible. Interestingly, the +1 and +2 positions do not appear to solely control promoter strength; variants in which these residues were the same but where the +3 to +6 sequences were different had quite different transcription activities.

3. Discussion

A number of synthetic genetic circuits have previously been designed and assayed in the context of organisms. In many instances, the overall function of these circuits had to be optimized by varying the levels of gene expression of one or more specific components. In such circuits, subtle variation of individual parts can lead to global changes in circuit behavior. Anderson et al. approached this problem by randomizing the ribosome binding site and selecting for the desired system behavior[1]. Another approach has been to screen through a small number of parts of varying strength to find one suitable for a specific application. Alper et al. used this strategy to identify promoters of varying strength to optimize metabolic pathway output. While library members spanned ~200 fold activity, changes in promoter strength of less than 10 fold were sufficient to optimize metabolite output[15]. Since the performance of biological parts in organisms is idiosyncratic, these optimizations have so far been largely empirical. To date, it has been difficult to match the actual and predicted performances of biological parts in circuits.

This problem is further confounded for synthetic biologists because a few, highly characterized parts, such as the lac and tet promoters, have typically been used again and again. More recent synthetic biology[16] and metabolic engineering[15] efforts have attempted to fine tune promoter strengths, either by screening or by design. Even greater standardization has been achieved by Kelly et al.,[17] who compared the activities of a series of *E. coli* promoters in the context of a single expression casette (GFP) *in vivo*. The establishment of such a standardized system ultimately allows the direct comparison of promoters from different labs or different constructs.

Unfortunately, such efforts will not translate to *in vitro* genetic circuits, and therefore we attempted to recapitulate such standardization in the context of *in vitro* compartments. We chose to standardize transcription using the highly active T7 RNAP and its widely used promoter. Given that the activities of proteins and regulatory sequences in emulsions can be very different from *in vivo* or even *in vitro*, we could not just assume that the previous rule sets for T7 RNA polymerase promoters would be operative in emulsion microvessicles. Indeed, we have previously found that T7 RNA polymerase loses activity in emulsions relative to *in vitro* (data not shown).

We therefore directly selected for different levels of function *in vitro* by modulating the short sequence of the initially transcribed (ITS) region and thereby the promoter strength. Promoter variants were emulsified and differing transcription activities were directly selected by FACS. We initially focused on the most highly active fraction of promoters, although a wide range of transcription activities is obviously possible. The relative transcription activities of ten different selected promoters were determined both in emulsions (via FACS) and in

transcription reactions *in vitro*. The strongest promoters generally had a guanine at +1 and favored G or A at +2, which was expected based on the known strengths of T7 TNAP promoter variants. However, we unexpectedly selected two promoters that initiated with AG, as opposed to all known phi10 promoters that initiate with G. This necessarily expands the rules for T7 RNAP promoter design *in vitro*. The 8 most active promoters spanned a dynamic range of approximately 3-fold.

The definition of a wide range of promoters with varying sequences and activities will provide a unique tool set for the construction of synthetic genetic circuits *in vitro*. The fact that only 6 residues need be varied to vary transcription is reminiscent of the varying translation by simply modulating short ribosome-binding sites, and the two techniques can obviously be used together to exquisitely control the strength of gene expression. While we have begun to standardize, characterize, and compare these new promoter parts, their behaviors can be even more accurately represented once they are tested by predictive models in complex systems.

4. Materials and Methods

4.1. *Template and library construction*

Oligonucleotides were ordered from IDT or, in the case of oligonucleotides with randomized positions, produced in house. For a full list of oligonucleotide and template sequences, see Appendix Table A2. PCR amplifications were performed with Taq DNA polymerase. For bead based experiments and sorting, ED.5Bio template F and ED.Pt7.temp R primers were used. For non-emulsion transcriptions and for cloning, ED.template F was substituted. To construct the ITS library (lib1), the template was amplified with ED.lib1 F and ED.Pt7.temp R. The product of this PCR was agarose gel purified and amplified with ED.lib1build F and ED.Pt7.temp R. The product of this PCR was gel purified and amplified with ED.5Bio template F and ED.Pt7.temp R for immobilization on streptavidin beads.

4.2. *Compartmentalized reactions*

The protocol used for this work was adapted from reference 10. All steps prior to emulsification were performed on ice and all bead centrifugation steps were carried out at 4C and 6,000 RPM for 5 minutes unless otherwise specified. See Appendix A3 for solution recipes.

20 microliters of 1 micrometer diameter streptavidin coated beads (or approximately 4E8 individual beads; Bang Laboratories) was added to 180 microliters PBSTE. The solution was gently mixed and the beads pelleted by centrifugation. The beads were resuspended in 50 microliters of PBSTE with approximately 1 template molecule for every 2 beads and incubated at room temperature for 15 minutes. 3 microliters of 20 micromolar ED.5bio.handle was added and incubated for an additional 45 minutes. The beads were centrifuged and the pellet washed with 100 microliters PBSTE once. This step was repeated two additional times but the beads were washed with 100 microliters of transcription wash buffer.

The oil phase was assembled by adding 500 microliters of oil mixture to a 13mL (95 x 16.8 mm) Starstedt polypropylene tube containing a Spinplus (9.5 x 9.5 mm Teflon) stir bar and placed in a beaker of ice on a magnetic stir plate. The bead pellet was resuspended in a 200 microliter transcription reaction immediately prior to emulsification. The transcription reaction was added in ~10 microliter aliquots over 3 minutes to the stirring oil mixture. The total stir time was 5 minutes. The reaction was transferred to a 1.5 milliliter Eppendorf tube and incubated at 37 degrees.

The transcription reaction was terminated by incubating the reaction on ice for 10 minutes with the addition of 40 microliters of 500 millimolar EDTA and 160 microliters of PBSTEBB. The emulsion was broken by the addition of 0.7 milliliters of diethyl-ether followed by thorough mixture and centrifugation at 13,000 RPM for 4 minutes at 4C. Occasionally the beads would fail to precipitate at this step and remain in the aqueous solution. In this event, the ether phase was removed, 300 microliters of PBSTE was added, gently mixed and centrifuged again. The failure of beads to centrifuge in the first step was not seen to affect the quality of the fluorescent signal.

Once the beads were pelleted, the aqueous phase was removed. The beads were washed in 100 microliters PBSTEBB 3 times before a final resuspension in 200 microliters of PBSTE.

4.3. *Bead labeling and FACS analysis*

3 microliters of 100 micromolar ED.5Fam3.probe was added and incubated on ice for 30 minutes. This fluorescently labeled DNA oligonucleotide hybridizes to the 5' half of the RNA transcript (the 3' half of the RNA hybridizes to the biotinylated capture oligonucleotide). The beads were washed twice in 100 microliters PBSTEBB to remove non-hybridized ED.5Fam3.probe. The fluorescence from the first labeling was insufficient to differentiate active from inactive beads by FACS so an antibody based fluorescence amplification reagent was utilized (Alexa Fluor 488 Signal Amplification Kit, Invitrogen). The beads were resuspended in 200 microliters Antibody Labeling Buffer with 10 milligrams of rabbit anti-fluorescein antibody and incubated for 20 minutes on ice, followed by two 100 microliter PBSTEBB washes. The beads were then resuspended in 200 microliters Antibody Labeling Buffer with 10 milligrams of goat anti-rabbit IgG antibody and incubated on ice for 20 minutes. The beads were washed twice in 200 microliters of PBSTEBB and finally resuspended in 100 microliters PBS.

4.4. *Compartmentalized selection*

Fluorescence based bead sorting was performed on a BD FACSAria Cell Sorting System. Beads were diluted additionally in PBS to an appropriate concentration for sorting. Fractions were collected and amplified by PCR for additional sorting and for cloning and sequencing, as seen in Figure 2 and Figure 3.

4.5. *Cloning and sequencing*

After amplification of templates from sorted beads, PCR products were cloned using a Topo TA cloning kit (PCR4topo, Invitrogen). Individual colonies were chosen for sequencing and to generate PCR templates for transcription analysis.

4.6. *Transcription analysis*

In vitro transcription was performed on selected clones. Reactions were setup similar to the selection protocol, except 0.01 mCi of alphaP^{32}-ATP was added, the template was not immobilized and the total reaction volume was 20 microliters. 50 nanograms of a PCR template was added to each transcription and incubated at 37 degrees for 40 minutes. The reaction was stopped by adding 20 microliters of 18 millimolar EDTA in formamide and heated at 90 degrees for 10 minutes. Transcription products were run on a denaturing (7 M urea) 8% polyacrylamide gel which was then dried and analyzed using a Phosphorimager (Molecular Dynamics).

Acknowledgments

This work was funded by the NIH (National Institutes of Health). We would like to acknowledge the Welch Foundation's (F1654) continued support for the efforts of Dr. Andrew D. Ellington.

Appendix

Table A1. Sequences cloned from R3P6 and R3P5 of the Lib1 selection.

R3P5	
R3P5- 9	TAATACGACTCACTATAGGCGG~TTCCCC
R3P5- 8	TAATACGACTCACTATAGGTAGCTTCCCC
R3P5- 1	TAATACGACTCACTATAGAGAATTTCCCC
R3P5- 10	TAATACGACTCACTATAGACTCCTTCCCC
R3P5- 5	TAATACGACTCACTATAGCTCA~TTCCCC
R3P5- 6	TAATACGACTCACTATAGTGGAATTCCCC
R3P5- 4	TAATACGACTCACTATAGTAGTATTCCCC
R3P5- 2	TAATACGACTCACTATAAGGGGTTTCCCC
R3P5- 7	TAATACGACTCACTATAAGATATTTCCCC
R3P6	
R3P6- 7	TAATACGACTCACTATAGGATTGTTCCCC
R3P6- 3	TAATACGACTCACTATAGAATAATTCCCC
R3P6- 1	TAATACGACTCACTATAGCTCACTTCCCC
R3P6- 8	TAATACGACTCACTATAGCAACTTTCCCC
R3P6- 9	TAATACGACTCACTATAGCACCCTTCCCC
R3P6- 4	TAATACGACTCACTATAAGTCGCTTCCCC
R3P6- 2	TAATACGACTCACTATACTTCCCTTCCCC
R3P6- 6	TAATACGACTCACTGTACTTAGATTCCCC
R3P6- 10	TAATACGACTCACTATACTTAGATTCCCC
R3P6- 5	TAATACGACTCACTATATTCAAATTCCCC

Table A2. Template and oligonucleotide sequences.

Name	Sequence
highly active template	GTCGACAAGCTTGCGGCCGCATAATGCTTAAGTCGAACAGAAAGTAATCGTATTGTACACGGCCGCATAATCGAAAT taatacgactcactatagggaatTCCCCATCTTAGTATATTAGTTAAGTATAAGAAGGAGATATACATATGAGCCA TATTCAACGGGAAACGTCTTGCTCTAGGCCGCGATTAAATTCCAACATGGATGCT
weakly active template	GTCGACAAGCTTGCGGCCGCATAATGCTTAAGTCGAACAGAAAGTAATCGTATTGTACACGGCCGCATAATCGAAAT taatacgactcactatacctattTTCCCCATCTTAGTATATTAGTTAAGTATAAGAAGGAGATATACATATGAGCCA TATTCAACGGGAAACGTCTTGCTCTAGGCCGCGATTAAATTCCAACATGGATGCT
lib1 template	GTCGACAAGCTTGCGGCCGCATAATGCTTAAGTCGAACAGAAAGTAATCGTATTGTACACGGCCGCATAATCGAAAT taatacgactcactatannnnnnTTCCCCATCTTAGTATATTAGTTAAGTATAAGAAGGAGATATACATATGAGCCA TATTCAACGGGAAACGTCTTGCTCTAGGCCGCGATTAAATTCCAACATGGATGCT
ED.lib1build F	GTCGACAAGCTTGCGGCCGCATAATGCTTAAGTCGAACAGAAAGTAATCGTATTGTACACGGCCGCATAATCGAAATtaatacgactcac
ED.5bio template F	Bio-GTCGACAAGCTTGCGGCCGCATAATG
ED.template F	GTCGACAAGCTTGCGGCCGCATAATG
ED.PT7.temp R	AGCATCCATGTTGGAATTTAATCGCGGCCTAGAGC
ED.5Fam3.probe	6Fam-AGAGGTTTCCGTTGAATATGGCTCATATGTATATC-6Fam
ED.5bio.handle	Bio-AGCATCCATGTTGGAATTTAATCGCGGCCTAGAGC
ED.lib1 F (ITS)	ATCGAAATtaatacgactcactatannnnnnTTCCCATCTTAGTATATTA

A3. Solutions.

PBSTE	10mM	NaPO4 pH 7.4
	150mM	NaCl
	0.1%	Tween-20
	10mM	EDTA
PBSTEB	PBSTE plus 0.1% BSA	
PBSTEBB	PBSTE plus 0.1% BSA and 100uM biocytin	
Antibody Labeling Buffer	PBSTE plus 100uM biocytin, 1% BSA, 100ug tRNA (1uL), 80U RNAsin (2uL) and 10ug Antibody (5uL) per 200uL	
Transcription Buffer	50mM	EPPS pH 8.0
	2mM	Spermidine
	10mM	DTT
	50mM	KCl
	30mM	MgCl2
Transcription Mix	200uL total Transcription buffer plus 4U yeast pyrophosphatase, 160U RNAsin Plus, 1000U T7 RNAP and 2mM NTPs	
Transcription Wash Buffer	Transcription buffer plus 0.1% BSA	
Oil Mixture	475uL	Mineral Oil
	22.5uL	Span 80
	2uL	Tween 80
	0.25uL	Triton X-100

References

1. J.C. Anderson, C.A. Voigt and A.P. Arkin, *Mol Syst Biol.* **3**, 133 (2007).
2. S.W. Santoro, L. Wang, B. Herberich, D.S. King and P.G. Schultz, *Nat Biotechnol*. **20**, 1044 (2002).
3. J. Chelliserrykattil, A.D. Ellington, *Nat Biotechnol*. **22**, 1155 (2004).
4. S. Tabor, *Curr Protoc Mol Biol*. **Unit 16.2** (2001).
5. A.C. Forster and G.M. Church, *Mol Syst Biol*. **2**, 45 (2006).
6. V. Noireaux, R. Bar-Ziv and A. Libchaber, *Proc Natl Acad Sci U S A.* **100**, 12672 (2003).
7. K. Ishikawa, K. Sato, Y. Shima, I. Urabe and T. Yomo, *FEBS Lett.* **576**, 387 (2004).
8. M. Isalan, C. Lemerle and L. Serrano, *PLoS Biol.* **3**, e64 (2005).
9. D.S. Tawfik and A.D. Griffiths, *Nat Biotechnol.* **16**, 652 (1998).
10. M. Levy, K.E. Griswold and A.D. Ellington, *RNA*. **11**, 1555 (2005).
11. M.C. Jewett, K.A. Calhoun, A. Voloshin, J.J. Wuu and J.R. Swartz, *Mol Syst Biol*. **4**, 220 (2008).
12. J. Kim, K.S. White and E. Winfree, *Mol Syst Biol*. **2**, 68 (2006).
13. N. Mishima, K. Mizumoto, Y. Iwasaki, H. Nakano and T. Yamane, *Biotechnol Prog*. **13**, 864 (1997).
14. W.P. Kennedy, J.R. Momand and Y.W. Yin, *J Mol Biol*. **370**, 256 (2007).
15. H. Alper, C. Fischer, E. Nevoigt and G. Stephanopoulos, *Proc Natl Acad Sci U S A.* **102**, 12678 (2005).
16. T. Ellis, X. Wang and J.J. Collins, *Nat Biotechnol.* **27**, 465 (2009).
17. J.R. Kelly, A.J. Rubin, J.H. Davis, C.M. Ajo-Franklin, J. Cumbers, M.J. Czar, K. de Mora, A.L. Glieberman, D.D. Monie, D. Endy, *J Biol Eng.* **3**, 4 (2009).

CLUSTERING CONTEXT-SPECIFIC GENE REGULATORY NETWORKS*

ARCHANA RAMESH, ROBERT TREVINO

School of Computing, Informatics & Decision Systems Engineering,
Arizona State University, 699 S Mill Avenue, Tempe, AZ 85281, USA

DANIEL D. VON HOFF

Clinical Translational Research Division, Translational Genomics Research Institute,
445 North Fifth Street, Phoenix, AZ 85004, USA

SEUNGCHAN KIM

School of Computing, Informatics & Decision Systems Engineering,
Arizona State University, 699 S Mill Avenue, Tempe, AZ 85281, USA
Computational Biology Division, Translational Genomics Research Institute,
445 North Fifth Street, Phoenix, AZ 85004, USA
E-mail: dolchan@asu.edu

Gene regulatory networks (GRNs) learned from high throughput genomic data are often hard to visualize due to the large number of nodes and edges involved, rendering them difficult to appreciate. This becomes an important issue when modular structures are inherent in the inferred networks, such as in the recently proposed context-specific GRNs.[12] In this study, we investigate the application of graph clustering techniques to discern modularity in such highly complex graphs, focusing on context-specific GRNs. Identified modules are then associated with a subset of samples and the key pathways enriched in the module. Specifically, we study the use of Markov clustering and spectral clustering on cancer datasets to yield evidence on the possible association amongst different tumor types. Two sets of gene expression profiling data were analyzed to reveal context-specificity as well as modularity in genomic regulations.

Keywords: Markov clustering; spectral clustering; cancer; gene regulatory networks; cellular context

1. Introduction

A cell maintains a specific state (such as 'healthy') by tightly regulating a set of molecules. When exposed to environmental changes, the cell adjusts its regulatory mechanisms and transitions to a state (such as 'tumor') significantly different from the original state. Since the manner in which the system reacts to inputs is altered, we term this as a change in cellular context.[8]

Kim et al.[12] have proposed an algorithm which uses a probabilistic framework to learn contexts from gene expression data. More recently, Sen et al.[19] have applied this method to identify context-specific gene regulatory networks (GRNs). Unlike conventional GRNs, edges in context-specific GRNs represent the interaction conditioned on a subset of samples, i.e. *their biological context*, thus lending adaptability to the model of biological regulation.

However, GRNs learned by the algorithm are often made of a few thousand nodes (genes) and tens of thousands of interactions rendering manual curation of the edges and sub-network to identify its modular structure and context-specificity quite difficult if not impossible. Hence, the need for the automatic extraction and curation of relevant *context clusters* from these networks is critical.

Graph clustering is defined as the task of grouping the vertices of a graph in such a way that there are many edges within a cluster and relatively few edges between the clusters.[18] The most significant difference between conventional clustering and graph clustering is in the notion of the relationship between the elements being clustered. When similarity is expressed through whether elements "share a property" or not (such as a regulatory relationship where genes are co-regulated), rather than the distance between the elements, graph

*Code, scripts and supplementary information available online at http://sysbio.fulton.asu.edu/publications/2010/psb2010/

clustering is appropriate for the problem. Moreover, as the problem of clustering GRNs is also directly related to the connectivity between nodes, we believe that graph clustering is better suited for solving this problem.

Our work looks into the applicability of two graph clustering algorithms, namely Markov clustering and spectral clustering in identifying the modular structure of context-specific GRNs. Markov clustering was chosen due to its scalability and ability to automatically determine the number of clusters. Spectral clustering was chosen due to its ability to find an optimal minimum cut while creating well-balanced clusters. In addition, previous applications in the bioinformatics field have yielded promising results,[7,10,17,23] leading us to believe it would be well-suited for this problem.

Our paper is organized as follows. We begin with an overview of the existing applications of graph clustering to bioinformatics. Following this, we provide a mathematical formulation of our problem and then describe the graph clustering methods and enrichment analysis techniques that we apply. Subsequently we demonstrate how our methods could be applied to yield insights on the underlying mechanisms of cancer. Finally, we conclude with the future direction of our work.

2. Relevant Work

Clustering is defined as the unsupervised classification of patterns (observations, data items, or feature vectors) into groups (clusters).[11] The clustering task usually involves pattern representation, definition of pattern proximity, clustering or grouping, data abstraction and assessment.[5]

Both Markov clustering and spectral clustering have been previously applied to bioinformatics. Lattimore et al.[17] have applied MCL to the analysis of microarray data using a graph constructed from the correlation of gene expression measurement to which MCL is applied. The algorithm has been applied to a breast cancer dataset and shown to identify underlying biological mechanisms. In a similar study,[7] Freeman et al. have used MCL in the clustering and visualization of transcription networks from microarray data. This work focuses on applying MCL to transcription networks derived from mouse gene expression data, again using the Pearson's correlation coefficient.

Spectral clustering has been used in the analysis of gene expression data. Clustering has been performed in terms of genes, samples and both dimensions. While Tritchler et al.[23] apply Eigen analysis (spectral methods) to cluster genes from gene expression data, Higham et al.[10] apply spectral methods to cluster gene expression data based on samples. Kluger et al.[13] have shown that the eigenvectors in matrices of gene expression data formed a distinct "checkerboard" pattern that can be exploited to simultaneously cluster genes and conditions in cancer datasets. In all cases, while our method shares similar objectives, our work differs in the sense that we apply spectral clustering to the extracted context-specific gene regulatory networks as a graph clustering approach.

The primary contributions of our work include developing methods for clustering the recently proposed context-specific gene regulatory networks. Context-specific gene regulatory networks provide a means to specify genomic regulations conditioned by a subset of samples. Secondly, our work focuses on comparisons between Markov clustering and two variants of Spectral clustering, suited for both symmetric and asymmetric graphs. Comparisons are performed in two dimensions- using performance measures such as coverage and performance which measure the goodness of the obtained clusters and using enrichment analysis which allows for the biological interpretation of the results. Finally, both algorithms are applied to two cancer gene expression datasets yielding insights on possible associations between tumor-types and several useful clinical implications.

3. Problem Definition

A cellular context is defined as a set of genes, one or more of which function as drivers and the rest as driven genes, exhibiting consistent transcriptional behavior across a set of samples, drawn from a cellular process governed by tightly regulated mechanism(s) involving the set of genes. Mathematically, a context can be represented as $C_i = (G_i, Y_i, S_i, M_i)$ where G_i represents a set of driver genes, Y_i represents the possible

states of the genes (an example would be $\{-1, 0, +1\}$ for a ternary quantized dataset), S_i represents a set of driven genes and M_i represents the set of samples under which consistent expression is observed.

Each context defines regulatory relationships between the driver genes and the driven genes, i.e. $G_i \rightarrow g \in S_i$, specific to M_i with G_i (drivers) conditioned on a specific state $Y_i = y_i$. A driver g_j in context C_j could be driven by g_i in another context C_i. When such relationships are added to the implicit driver-driven relationships $g_i \rightarrow g_j$, we obtain an interesting graphical structure representing the relationships between contexts. We call this graph, a *context-specific GRN* as each regulatory relationship $g_i \rightarrow g \in S_i$ is specific to a subset of samples M_i.

In graph theoretic terms, a context-specific GRN is a directed graph $G = (V, E)$ where V is a set of vertices representing genes and E is a set of edges representing context-specific driver-driven relationships.

A partition of the graph into two non-empty sets S and $V \setminus S$ is called a *cut* and denoted by $(S, V \setminus S)$. Usually a cut is uniquely defined by a set S, and hence any sub-set of V can be called a cut. The *cut-size* is the number of edges that connects vertices in S to those in $V \setminus S$.

Given a context-specific GRN $G = (V, E)$ as defined above, our goal is to determine the clusters of the network; where a cluster C may be defined as an induced subgraph of the graph, such that $C = (V_c, E_c)$, where $V_c \in V, E_c \in E$; (i) for every edge $(u, v) \in E_c$, $u \in V_c$ and $v \in V_c$ and (ii) the cut size of the cluster C is minimal.[18]

4. Methods

Contexts are learned from gene expression data through the cellular context mining algorithm.[12] Given a gene g_k and a cellular context c_j defined by a subset of samples M_j, the algorithm uses probabilistic measures to identify a set of genes with consistent expression levels within the context. The resulting contexts dictate implicit driver-driven relationships. When these relationships are captured in the form of a graph, we obtain a context-specific GRN.[19] In our study, we used a variant of this context-specific GRN, where each node was the driver of a context. The set of genes regulated by driver S_j was used in the enrichment analysis.

4.1. *Markov Clustering*

Markov clustering derives its inspiration from the notion of random walks in graphs. If a random walk visits a node in a cluster, it would be likely to visit several other members of the cluster before leaving the cluster.[24]

The Markov clustering algorithm simulates flow using two (alternating) algebraic operations on matrices. Expansion (identical to matrix multiplication) represents the homogenization of flow across different regions of the graph. Inflation, mathematically equivalent to a Hadamard power followed by diagonal scaling, represents the contraction of flow, making it thicker in regions of higher current and thinner in regions of lower current. Intuitively, expansion corresponds to augmenting the neighbors of a given vertex, and inflation corresponds to promoting those neighbors which have a higher transition probability from a given vertex. The MCL process causes flow to spread out within natural clusters and disappear in between different clusters.[24] The iteration is continued until a recurrent state or fixpoint is reached. The exact steps are explained in Algorithm 4.1. The connected components of the graph induced by the non-zero entries of M provide the required clustering. Proof of concept, mathematical properties and analyses on the complexity and scalability of the algorithm can be found in van Dongen's work.[25] Our implementation of Markov clustering used the publicly available tool BioLayout Express.[7]

4.2. *Spectral Clustering*

Spectral clustering uses the Eigen decomposition of matrix representations of a graph to determine the optimal partitioning of the graph. Although, there has been extensive research in the spectral clustering field, we used the algorithms developed by Shi and Malik[20] and Meila and Pentney[15] because they incorporate information from the edges (in our case, computationally predicted biological interactions) in determining the optimal clustering of a graph.

Algorithm 4.1 Markov Clustering

Input: $G = (V, E)$, expansion parameter e, inflation parameter r
while M is not fixpoint **do**
  $M \leftarrow M^e$
  for all $u \in V$ **do**
    for all $v \in V$ **do**
      $M_{uv} \leftarrow M_{uv}^r$
    end for
    for all $v \in V$ **do**
      $M_{uv} \leftarrow \frac{M_{uv}}{\sum_{w \in V} M_{uw}}$
    end for
  end for
end while

4.2.1. *Symmetric Cuts:*

In graph theory, a cut is defined as

$$cut(A, B) = \sum_{u \in A, v \in B} w_{uv}, \tag{1}$$

where A and B are the clusters resulting from the cut between vertices u and v. Finding the minimum cut for Equation 1 could result in singletons or clusters with very few nodes, leading to poorly distributed clusters. Thus, there exists a need to balance the clusters. Shi and Malik, have proposed a solution to this problem by normalizing the cuts that create clusters.[20] The cut cost is calculated as a fraction of the weights of the edges in the induced sub-graphs. As finding the exact solution to the normalized minimum cut problem is considered NP-complete, the authors have found that using the eigenvector corresponding to the second smallest eigenvalue of the Laplacian of an undirected graph (also known as the Fiedler vector) could efficiently provide an approximate discrete solution.[20] The algorithm, referred to as the normalized cut algorithm, recursively splits clusters thresholding the Fiedler vector of the induced sub-graphs until the desired number of clusters are reached.

4.2.2. *Asymmetric Cuts:*

Meila and Pentney[15] provide for the expansion of spectral clustering in multi-way cuts to directed graphs, as the normalized cut is applicable only to undirected graphs. In gene regulation directionality could provide useful information. The weighted cut algorithm, proposed by Meila and Pentney, mathematically transforms a directed graph (with a non-normalized Laplacian matrix, D-A), into a symmetric Hermitian matrix[15] and finds an approximate solution to minimizing a normalized cut. Using the k eigenvectors pertaining to the k smallest eigenvalues of the Hermitian matrix, the weighted cut algorithm applies the k-means algorithm to cluster the graph. In addition, the algorithm allows for user input, balancing parameters T and T' , to normalize the cuts produced by the algorithm. Thus the normalized minimum cut for directed graphs can be expressed as:

$$MNCut(x) = \min_{z_k \in R^n \text{orthon}} \sum_{k=1}^{K} z_k^* H(B) z_k \tag{2}$$

where $B = T^{-\frac{1}{2}}(D - A)T^{-\frac{1}{2}}$, K is the number of desired clusters and $H(B)$ is the Hermitian matrix of B.

4.3. *Enrichment Analysis*

Subsequent to clustering the context-specific GRNs, it is interesting to study the pathways and phenotypic characteristics that the resulting clusters are enriched with. To this end, we employ the following mechanisms

to evaluate the biological significance of the obtained clusters.

4.3.1. *Gene Set Enrichment Analysis:*

We investigate the enrichment of each context cluster using gene sets. The hypergeometric test is used to measure the significance of the enrichment and the p-values are corrected for False Discovery Rate (FDR) using Benjamini and Hochberg's method.[1] The Molecular Signatures Database (MSigDB) is used as a reference knowledge source.[22] MSigDB contains a collection of gene sets, including positional gene sets, curated pathways, conserved motifs, computationally predicted expression neighborhoods (defined on 380 cancer-associated genes) and Gene Ontology gene sets.

4.3.2. *Tumor Type Enrichment Analysis:*

Sample Association Score: As the context clusters derived from the clustering process consist of a set of cellular contexts, it is relevant to study the samples that occur in more than one context within each context cluster. Samples were scored based on their occurrence within each context, over all the contexts within the cluster. A sample s, given a context cluster CC with m contexts $C_1, C_2, \cdots, C_m$, would have the scoring

$$\mathrm{SAS}(s, CC) = \sqrt[m]{\prod_{i=1}^{m} f_i(s)}, \tag{3}$$

where $f_i(s) = k_i/N$, when $s \in C_i$ and 1, otherwise.

Motivated by the differences in the number of samples in each context, a sample belonging to a larger context would have a lesser contribution to the score than a sample belonging to a smaller context. Only samples that had a sample association score < 0.5 were considered. Following this, the context clusters were analyzed for enrichment of specific tumor types using the Hypergeometric test. FDR correction was applied using Benjamini and Hochberg's correction method.[1]

5. Results

We applied our methods to two gene expression datasets – the Target Now dataset and the REMBRANDT study. In the following section, we discuss the study that was conducted, the results obtained including biological significance and performance comparisons of the three algorithms.

5.1. *Target Now Data*

Our input graph constituted a variant of the context-specific GRN produced by Sen et al.[19] from the Target Now (TN) dataset; a study aimed at determining if patients with refractory cancer, who did not benefit from the standard types of treatment, could derive benefit from therapy with a drug not normally used for their particular form of cancer.[26]

The dataset consists of 17,085 unique probes (Agilent-011521 Human 1A Microarray G4110A) from 146 patients with different types of refractory cancer. We used the graphs corresponding to the relationships derived from statistically significant contexts (using a p-value < 0.001). The graph consisted of 391 contexts and was organized into six strongly connected components.

5.1.1. *Markov Clustering*

As Markov clustering has a propensity towards undirected graphs, we used the undirected version of the context-specific GRN obtained from the filtered contexts. Clustering was performed on the graph using an inflation value of 2.0. The inflation parameter is used to control the granularity of the clusters obtained, and was set to 2.0 as it provided the desired granularity. Clusters with less than 3 nodes were not considered.

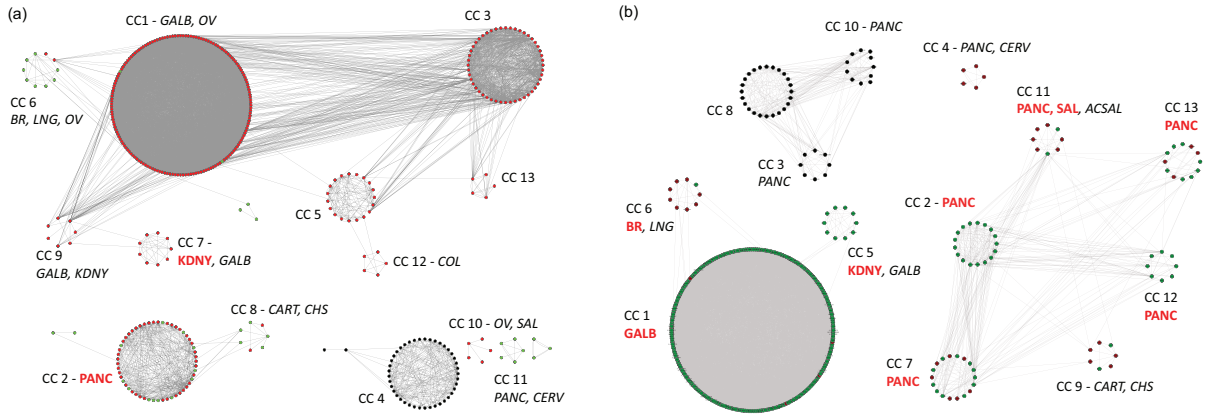


Fig. 1: (a) Markov Clustering Results [TN study]. (b) Asymmetric Spectral Clustering Results [TN study]. The following acronyms indicate enriched tumor types - BR:Breast, LNG:Lung, OV:Ovarian, PANC:Pancreas, GALB:Gall Bladder, CART:Cartilage, CHS:Chondrosarcoma, COL:Colon, SAL:Salivary, KDNY:Kidney. Tissue types in red indicates the type is over-represented in the corresponding context-cluster with *adjusted* p-value < 0.05. Tissue types with italicized font indicate the tissue type is over-represented in the corresponding context-cluster with p-value < 0.05.

The clusters obtained are shown in Figure 1(a). As seen in the illustration, the algorithm identified thirteen distinct clusters, dividing the six strongly connected components into smaller clusters. Further, we also note that within most clusters, the expression levels of all genes belonging to the cluster are similar.

Nine out of thirteen clusters were found to be enriched with several tumor types. Further discussion of the biological significance of these results can be found in Section 5.1.3. The performance of the algorithm, along with a discussion of the relevant MSigDB terms identified within the clusters is outlined in section 5.4.

5.1.2. *Spectral Clustering*

Figure 1(b) shows the clustering results obtained when the weighted cut algorithm was applied to the directed version of the context-specific GRN. The desired number of clusters was set to 13, based on the number of clusters obtained from the MCL study, to allow for comparisons between the two algorithms. The out-degree was chosen to normalize the cut as interaction is known to be a key aspect of biological networks. Further we were interested in studying if the incorporation of direction in normalizing the cut, would provide better results compared with clustering on undirected graphs.

As seen in Figure 1(b), eight out of thirteen clusters were found to be significantly enriched with the different tumor types. The biological significance of these results is elaborated upon in Section 5.1.3.

The normalized cut algorithm (symmetric spectral clustering) was found to produce results similar to the MCL cluster assignments. Comparing the results from the symmetric clustering algorithms with asymmetric spectral clustering, we note that the clusters produced by the asymmetric variant are more balanced.

It is interesting to note that the tumor type associations derived from these cluster assignments (Figures 1(a) and (b)) corroborate the evidence obtained by Sen et al.[19] We augment the authors' findings with a further refinement of the clusters and possible associations amongst them.

5.1.3. *Biological Validation*

Following graph clustering, the obtained results were analyzed for tumor type and gene set enrichment using the methods described in Section 4.3. Significant terms were considered based on an adjusted p-value cut-off of 0.05 following which the terms were filtered based on a minimum enrichment ratio criterion of 0.1. Terms were then grouped on the basis of the source of the annotation. Pathways, Gene Ontology associations and

Table 1: Biological Significance of Clustering Results of the TN Dataset. Tumor types in bold indicate tumor types enriched with an *adjusted* p-value < 0.05. MSigDB Terms in bold indicate terms unique to the context cluster (when compared with other context clusters obtained using the same method). Acronyms in brackets indicate source of annotation (G:GenMAPP, K:KEGG, B:BioCarta, T:TRANSFAC, GO BP:Gene Ontology Biological Process, SA:SigmaAldrich, ST:Signaling Transduction KE)

Context Cluster & Tumor Type	*Relevant MSigDB Terms*
Markov Clustering	
CC2[**Pancreas**]	**Apoptosis, Purine & Pyrimidine Metabolism[G]**
	Gycolysis & Gluconeogenesis*[G], Ribosome[K]
	FAS Signaling Pathway*[B]
	AP2 Family TF Binding Site[T], MTA3, RAR-RXR, MTOR Pathways[B]
	Oxidative phosphorylation[G], Proteasome[K]
	Positive Regulation of Signal Transduction[GO BP]
CC6[Breast, Lung, Ovarian]	**Classic Pathway[B], Comp Pathway[B]**
	Complement Activation Classical[G]
CC8[Cartilage, Chondrosarcoma]	**Methane Metabolism[G], Cholera Infection[K]**
	Stilbene Coumarine and Lignin Biosynthesis[G]
CC10[Salivary, Ovarian]	**EI2F Pathway[B], Translation Factors[G]**
Spectral Clustering Asymmetric	
CC2[**Pancreas**]	**SA G1 & S Phases[SA], MTOR & PTDINS Pathway[B],**
	Gycolysis, RAR-RXR and MTA3 Pathway[B],
CC7[**Pancreas**]	**Gycolysis & Gluconeogenesis*[G, K] FAS Signaling Pathway*[ST]**
	Ribosome Pathway[K], Pyrimidine Metabolism, Proteasome[G],
	Regulation of JAK-STAT Cascade, Nuclear Export, RNA Splicing[GO BP]
CC12[**Pancreas**]	**TNFR1, IL2 2BP Pathway[B], Starch & Sucrose Metabolism*[G],**
	IL6 Pathway[B], Gycolysis Pathway*[B], MTA3 Pathway[B]
CC13[**Pancreas**]	**Actin Y Pathway[B], Translation Factors[G], ECM Pathway[B]**
	I Kappa B Kinase NFKappa B Cascade[GO BP],
	Glucose Catabolic Process*[GO BP], Proteasome[G],
	MTA3 Pathway, Gycolysis Pathway*[B], Oxidative phosphorylation*[G]
CC11[**Pancreas, Salivary**, ACSAL]	**BAD Pathway[B]**,
	Pathogenic *E.Coli* Infection EHEC & EPEC[K], Gycolysis Pathway*[B]
CC1[**Gall Bladder**]	Ribosome Pathway[K], Proteasome Pathway[K],
	mRNA Metabolic Process, mRNA processing[GO BP],
	Mitochondrion, Nuclear Part[GO Cellular Component],
	Oxidative phosphorylation[G]
CC6[**Breast**, Lung]	**Classic & Comp Pathway[B],, Complement Activation Classical[G]**
CC9[Cartilage, Chondrosarcoma]	**Methane Metabolism[G], Cholera Infection[K],**
	Stilbene Coumarine and Lignin Biosynthesis[G]

transcription factors found to be relevant to the study, along with the context clusters in which they were enriched are shown in Table 1. Context clusters enriched with tumor types but not significantly associated with MSigDB terms are not listed in the table. A complete version of the results is available on our website along with our supplementary materials.

Markov Clustering: Of the thirteen context clusters produced by MCL, CC2 and CC7 were found to be enriched with pancreatic and kidney tumors respectively (using the adjusted p-value). We also found that Symmetric Spectral clustering produced similar results. Upon closer examination of the contexts forming a part of the two clustering results, we observe that more than 90 % of the cluster assignments were identical. As seen in Table 1, CC2 (MCL) was found to be enriched with the transcription factor binding site for the AP2 family of proteins, known to play a role in the repression of pancreatic cell proliferation.[6]

Asymmetric Spectral Clustering: Tumor type enrichment analysis of the cluster assignments produced by Asymmetric Spectral Clustering yielded twelve out of the thirteen clusters enriched with different tumor types. CC6 consists of members of the HLA family; HLA-DM, whose expression when combined with that of HLA-DR, is considered to influence breast tumor progression and patient outcome.[16] Enriched subsets of

the genes were also found to participate in the BRG1-induced tumor arrest in breast cancer cells.[9]

Of the remaining clusters, CC 2, 7, 11, 12, 13 were all found to be enriched with pancreatic tumor. In addition, CC11 was also enriched with salivary gland tumor, and ACSAL. It is of note that pancreatic cancer and salivary gland tumor are both tumors from secretory organs which secrete certain common enzymes (eg. Amylose). As seen in Table 1, several pathways known to play a role in tumor progression were enriched in these clusters. The *pyrimidine metabolism pathway* activity seen in CC2 could indicate a subset of pancreatic cancers which would be responsive to the agent Gemcitabine, which has been shown to improve survival for patients with pancreatic cancer (about 5 – 14 percent patients respond).

All clusters enriched with any tumor type, contained several gene sets enriched with cancer modules and cancer gene neighborhoods as defined by Brentani et al.[3] Of particular interest is the finding of a great deal of activation of metabolic genes (indicated in Table 1 by an asterisk), which is consistent with the linking of tumor cell metabolism as a target in pancreatic cancer.[21] The clinical implications of the results are further elaborated in Section 5.3

5.2. *REMBRANDT Data*

In order to test the scalability of the algorithms, and applicability to other studies, we applied the graph clustering algorithms to the REMBRANDT dataset. REpository for Molecular BRAin Neoplasia DaTa (REMBRANDT)[14] is a knowledge base consisting of clinical and functional genomics data from clinical trials involving patients suffering from gliomas. Gene expression data was collected from 417 different glioma tissue samples, and analyzed using the Affymetrix HG U133 Plus 2 microarray chip.

The raw expression values were quantized on the basis of two fold changes, and then filtered to remove transcripts with no change across all samples. Following this, the cellular context mining algorithm was applied in order to extract meaningful contexts. Statistically significant contexts (p-value < 0.0005) were then used in the construction of the graph. The resulting graph consisted of 1,901 nodes and 33,820 edges.

MCL was used with the same parameters and resulted in 32 clusters of varying sizes. Spectral clustering using the normalized-cut algorithm was applied to the undirected graph and the weighted-cut algorithm was applied to the directed graph. In both cases, the number of desired clusters was set to 32. Clusters having fewer than three nodes were not considered. Subsequently, gene and sample enrichment analysis was performed on the resulting clusters.

Tumor type enrichment and gene set enrichment analysis were conducted using the methods described in Section 4.3. Significant terms were considered based on a corrected p-value cut-off of 0.05 following which the terms were filtered based on a minimum enrichment ratio criterion of 0.1.

Table 2 shows the tumor type enrichments of the clusters obtained using MCL (on the undirected graph) and the Spectral Weighted Cut algorithm (on the directed graph). The table lists out the context clusters that were significantly enriched with each tumor type. It is to be noted that the numbering of context clusters does not match across different methods. As seen in the table we note that out of the 32 context clusters that the two algorithms produced, eleven (in the case of MCL) and ten (in the case of Spectral Asymmetric) have been enriched with tumor type associations. Interestingly, the tumor type Astrocytoma was significantly associated with Oligodendroglioma in at least two clusters.

An analysis of the MSigDB terms enriched in the clusters showed several MSigDB terms including *Cell Signaling Pathways, Cell Cycle, Cell Adhesion, Apoptosis, Regulation of DNA Replication, the E2F transcription factor pathway and the ERK Pathway.* These terms are linked to cell growth and proliferation characterizing tumor behavior. The context clusters were also found to be enriched with several cancer modules.[3] Detailed results are provided on our website along with other supplementary information.

5.3. *Clinical Implications of the Results*

The data generated by these two graph clustering methods should be of interest to clinical investigators because they have potential clinical implications. To begin with pancreatic cancer, it is comforting to see

Table 2: REMBRANDT Tumor Type Enrichment (Using Context-Specific GRN with p-value < 0.0005): Tumor types in bold indicate tumor types enriched with an *adjusted* p-value < 0.05. MSigDB Terms listed consist of terms unique to the context cluster (when compared with other context clusters obtained using the same method). Please refer to Table 1 for annotation sources. Tumor types include Glioblastoma(GBM), Astrocytoma(Astro), Oligodendroglioma(Oligo) and Mixed.

Context Cluster & Tumor Type	*Relevant MSigDB Terms*
Markov Clustering	
CC5[**GBM**]	Cell Cycle, ATR- BRCA, PLK3, P27, MPR, SKP2 E2F, G1 Pathways[B], DNA Polymerase[K] G1 to S Cell Cycle Reactome, DNA Replication[G], Pyrimidine Metabolism[K, G]
CC31[**GBM**]	HIF Pathway[B]
CC6[Astro]	IL 12, TC Apoptosis Pathway[B], Breast Cancer Estrogen Signaling[GE], Peptide GPCRS[G], Toll Like, B Cell & T Cell Receptor Signaling Pathways[K], N Glycan Degradation[G], Hematopoitic Cell Lineage, Cytokine Cytokine Receptor Interaction[K], FC Epsilon RI Signaling Pathway, Natural Killer Cell Mediated Cytotoxicity[K], JAK STAT Signaling Pathway, Arachidonic Acid Metabolism[K], Leukocyte Transendothelial Migration, N Glycan Degradation[K]
CC7[Astro]	PIP3 Signaling in B Lymphocytes[SIG], Inflam Pathway[B], IL13 Pathway[ST], PML, AS B Cell, BB Cell, IL5, SODD Pathway[B], Glycosaminoglycan Degradation[B, K] B Cell Receptor Complexes[SA], Eosinophils Pathway[B], BCR Pathway[B, SIG], B Cell Antigen Receptor[ST], Interleukin 13 Pathway[ST], Alzheimer's Disease[K],
CC12[**Astro, Oligo**]	Keratan Sulfate Biosynthesis[K]
CC14[**Astro, Oligo**]	RECK Pathway[B], ERK Pathway[B]
CC27[**Oligo**, Mixed]	IL2 2BP Pathway[B], IL10 Pathway[B]
Spectral Clustering Asymmetric	
CC11[GBM]	PML, Eosinophils, AS B Cell, IL5 Pathways[B], Prostaglandin Synthesis Regulation Pathway[G]
CC13[GBM]	ATR BRCA, PLK3, P27, G1 Pathways[B], DNA Polymerase[K], G1 to Cell Cycle Reactome[G], Pyrimidine Metabolism[G, K], DNA Replication Reactome[G], P53 Signaling Pathway[K],
CC19[Astro]	IL12, CSK, T Cytotoxic, D4GDI, NKT, CTL Pathway[B], Eicosanoid Synthesis[G], Monocyte, AMI, TC Apoptosis, Lymphocyte, CBL, T Helper & Neutrophil Pathways[B], Breast Cancer Estrogen Signaling Pathway[GE], B Cell Antigen Receptor[ST], Peptide GPCRS[G], N Glycan Degradation[G, K], GPCRDB Class B Secretin Like[G], Hematopoitic Cell Lineage[K], Cytokine Cytokine Receptor Interaction[K], T Cell, B Cell & Toll Like Receptor Signaling Pathway[K] FC Epsilon RI Signaling Pathway[K],JAK STAT Signaling Pathway[K], Natural Killer Cell Mediated Cytotoxicity[K], Arachidonic Acid Metabolism[K] Glycan Structures Degradation[K], Colorectal Cancer[K], Apoptosis[K] Leukocyte Transendothelial Migration[K], Regulation of Actin Cytoskeleton[K]
CC28[**Oligo, Astro**]	Regulation Cascade of Cyclin Expression[SA]
CC29[**Oligo**]	FAS Signaling Pathway[ST]

that the *pyrimidine pathway* appears in the results of both methods of clustering. This appearance corresponds with the pathway being a target in the disease and indeed the pathway is the target for the only drug oncologists have with some clinical activity against pancreatic cancer (it very modestly improves survival).[4] Again for pancreatic cancer the clustering data implies several metabolic pathways such as *glycolysis, gluconeogenesis, fatty acid synthase*. Since we have been so bereft of targets to go after in pancreatic cancer the present data gives us some confidence that targeting metabolic pathways in pancreatic cancer (with drugs such as phenformin) could be a very productive way to attack the disease.[27] From the Target Now clustering analyses, other possible leads for the clinic include methods to selectively go after tumor metabolism for salivary and gallbladder cancers as well.

From the clustering analysis of the REMBRANDT data, the clinical implications appear to be more limited. Concentrating on glioblastoma multiform (GBM), the worst type of brain cancer where advances are greatly needed, a possible target that appears worthy of pursuit is polo-like kinase -3. This is an important finding given the fact that *polo-like kinase inhibitors* are only now being brought into the clinic. Because of

the findings in the current study, we can now include patients with GBM in the phase I study with the new PLK inhibitor NMS-1286937H.

5.4. *Performance Comparison*

In order to evaluate the clustering results obtained and compare the algorithms, we used the performance metrics – coverage and performance, as described by Brandes et al.[2]

Coverage: The coverage of a clustering C is the fraction of intra-cluster edges ($m(C)$) within the complete set of edges (m), i.e

$$\text{coverage}(C) = \frac{m(C)}{m} = \frac{m(C)}{m(C) + \overline{m(C)}} \tag{4}$$

We choose this metric as it measures the wellness of a cut in a graph by taking the edges within the cluster(s) of a graph as a fraction of all the edges. Thus, the smaller a cut, the better the coverage it would have. Both a graph with no clusters at all and a graph with several disconnected components would have a coverage of 1 due to the absence of inter-cluster edges. Sparsity of the graph would not influence the coverage as long as the intra-connectivity is much higher than the inter-connectivity. Thus we anticipate that sub-graphs created by a minimum cut would have optimal coverage.

Performance: The performance of a clustering C counts the number of "correctly interpreted" pairs of nodes in a graph. More precisely, it is the fraction of intra-cluster edges together with non-adjacent pairs of nodes in different clusters, within the set of all pairs of nodes, i.e.

$$\text{performance}(C) = \frac{m(C) + \sum_{v,w \notin E, v \in C_i, w \in C_j, i \neq j} 1}{\frac{1}{2}n(n-1)} \tag{5}$$

We choose this measure as a means to assess the connectivity within the clusters of the graph. The fewer non-edges (pairs of nodes within the same cluster but lacking an edge between them) there are within a graph, the higher its performance would be. Further, a graph containing several singleton nodes, as well as a fully connected graph with a single giant cluster, would both have a performance of 1, as the number of non-edges would be zero in both cases. The goal is to maximize connectivity within a cluster for better performance and by maximizing intra-connectivity (approaching the number of possible edges of a graph), one can minimize the inter-connectivity. Performance will not do well in sparsely connected large graphs and clusters even though there may be substantially fewer edges between clusters.

Equations 4 and 5 are specific to undirected graphs. In the case of directed graphs, the maximum number of edges possible is twice as many as the edges possible in undirected graphs and the equations are correspondingly modified.

In our first study, we compare three spectral clustering variants – symmetric spectral clustering with two variants of asymmetric spectral clustering, using different balancing parameters (the average cut and the out-degree cut). The average of performance and coverage is used as a measure of the wellness of the clusters, and is plotted against the number of clusters produced, shown in Figure 2. Spectral clustering performed well both on undirected graphs and directed graphs. We note that the asymmetric algorithms peaked at a higher number of clusters than the symmetric algorithm. This implies that the normalized cut algorithm left intact large, well connected clusters until a certain threshold was reached. We also note that using the average cut exhibits less fluctuation in performance across different cluster sizes than using the out-degree of the nodes, explained by the fact that the average cut uses the number of nodes as the balancing parameter. However, if in fact a GRN follows a scale-free topology then the average cut may not prove to be the most useful in identifying biologically significant clusters because it does not take into account the interactions within a cluster.

In our second study, we compare spectral clustering (symmetric and asymmetric) with Markov clustering. As seen in Table 3, in terms of coverage, spectral clustering performed well over both directed and undirected graphs. In terms of performance, we find that the asymmetric case shows a lower performance value than the

Table 3: Performance Comparison of Markov and Spectral Clustering

Metric	*Dataset*	*Spectral Asym.*	*Spectral Sym.*	*MCL*	*Dataset*	*Spectral Asym.*	*Spectral Sym.*	*MCL*
Coverage	TN	0.9533	0.9693	0.9366	REM	0.7144	0.9680	0.9386
Performance	TN	0.6038	0.7930	0.7696	REM	0.8804	0.9271	0.8914

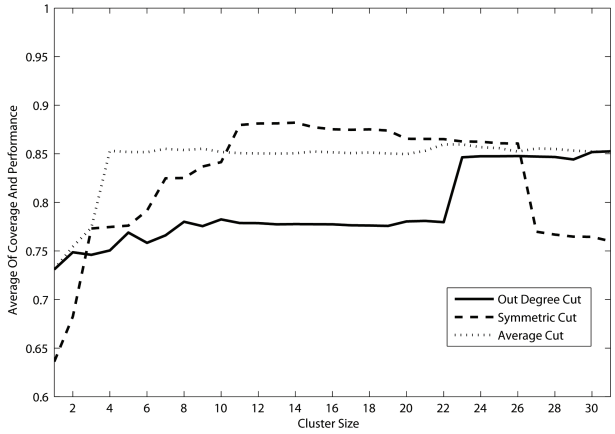


Fig. 2: Performance and Coverage Average of Spectral Clustering

other two. We also note that both MCL and symmetric spectral clustering performed well on the much larger REMBRANDT dataset, exhibiting good scalability. Comparing these results with the enrichment analyses (Tables 1 and 2, we conclude that well-balanced clusters need not necessarily correspond with biological meaningful clusters. Further we also observe that incorporating directionality did not correspond with a significant impact on the clustering, in terms of both biological significance and performance metrics.

6. Conclusion

The main contributions of our paper have been in a novel application of graph clustering, namely to identify clusters in context-specific GRNs. We have used Markov clustering and spectral clustering to identify context clusters in a two gene expression studies. The methods were compared to assess their ability to produce disjoint balanced clusters and scale to large graphs. Functional annotation of the genes and sample association studies show graph clustering to be promising in this area.

Future work includes studying the cluster enrichments obtained at increasing levels of cluster granularity, as well as the incorporation of prior biological knowledge into the clustering framework.

Acknowledgments

The authors wish to acknowledge Ina Sen, Michael Verdicchio and Sara Nasser for their help in the construction of the graphs and preparation of the manuscript. AR was supported by NIH R21 LM009706, SK was partially supported by NIH LM009706, CAA 0243-08 and P01 CA109552, and DVH by NIH P01 CA109552.

References

1. Y. Benjamini and Y. Hochberg. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 289–300, 1995.
2. U. Brandes, M. Gaertler, and D. Wagner. Experiments On Graph Clustering Algorithms. *Lecture Notes in Computer Science*, pages 568–579, 2003.
3. H. Brentani, O. Caballero, A. Camargo, A. da Silva, W. da Silva, E. Neto, M. Grivet, A. Gruber, P. Guimaraes, W. Hide, et al. The Generation and Utilization of a Cancer-Oriented Representation of the Human Transcriptome by Using Expressed Sequence Tags. *Proceedings of the National Academy of Sciences*, 100(23):13418–13423, 2003.

4. H. Burris 3rd, M. Moore, J. Andersen, M. Green, M. Rothenberg, M. Modiano, M. Cripps, R. Portenoy, A. Storniolo, P. Tarassoff, et al. Improvements in Survival and Clinical Benefit with Gemcitabine As First-Line Therapy for Patients with Advanced Pancreas Cancer: A Randomized Trial. *Journal of Clinical Oncology*, 15(6):2403, 1997.
5. R. Dubes and A. Jain. Algorithms for Clustering Data. *Prentice Hall*, 355:356, 1988.
6. V. Fauquette, S. Aubert, S. Groux-Degroote, B. Hemon, N. Porchet, I. Van Seuningen, and P. Pigny. Transcription Factor AP-2 {Alpha} Represses Both the Mucin MUC4 Expression and Pancreatic Cancer Cell Proliferation. *Carcinogenesis*, 28(11):2305, 2007.
7. T. Freeman, L. Goldovsky, M. Brosch, S. van Dongen, P. Mazière, R. Grocock, S. Freilich, J. Thornton, and A. Enright. Construction, Visualisation, and Clustering of Transcription Networks From Microarray Expression Data. *PLoS Comput Biol*, 3(10):2032–2042, 2007.
8. W. Hahn and R. Weinberg. Modelling the Molecular Circuitry of Cancer. *Nat. Rev. Cancer*, 2(5):331–341, 2002.
9. K. Hendricks, F. Shanahan, and E. Lees. Role for BRG1 in Cell Cycle Control and Tumor Suppression. *Molecular and Cellular Biology*, 24(1):362–376, 2004.
10. D. Higham, G. Kalna, and M. Kibble. Spectral Clustering and Its Use in Bioinformatics. *Journal of computational and applied mathematics*, 204(1):25–37, 2007.
11. A. Jain, M. Murty, and P. Flynn. Data Clustering: A Review. *ACM Computing Surveys*, 31(3), 1999.
12. S. Kim, I. Sen, and M. Bittner. Mining Molecular Contexts of Cancer Via in-Silico Conditioning. In *Computational Systems Bioinformatics: Proceedings of the CSB 2007 Conference*. Imperial College Press, 2007.
13. Y. Kluger, R. Basri, J. Chang, and M. Gerstein. Spectral Biclustering of Microarray Data: Coclustering Genes and Conditions, 2003.
14. S. Madhavan, J. Zenklusen, Y. Kotliarov, H. Sahni, H. Fine, and K. Buetow. Rembrandt: Helping Personalized Medicine Become a Reality Through Integrative Translational Research. *Mole. Cancer Res.*, 7(2):157, 2009.
15. M. Meila and W. Pentney. Clustering by Weighted Cuts in Directed Graphs. In *Proceedings of the 2007 SIAM International Conference on Data Mining*, 2007.
16. S. Oldford, J. Robb, D. Codner, V. Gadag, P. Watson, and S. Drover. Tumor Cell Expression of HLA-DM Associates with a Th1 Profile and Predicts Improved Survival in Breast Carcinoma Patients. *International immunology*, 18(11):1591, 2006.
17. B. Samuel Lattimore, S. van Dongen, and M. Crabbe. GeneMCL in Microarray Analysis. *Computational Biology and Chemistry*, 29(5):354–359, 2005.
18. S. Schaeffer. Graph Clustering. *Computer Science Review*, 1(1):27–64, 2007.
19. I. Sen, M. Verdicchio, S. Jung, R. Trevino, M. Bittner, and S. Kim. Context-Specific Gene Regulations in Cancer. In *Proceedings of the Pacific Symposium on Biocomputing*, 2009.
20. J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 888–905, 2000.
21. J. Spratlin, N. Serkova, and S. Eckhardt. Clinical Applications of Metabolomics in Oncology: A Review. *Clinical Cancer Research*, 15(2):431, 2009.
22. A. Subramanian, P. Tamayo, V. Mootha, S. Mukherjee, B. Ebert, M. Gillette, A. Paulovich, S. Pomeroy, T. Golub, E. Lander, et al. Gene Set Enrichment Analysis: A Knowledge-Based Approach for Interpreting Genome-Wide Expression Profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.
23. D. Tritchler, S. Fallah, and J. Beyene. A Spectral Clustering Method for Microarray Data. *Computational Statistics and Data Analysis*, 49(1):63–76, 2005.
24. S. van Dongen. Graph Clustering by Flow Simulation. *University of Utrecht*, 2000.
25. S. van Dongen. Technical Report INS-R0010: A Cluster Algorithm for Graphs. *National Research Institute for Mathematics and Computer Science*, 2000.
26. D. Von Hoff, R. Penny, S. Shack, E. Campbell, D. Taverna, M. Borad, D. Love, J. Trent, and M. Bittner. Frequency of Potential Therapeutic Targets Identified by Immunohistochemistry (IHC) and DNA Microarray (DMA) in Tumors From Patients Who Have Progressed On Multiple Therapeutic Agents. *Journal of Clinical Oncology*, 24(18_suppl):3071, 2006.
27. D. Wise, R. DeBerardinis, A. Mancuso, N. Sayed, X. Zhang, H. Pfeiffer, I. Nissim, E. Daikhin, M. Yudkoff, S. McMahon, et al. Myc Regulates a Transcriptional Program That Stimulates Mitochondrial Glutaminolysis and Leads to Glutamine Addiction. *Proceedings of the National Academy of Sciences*, 105(48):18782, 2008.

Writing and Compiling Code into Biochemistry*

Adam Shea, Brian Fett, Marc D. Riedel and Keshab Parhi
Department of Electrical and Computer Engineering
University of Minnesota
Minneapolis, Minnesota 55455
Email: {shea0097, fett, mriedel, parhi}@umn.edu

This paper presents a methodology for translating iterative arithmetic computation, specified as high-level programming constructs, into biochemical reactions. From an input/output specification, we generate biochemical reactions that produce output quantities of proteins as a function of input quantities, performing operations such as addition, subtraction, and scalar multiplication. Iterative operations – "while loops" and "for" loops – are implemented by transferring quantities between protein types, based on a clocking mechanism. Synthesis first is performed at a conceptual level, in terms of abstract biochemical reactions – a task analogous to *high-level program* compilation. Then the results are mapped onto specific biochemical reactions, selected from libraries – a task analogous to *machine language* compilation. A possible experimental chassis is the mechanism of DNA strand displacement developed by Erik Winfree's group at Caltech. We demonstrate our approach through the compilation of a variety of standard iterative functions: multiplication, discrete logarithms and linear transforms on time series. The designs are validated through transient stochastic simulation of the chemical kinetics.

1. Introduction

Recent accomplishments in synthetic biology portend of a coming revolution. From *Salmonella* that secretes spider silk proteins, to yeast that degrades biomass into ethanol,[1] to *E. coli* that produces antimalarial drugs,[2] the potential impacts are far-reaching.

The scope of the field is, in fact, broader. The J. Craig Venter Institute's team has made significant progress toward the goal of artificial life: a living bacterial cell with fully synthetic DNA.[3,4] In engineering terms, the objective is to assemble a machine (a synthetic bacterium) in which the functionality of all the parts (the genes, the proteins that they code for, and how these interact biochemically) are understood. If the machine works, this vindicates the scientific understanding; if it doesn't – and surely it won't at first – then new understanding can be achieved by examining where and how it breaks. Of course, with a working blueprint for a synthetic machine, new functionality can be engineered robustly and effectively.

The set of constitutive parts that can be used for genetic manipulation in synthetic systems is vast. Comprehensive repositories of genetic data have been assembled – some public, some commercial – cataloging genes, their DNA sequences, and their products. A concerted effort has been made to assemble repositories of standardized and interoperable parts for synthetic applications. The platforms used will depend on the application, but the technology for synthesizing DNA is becoming routine: firms have started offering custom-gene synthesis through e-commerce websites (the going rate is $0.49 per base pair). So, in a real sense, the hardware for synthetic biology exists, i.e., the technology and infrastructure for obtaining cells with custom-designed genes. The instruction set is, to a large extent, known, i.e., genes and their function, cataloged in libraries. The challenge is: *how can we write code with these instructions on this type of hardware*?

Conceptually, the rules of biochemistry are straight-forward: each biochemical reaction is a primitive process that specifies how and at what rate different types of proteins combine to form other types of proteins. The complexity stems from the dynamics at play among the multitude of coupled reactions operating on the different protein types, asynchronously and in parallel. Techniques for *analyzing* such processes are well established.[5] However, *synthesizing computation* with such mechanisms requires entirely new techniques – and an entirely new mindset.

One of the great successes of computer engineering has been in abstracting and scaling the design problem. The physical behavior of transistors is understood in terms of differential equations – say, with models found in tools such as SPICE.[6] However, the design of circuits proceeds at a more abstract level – in terms of switches, gates, and functional units. Software is conceived of and validated independently of the hardware platform. This modular approach makes the design tractable; furthermore, it permits a systematic

exploration of different configurations, leading to optimal designs. Although driven by experimental expertise, synthetic biology has reached a stage where it calls for a similar degree of abstraction.

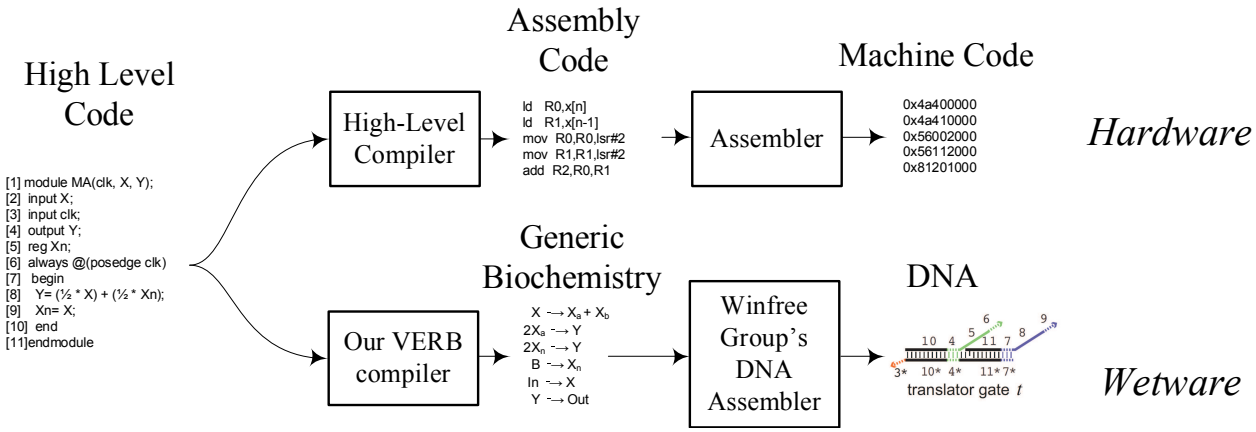


Fig. 1: Analogy between compiling code for hardware and for biochemistry.

2. Overview

Our approach brings a design perspective: we tackle the problem of synthesizing biochemical reactions that implement specified input/output functionality. From an input/output specification in a high-level language like C, we generate biochemical reactions that produce output quantities of proteins as a function of input quantities, performing operations such as addition, subtraction, and scalar multiplication. Iterative operations – "while loops" and "for loops" – are implemented by transferring quantities between protein types, based on a clocking mechanism.

2.1. *Computation with Biochemistry*

Interesting biochemistry typically involves complex molecules such as proteins and enzymes. Within the confines of a cell, the *quantities* of such molecules are often surprisingly small: on the order of tens, hundreds, or thousands of molecules of each type. At this scale, individual reactions matter and the problem must be modeled discretely.[5]

In our view of biochemical computation, the quantities of proteins are whole numbers (i.e., non-negative integers). We will refer to these quantities as "**registers**". Biochemical reactions alter these quantities: the reactions fire repeatedly, modifying the protein quantities by small integer amounts. Consider the reaction

$$a + b \xrightarrow{\text{fast}} 2c. \tag{1}$$

When this reaction fires, one molecule of a is consumed, one of b is consumed, and two of c are produced. (Accordingly, a and b are called the *reactants* and c the *product*.) Each reaction has an associated *rate* (listed above the arrow in our notation). Given several reactions, the probability of each firing is proportional both to its rate and to the quantities of its reactants present. Although we refer to rates in relative and qualitative terms – e.g., "fast" vs. "slow" – these are, in fact, quantitative values that are either deduced from biochemical principles or measured experimentally. The functionality of a biochemical system can be analyzed using stochastic simulation.[5,7,8]

Our contribution is to tackle the problem of computation at this abstract level – working not with specific molecular types but rather with arbitrary types (a, b, c, etc.). This is illustrated in Figure 1. For conventional hardware, programs are specified in a high-level language like C; a compiler translates this into assembly language; then an assembler produces the machine code. In our bio-design flow, we begin with the same sort of high-level description (we use Verilog, a hardware description language[9]). Our prototype compiler

called VERB (Verilog Elements for Register-Based Biochemistry) compiles these specification into generic biochemical reactions. Then this design is mapped on a chemical substrate. The end result is a description of the actual biochemistry: protein-protein reactions or DNA interactions.

2.2. *Compiling the Programs into Biochemistry*

A possible experimental chassis for our method is the mechanism of DNA-based computation advocated by Erik Winfree's group at Caltech.[10] They have shown that the kinetics of arbitrary chemical reaction networks can be implemented through DNA strand-displacement reactions. They provide an assembler that accepts a set of arbitrary biochemical reactions with nearly any rate structure and delivers the corresponding DNA sequences for the displacement reactions. Reaction rates are controlled by designing sequences with different binding strengths; the binding strengths are controlled by the length and sequence composition of toeholds.[10] Accordingly, our contribution can be positioned as the "front end" of the compiilation flow; the DNA assembler and experimental chassis described by these authors constitute the "back-end".

3. Related Work and Context

There has been considerable research directed at the question of computation with genetic regulatory mechanisms.[11] DNA and RNA-based computation have been explored theoretically and demonstrated experimentally.[12–14] Mathematical expertise from control and dynamical systems has been applied to the analysis of biochemical systems.[15] Oscillatory mechanisms, suitable for the sort of clocking used in our designs, have been demonstrated experimentally.[16] Samoilov, Arkin and Ross established a comprehensive analytic framework for studying the dynamics of biological systems in terms of the signal processing functions that they perform.[17] Soloveichik, Cook, Winfree and Bruck discuss theoretical aspects of molecular computation.[18] The concepts of register-based computation and clocking that we use are due to these authors.[18]

4. A Toolkit for Biochemical Arithmetic

We describe elements of flexible toolkit of functional modules. In our view of biochemical computation, the input quantities of proteins are non-negative integers. Computation is implemented by biochemical reactions. These fire repeatedly, modifying the protein quantities by small integer amounts; the end results are output quantities of proteins. The challenge in setting up such computing, of course, is that the biochemical reactions execute asynchronously and in parallel.[19]

The computation that we propose in the modules below is exact and independent of the specific values of the rates, although it requires that the rates in different categories differ by a sufficient amount. For instance, we assume that when a "fast" reaction can fire it does so – repeatedly, until it runs out of reactants – before a "slow" reaction ever fires.

Scalar Multiplication and Addition

$$\boxed{|z| = \tfrac{a}{b}|x| + \tfrac{c}{d}|y|}$$

Reactions:

$$bx \longrightarrow az$$
$$dy \longrightarrow cz$$

Fig. 2: **A biochemical module for addition and scalar multiplication.** When the first reaction fires, it consumes b molecules of x and produces a molecules of z. When the second reaction fires, it consumes d molecules of y and produces c molecules of z. When both reactions have fired to completion, the number of molecules of z will be the scaled sum of the number molecules x plus the number of molecules of y.

4.1. *Addition and Scalar Multiplication*

The first and simplest of these computations is addition with scalar multiplication. We can implement this by merely choosing reactions with the correct stoichiometry. This is shown in Figure 2. (Here and throughout we use $|\cdot|$ to denote the quantity of a type of molecule.)

4.2. *Multiplication*

We illustrate how to implement multiplication. This is shown in Figure 3. In this set of reactions, note that none can fire until the first one does, producing a molecule of type i. When it does, it initiates an iteration of a *loop*: the quantity of z increases as the second reaction fires repeatedly until there is no more y remaining. Once this process terminates, the third and fourth reactions fire, ending the iteration and restoring y to its initial value. In each iteration, the quantity of x is decremented by one and the quantity of z is incremented by y. The final result is a quantity of z equal to the initial quantity of x times the quantity of y.

4.3. *More Complex Arithmetic*

Modules for computing exponentiation, discrete logarithms and raising to a power, are shown in Figures 4, 5, and 6, respectively.[19] With the latter, our scheme can be used to implement arbitrary polynomial functions; hence, in principle, it could be used to approximate complex functions through Taylor series expansions.

Multiplication

$$\boxed{|z| = |x| \times |y|}$$

Pseudo-code:

```
while x > 0 {
   z = z + y
   x = x - 1
}
```

Reactions:

$$\begin{aligned} x &\xrightarrow{\text{slowest}} i \\ i + y &\xrightarrow{\text{fastest}} i + y\prime + z \\ i &\xrightarrow{\text{fast}} \varnothing \\ y\prime &\xrightarrow{\text{slow}} y \end{aligned}$$

Fig. 3: **A biochemical module for multiplication.** The module consumes molecules of x one at a time, adding the quantity of y to the quantity of z each time. (Here i and $y\prime$ are intermediate types; it is assumed that no molecules of these types are present initially. The symbol $\varnothing$ as a product indicates "nothing," meaning that the type degrades into products that are no longer tracked or used.)

Exponentiation

$$\boxed{|y| = 2^{|x|}}$$

Pseudo-code:

```
y = 1
while x > 0 {
  y = 2 * y
  x = x - 1
}
```

Reactions:

$$\begin{aligned} c + 2y &\xrightarrow{\text{fast}} c + y \\ c &\xrightarrow{\text{medium}} \varnothing \\ x &\xrightarrow{\text{slow}} i \\ i + y &\xrightarrow{\text{faster}} i + 2y\prime \\ i &\xrightarrow{\text{fast}} \varnothing \\ y\prime &\xrightarrow{\text{medium}} y \end{aligned}$$

Fig. 4: **A biochemical module for exponentiation.** First, a pair of reactions set the quantity of y to 1. Then molecules of x are consumed one at a time, doubling the quantity of y each time. (Here c and $y\prime$ are additional types; it is assumed that initially there is some non-zero quantity of c, and zero quantity of $y\prime$.)

4.4. *Clocking*

An important constraint in our design methodology is the timing captured in the relative rates of the biochemical reactions. With modules such as multiplication described above, there is an implicit ordering of the reactions. To achieve this, the reaction rates must sometimes be separated by orders of magnitude: some much faster than others, some much slower. This may be unrealistic.

Logarithm

$$\boxed{|y| = log_2(|x|)}$$

Pseudo-code:

```
while x > 1 {
  x = x/2
  y = y + 1
}
```

Reactions:

$$\begin{aligned} b &\xrightarrow{\text{slow}} a + b \\ a + 2x &\xrightarrow{\text{faster}} c + x' + a \\ 2c &\xrightarrow{\text{faster}} c \\ a &\xrightarrow{\text{fast}} \varnothing \\ x' &\xrightarrow{\text{medium}} x \\ c &\xrightarrow{\text{medium}} y \end{aligned}$$

Fig. 5: **A biochemical module for computing a base-2 logarithm.** The input x repeatedly halves itself; each time it does so, y is incremented by one. (Here a, b, c and x' are are additional types; it is assumed that initially there is some non-zero quantity of b, and zero quantity of a, c and x'.)

Raising to a Power

$$\boxed{|y| = |x|^{|p|}}$$

Pseudo-code:

```
y = 1
d = 0
while p > 0 {
  w = x
  while w > 0 {
    d = d + y
    w = w - 1
  }
  y = d
  d = 0
  p = p - 1
}
```

Reactions:

$$\begin{aligned} c + 2y &\xrightarrow{\text{fast}} c + y \\ c &\xrightarrow{\text{medium}} \varnothing \\ p &\xrightarrow{\text{slowest}} a \\ a + x &\xrightarrow{\text{medium}} b + a + x' \\ b + y &\xrightarrow{\text{fastest}} y' + d + b \\ b &\xrightarrow{\text{faster}} \varnothing \\ y' &\xrightarrow{\text{fast}} y \\ a &\xrightarrow{\text{slow}} e \\ e + y &\xrightarrow{\text{faster}} e \\ e + x' &\xrightarrow{\text{faster}} e + x \\ e &\xrightarrow{\text{fast}} \varnothing \\ d &\xrightarrow{\text{slower}} y. \end{aligned}$$

Fig. 6: **A biochemical module for computing a raising to a power.** First, a pair of reactions set the quantity of y to 1. Then a pair of nested loops achieves the computation: the inner loop computes $|x|$ times the current result (starting with 1); the outer loop does this operation $|p|$ times. (Here a, b, c, d, e, x', and y' are are additional types. It is assumed that that quantities of all of these except for c are initially zero; the quantity of c must be any non-zero value.)

To overcome this issue, we have have proposed a technique that we call "module-locking".[20] The scheme involves adding a *key* requirement to each phase of the computation. *Keysmiths* are produced occasionally;

if other keys are present, they quickly disappear – before they can produce their key. Only if no other keys are present will they produce their key. This ensures that at most one type of key is present (thus allowing only one part of the loop to fire at a time); also it ensures that only one key of that type is present (thus allowing for re-locking). The template for reactions with this functionality is:

$$\begin{aligned} \forall i: \quad & \varnothing \xrightarrow{\text{slow}} \text{keysmith}_i \\ \forall i: \quad & \text{keysmith}_i \xrightarrow{\text{slow}} \text{key}_i \\ \forall i: \quad & \text{key}_i \xrightarrow{\text{slow}} \varnothing \\ \forall i,j: \quad & \text{key}_i + \text{keysmith}_j \xrightarrow{\text{fast}} \text{key}_i \end{aligned} \tag{2}$$

This is for all i, j phases of the computation, e.g., steps in an operation like multiplication. (The symbol $\varnothing$ as a reactant indicates that the reaction does not alter the quantity of the reactant types, perhaps because the quantity of these is large or replenishable; in such cases we can assume that the quantity is simply unity and adjust the rate accordingly). The first reaction of each phase must be modified so that it depends on the key:

$$\text{key}_i + \text{reactants} \rightarrow \text{products} + \text{key}_i. \tag{3}$$

Typically, the key will be a catalyst, appearing as both a reactant and a product, but this need not be the case. With locking, our method synthesizes robust computation that is nearly rate independent, requiring at most two speeds ("fast" and "slow"). The trade-off is with respect to the size of the solution: more reactions are needed. Further details are in.[20]

5. Compiling Iterative Code

By combining arithmetic operations and our clocking mechanism, we can implement iterative operations such as linear transforms on time series. Such operations are useful for performing filtering operations.

Example 1
Consider an application that calls for biochemistry that performs a filtering operation such as computing a moving average. *Given a noisy input signal $X[n]$, a moving average filter produces an output signal $Y[n]$ that is a smoother version of the input. The function is*

$$Y[n] = \frac{1}{2}X[n] + \frac{1}{2}X[n-1].$$

where the n-th value is the current value and the $(n-1)$-st value is the previous value of each signal. The iterative operation can be specified as follows (we use the syntax of Verilog[9]):

```
[1]  module MA(clk, X, Y);
[2]    input X;
[3]    input clk;
[4]    output Y;
[5]    reg Xn;
[6]    always
[7]      begin
[8]         Y= (1/2 * X) + (1/2 * Xn);
[9]         Xn= X;
[10]     end
[11]  endmodule
```

We translate this specification into a set of biochemical reactions, as follows. Each operation is translated into a biochemical reaction with the protein types that correspond to the variables. All these reactions are keyed, according to clock phases. These reactions are show in Figure 7. For simplicity, here we omit the

details of how the keys are generated. Key-keysmith reactions of the form of Equation 2 should be included for $\text{key}_0, \text{key}_1$, and key_2.

$$
\begin{array}{rr}
\text{R1:} & \text{key}_0 + X \xrightarrow{\text{fast}} X_a + X_b + \text{key}_0 \\
\text{R2:} & \text{key}_0 + Y \xrightarrow{\text{fast}} \text{key}_0 \\
\hline
\text{R3:} & \text{key}_1 + 2X_n \xrightarrow{\text{fast}} Y + \text{key}_1 \\
\hline
\text{R4:} & \text{key}_2 + 2X_b \xrightarrow{\text{fast}} Y + \text{key}_2 \\
\text{R5:} & \text{key}_2 + X_a \xrightarrow{\text{fast}} X_n + \text{key}_2 \\
\hline
\text{CR1:} & \text{key}_2 \xrightarrow{\text{slow}} \text{keysmith}_0 \\
\text{CR2:} & \text{keysmith}_0 \xrightarrow{\text{slow}} \text{key}_0 \\
\hline
\text{CR3:} & \text{key}_0 \xrightarrow{\text{slow}} \text{keysmith}_1 \\
\text{CR4:} & \text{keysmith}_1 \xrightarrow{\text{slow}} \text{key}_1 \\
\hline
\text{CR5:} & \text{key}_1 \xrightarrow{\text{slow}} \text{keysmith}_2 \\
\text{CR6:} & \text{keysmith}_2 \xrightarrow{\text{slow}} \text{key}_2
\end{array}
$$

Fig. 7: Biochemistry implementing the moving-average filter.

We simulate and validate our designs with transient stochastic simulation.[8] The simulation results shown in Figure 8, illustrate the functionality of the design: the moving average smooths high-frequency noise. Here, the input X is shown in green; it is a noisy sinusoid. The output Y is shown in red; note that it is a clearer sinusoidal waveform.

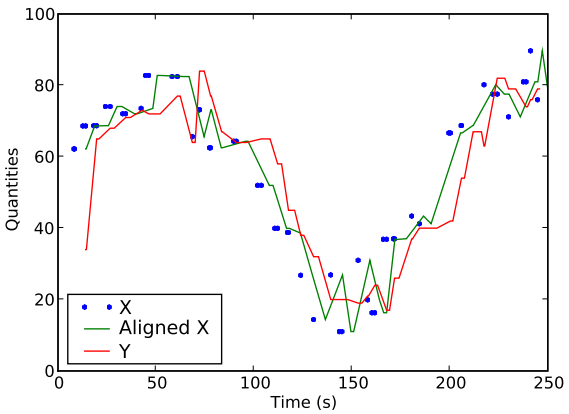


Fig. 8: Input and output waveforms for a biochemical moving-average filter. The input quantity X is shown both as points from the simulation and as the ideal curve. The output Y is shown in red. The green curve shows where the value of X is after the average delay of the system in order to be time-aligned with the red output curve.

6. Additional Design Examples

We briefly discuss a few other designs that we synthesized with our compiler. All perform iterative arithmetic on non-negative integer values; these integer values correspond to time-varying quantities of input and output proteins.

- The **deserializer** is simply an 8-element shift register. Instead of operating on bits, it operates on

Table 1: Compilation Results

	deserializer	vector matrix multiplier	integrator	differentiator
Reactions	27	28	14	19
Registers	23	18	4	10
Clocks	2	5	3	4

integer values. This module is useful as a starting point for convolution-based algorithms and in signal decoding.

- The **vector-matrix** multiplier multiplies a 3-element input vector by a fixed 3x3 matrix to obtain an output vector. It can be trivially expanded to higher dimensions; the only limitation is that it has nonnegative integers for input and output.
- The **integrator** implements a "bipolar" encoding: its inputs and outputs are represented as as the difference of two protein quantities. While each of these quantities is a nonnegative integer, their difference can be any integer, positive or negative. It computes a running sum of its input values. In order to ensure that the output quantities do not increase without bound, an "equalize" operation is implemented. This reduces both the positive and negative output quantities by the same amount until one is zero.
- The **differentiator** operates with this same bipolar encoding. It computes the running difference of the last two values.

The parameters of the biochemical designs that our compiler produces are shown in Table 1: the number of reactions, the number of registers and the number of clock phases, for each.

7. Discussion

This paper is forward-looking and positioned in the realm of synthetic biology; it presents concepts for designing new functionality with realistic yet abstract mechanisms of biochemistry. Evidently, we only implement of small subset of the functionality of a complete high-level language, such as C or Verilog. It is beyond the scope to implement a full specification such a language – to attempt to do so would be besides the point; syntactical constructs are are not germane. Rather the goal of this research is to demonstrate the feasibility of *computation* with biochemistry.

Our clocked and locked design methodology for biochemical computation is the most robust and general approach that has been suggested. With rapid progress in custom gene synthesis, it has potential applications on the horizon in domains such as biochemical sensing, drug production, and disease treatment.

Thus far, our work has focused on computational methods and tools for the high-level compilation phase of bio-design automation, generating designs in terms of generic biochemical reactions. We plan to implement and validate the technology-mapping phase using the universal DNA substrate mechanisms being developed by Erik Winfree's group at Caltech as the experimental chassis.

References

1. M. Sedlak and N. Ho, "Production of ethanol from cellulosic biomass hydrolysates using genetically engineered Saccharomyces yeast capable of cofermenting glucose and xylose," *Applied biochemistry and biotechnology*, vol. 114, no. 1, pp. 403–416, 2004.
2. D. Ro, E. Paradise, M. Ouellet, K. Fisher, K. Newman, J. Ndungu, K. Ho, R. Eachus, T. Ham, J. Kirby *et al.*, "Production of the antimalarial drug precursor artemisinic acid in engineered yeast," *Nature*, vol. 440, pp. 940–943, 2006.
3. D. Gibson, G. Benders, C. Andrews-Pfannkoch, E. Denisova, H. Baden-Tillson, J. Zaveri, T. Stockwell, A. Brownley, D. Thomas, M. Algire *et al.*, "Complete chemical synthesis, assembly, and cloning of a Mycoplasma genitalium genome," *Science's STKE*, vol. 319, no. 5867, p. 1215, 2008.

4. J. Glass, N. Assad-Garcia, N. Alperovich, S. Yooseph, M. Lewis, M. Maruf, C. Hutchison III, H. Smith, and J. Venter, "Essential genes of a minimal bacterium," *Proceedings of the National Academy of Sciences*, vol. 103, no. 2, pp. 425–430, 2006.
5. D. Gillespie, "Exact Stochastic Simulation of Coupled Chemical Reactions," *The Journal of Physical Chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977.
6. L. Nagel and D. Pederson, "Simulation program with integrated circuit emphasis," in *Midwest Symposium on Circuit Theory*, 1973.
7. M. Gibson and J. Bruck, "Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels," *Journal OF Physical Chemistry A*, vol. 104, no. 9, pp. 1876–1889, 2000.
8. B. Chen and M. Riedel, "Stochastic transient analysis of biochemical reactions," *Pacific Biocomputing Symposium*, 2009.
9. "IEEE standard Verilog hardware description language," *IEEE Std 1364-2001*, 2001.
10. D. Soloveichik, G. Seelig, and E. Winfree, "DNA as a Universal Substrate for Chemical Kinetics (Extended Abstract)," in *DNA 14: The 14th International Meeting on DNA Computing*, Prague, Czech Republic, Jun. 2008.
11. R. Weiss, S. Basu, S. Hooshangi, A. Kalmbach, D. Karig, R. Mehreja and I. Netravali, "Genetic circuit building blocks for cellular computation, communications, and signal processing," *Natural Computing*, pp. 47–84, 2003.
12. L. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, no. 11, pp. 1021–1024, 1994.
13. Y. Benenson, B. Gil, U. Ben-Dor, R. Adar, and E. Shapiro, "An Autonomous Molecular Computer for Logical Control of Gene Expression," *Nature*, vol. 429, no. 6990, pp. 423–429, 2004.
14. M. Win and C. Smolke, "From the Cover: A modular and extensible RNA-based gene-regulatory platform for engineering cellular function," *Proceedings of the National Academy of Sciences*, vol. 104, no. 36, p. 14283, 2007.
15. H. El-Samad, H. Kurata, J. Doyle, C. Gross, and M. Khammash, "Surviving heat shock: Control strategies for robustness and performance," *Proceedings of the National Academy of Sciences*, pp. 2736–2741, 2005.
16. M. Elowitz and S. Leibler, "A Synthetic Oscillatory Network of Transcriptional Regulators," *Nature*, vol. 403, no. 6767, pp. 335–338, 2000.
17. M. Samoilov, A. Arkin, and J. Ross, "Signal Processing by Simple Chemical Systems," *Journal of Physical Chemistry A*, vol. 106, no. 43, pp. 10 205–10 221, 2002.
18. D. Soloveichik, M. Cook, E. Winfree, and J. Bruck, "Computation with Finite Stochastic Chemical Reaction Networks," *Natural Computing*, To Appear.
19. B. Fett, J. Bruck, and M. Riedel, "Synthesizing Stochasticity in Biochemical Systems," *Design Automation Conference*, pp. 640–645, 2007.
20. B. Fett and M. Riedel, "Module Locking in Biochemical Synthesis," *International Conference on Computer-Aided Design*, 2008.
21. K. Parhi, "VLSI Digital Signal Processing Systems," *Wiley*, 1999.

SYNTHESIS OF PHARMACOKINETIC PATHWAYS THROUGH KNOWLEDGE ACQUISITION AND AUTOMATED REASONING

LUIS TARI, SAADAT ANWAR, SHANSHAN LIANG, JÖRG HAKENBERG, CHITTA BARAL

Department of Computer Science and Engineering,
Arizona State University, Tempe, AZ 85287, USA

Biological pathways are seen as highly critical in our understanding of the mechanism of biological functions. To collect information about pathways, manual curation has been the most popular method. However, pathway annotation is regarded as heavily time-consuming, as it requires expert curators to identify and collect information from different sources. Even with the pieces of biological facts and interactions collected from various sources, curators have to apply their biological knowledge to arrange the acquired interactions in such a way that together they perform a common biological function as a pathway. In this paper, we propose a novel approach for automated pathway synthesis that acquires facts from hand-curated knowledge bases. To comprehend the incompleteness of the knowledge bases, our approach also obtains facts through automated extraction from Medline abstracts. An essential component of our approach is to apply logical reasoning to the acquired facts based on the biological knowledge about pathways. By representing such biological knowledge, the reasoning component is capable of assigning ordering to the acquired facts and interactions that is necessary for pathway synthesis. We demonstrate the feasibility of our approach with the development of a system that synthesizes pharmacokinetic pathways. We evaluate our approach by reconstructing the existing pharmacokinetic pathways available in PharmGKB. Our results show that not only that our approach is capable of synthesizing these pathways but also uncovering information that is not available in the manually annotated pathways.

1. Introduction

Developing systems and algorithms to assist and guide biological researchers in reverse engineering and synthesis of biomolecular systems has been a long term goal for the fields of computational biology and bioinformatics. An important task of modeling biomolecular systems is the building of pathways for various biological processes. In building pathways for processes such as pharmacokinetics, the ability to collect and integrate information contained in existing databases and the literature is important so that partial pathways can be built. With the partial pathways, a researcher can identify what gap in knowledge needs to be filled.

Several knowledge bases have been created for the need of pathway information, focusing on different aspects of networks and pathways. Reactome [1], KEGG [2] and HumanCyc [3] are examples of knowledge bases for metabolic pathways, while Biocarta[1] and Panther [4] consist of knowledge bases for signaling pathways. PharmGKB [5] is a knowledge base for drug-oriented genomic information that includes pharmacokinetic and pharmacodynamic pathways. These knowledge bases rely on manual curation by experts and the pathway information is very precise but far from being complete due to the intensive process required in the annotation of pathways. It is therefore necessary to investigate another paradigm for the curation of pathways to speed up the annotation process.

In this paper, we propose a novel approach for the automated synthesis of pharmacokinetic pathways by first acquiring the necessary pharmacokinetic facts and interactions of the target drug from existing curated knowledge bases. As the curation effort of these knowledge bases is yet to be completed, our approach includes an automated text extraction component that extracts facts and interactions from Medline abstracts. The inclusion of knowledge extracted from Medline abstracts can lead to the synthesis of more comprehensive pathways, as compared to using only the curated knowledge bases for building pathways. In the synthesis of pathways, it is essential to indicate the ordering of the interactions in a pathway, as a pathway is a series of interactions that are triggered by one another, in which the consequences of the interactions include the activation of certain biological functions or generation of products such as metabolites as a result of drug metabolism. To assign the ordering of the interactions, our approach includes the logical representation of the general properties and behavior of pharmacokinetic pathways. Automated reasoning can then be applied to the acquired knowledge so that ordering of the interactions can be assigned to synthesize pathways. The inclusion of such reasoning capabilities on top of mining relevant knowledge distinguishes

[1] Biocarta – http://www.biocarta.com

our approach from typical data-driven and knowledge-driven approaches in pathway curation. Data can be collected by means of large-scale protein-protein interaction networks from experimental sources such as two-hybrid screening (Y2H) [6]. On the other hand, knowledge can be extracted by automated text extraction techniques to produce large-scale interaction networks and pathways [7-10]. While these networks of interactions provide insights to biological discovery, it is not always straightforward to identify which interactions are indeed biologically related within a large network of interactions. To determine which parts of the interaction networks indeed correspond to pathways, pathway curators rely on visualization and pathway building tools to synthesize pathways [11]. Pathway building tools such as CellDesigner [12] or proprietary tools such as Ingenuity IPA[2] and ActiveMotif[3] based on manually curated databases allow curators to visualize and assemble pathways from an interaction network. Such methodology still heavily depends on the biological knowledge of the expert curators for pathways synthesis. An alternative is the use of graph-theoretic methods on interaction networks (see [13] for a survey) or network alignment methods over interaction networks of multiple species to uncover conserved modules as pathways [14, 15].

It becomes apparent that fully-automated systems for pathway synthesis are required to have the capabilities of acquiring various kinds of information from multiple sources, as well as assigning appropriate order to the acquired interactions. To automatically arrange the interactions for pathway synthesis, our work includes a reasoning component for this purpose. With proper representation of pathways for a particular biological process such as pharmacokinetics, reasoning can be applied to the acquired knowledge and automatically assigned the appropriate ordering of the interactions to synthesize pharmacokinetic pathways. The approach of utilizing biological domain knowledge has been applied to various applications, such as the generation of metabolic networks based on stoichiometric constraints [16] and hypothesis generation in signaling pathways [17]. The need of formulating biological domain knowledge and applying reasoning to infer pathways from text is highlighted as new challenges in [18]. In pharmacokinetic pathways, an example of a biological property is that drug metabolites are generated as a result of drug metabolism. In other words, a metabolized drug is a precondition for the generation of drug metabolites to occur, and the effect of the interaction is the production of drug metabolites. By encoding the logic representation in the form of pre- and post-conditions of pharmacokinetic properties that describe the course of drug disposition in the body, which includes drug absorption, distribution, metabolism and excretion, reasoning can be applied to the interactions in order to find a sequence that satisfies the pharmacokinetic properties. Finding a sequence of actions is known as *planning* in the field of artificial intelligence, and planning is considered as one kind of *reasoning*. More specifically, planning can be described as given the initial states and the goals of the problem, find a sequence of actions such that the goals can be achieved from the initial states. In the case of the pharmacokinetics effects of drugs, the initial state is when a drug is administered and the goal state is when the drug is eliminated after being metabolized. The expected plan is the actions required for the administered drug to be delivered to the systemic circulation for drug consumption. With the acquired facts and interactions and the biological properties of pharmacokinetics in logical representation, the reasoning component of our proposed system arranges the interactions so that a series of interactions is generated as a pathway model.

The rest of the paper is outlined as follows. We describe the basic properties of pharmacokinetics that we encode in our system in Section 2. In Section 3, the processes of acquiring the necessary facts and interactions from existing knowledge bases and Medline abstracts are described. In addition, the reasoning component is illustrated in how the pharmacokinetic properties and behavior are encoded in order to synthesize pathways. In Section 4, we demonstrate the feasibility of our approach by illustrating the precision and recall of generated pathways. We concluded in Section 5.

2. Pharmacokinetics

Pharmacokinetics is concerned with the relationships between various processes during the course of the drug consumption in the body. The study of pharmacokinetics is important to biologists and drug designers, as the bioavailability of a drug, i.e. the effectiveness of a drug when it is absorbed into the systemic circulation, is heavily

[2] Ingenuity Pathway Analysis Tool: http://www.ingenuity.com
[3] Active Motif: http://www.activemotif.com

dependent on the processes involved in pharmacokinetics. When a drug is taken orally, the drug is absorbed in the intestine, and the corresponding drug transporters distribute the drug to the appropriate cellular locations of the intestinal cells. The drug is then delivered to the liver through the bloodstream, and the relevant drug transporters in the liver cells distribute the drug for metabolism by the enzymes. Drugs that are taken intravenously would bypass the drug absorption phase. The pharmacokinetics of a drug includes several processes such as the *distribution* of a drug through different tissues, the *metabolism* of a drug, the *excretion* of a drug, and the *absorption* of a drug into the systemic circulation [19]. Several essential elements are involved in different processes of pharmacokinetics, namely *drug transporters*, *enzymes* and *metabolites*. The typical processes involved in pharmacokinetic pathways are shown in Figure 1. Drug transporters are responsible for drug *distribution* for absorption (as in Region B_1 in Figure 1 (left)), metabolism (Region B_2 in Figure 1 (left)) and excretion (Region B_3 in Figure 1 (left)), and they can be expressed in many tissues such as intestine and liver [20]. Once the target drug is distributed into an appropriate cellular location, the enzymes play the role of metabolizing the drug (as in Region A in Figure 1 (left)), which take place mainly in the liver. Metabolites are produced as a result of the metabolism of the drug, shown in Region C.

Identifying the pharmacokinetic mechanism of a drug is essential in avoiding potential side effects of drug-drug interactions, even though in most cases the processes of drug disposition for co-administered drugs typically do not affect one another. However, drugs that are strong inducers or inhibitors of certain enzymes can influence the bioavailability of drugs that are metabolized by these enzymes [21]. The drug ketoconazole is an example of a powerful inhibitor that is known to inhibit CYP3A enzymes, which are responsible for the metabolism of a wide variety of drugs, such as midazolam. Such inhibition of CYP3A enzymes can affect the drug-metabolizing activity and lead to the increase of the bioavailability CYP3A substrates. On the other hand, drugs that are potent inducers of CYP3A enzymes, such as carbamazepine, can cause a reduction of the effect of CYP3A substrates. With the increasing availability of clinical drugs that are inducers or inhibitors of enzymes, the study of pharmacokinetic drug interactions becomes more critical [21]. While a drug can be involved in various parts of the body, our focus in the synthesis of pharmacokinetic pathways is on the processes involving drug absorption, distribution, metabolism and elimination in the intestine and liver.

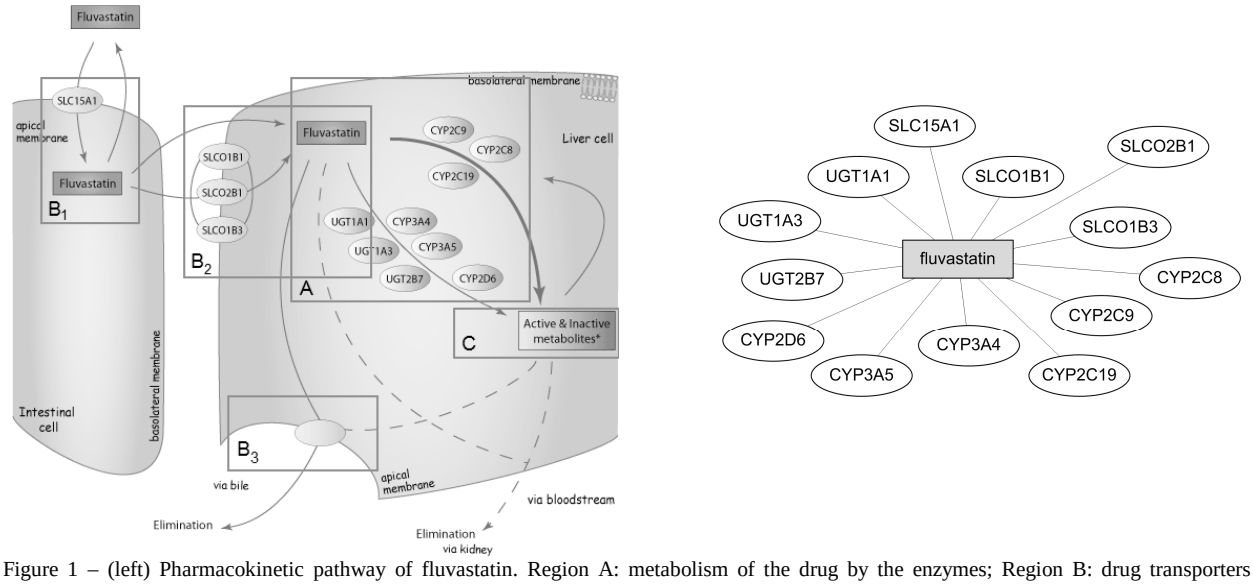


Figure 1 – (left) Pharmacokinetic pathway of fluvastatin. Region A: metabolism of the drug by the enzymes; Region B: drug transporters distribute the drug for absorption in intestine in B_1, for metabolism in B_2 and for elimination in B_3; Region C: the drug is metabolized to metabolites by the enzymes. (Diagram source: PharmGKB); (right) A network representation of the drug-protein interactions for fluvastatin.

3. Methods

The goal of our system is to synthesize the pharmacokinetic pathway of a given drug. Our approach in constructing pathways can be described as a two-stage approach: (i) *fact and interaction extraction* from knowledge bases and text; (ii) inferences of pathways through *reasoning* with the extracted facts and interactions based on the biological

knowledge of pharmacokinetic pathways as described in Section 2. Figure 2 illustrates how the facts and interactions extracted in step (i) are utilized together with the biological background knowledge to construct pathways.

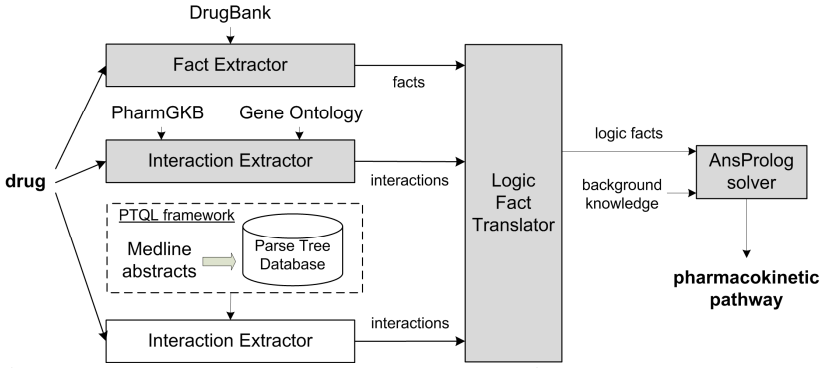


Figure 2 - An overview of the system architecture. Facts and interactions are acquired from knowledge bases such as Drugbank, PharmGKB and Gene Ontology annotations, as well as our PTQL framework for text extraction from Medline abstracts. The shaded components correspond to the novel features of this paper. The facts and interactions are translated into logic facts so that together with the background knowledge, the logic program solver (AnsProlog solver) assigns ordering to the interactions for the synthesis of pharmacokinetic pathways.

The first stage involves various kinds of fact extraction, such as identifying which proteins are drug transporters, as well as interaction extraction, such as finding which enzymes play a role in metabolizing the drug of interest. Fetching the facts and interactions alone leads to the formation of a network of interactions, but the resulting network lacks the information that describes which of the interactions appear ahead of the others. Using the gene-drug interaction network in Figure 1 (right) as an example, the outcome of the extraction process is the interactions between the drug fluvastatin and the proteins, such as SLCO1B1 and CYP3A4. However, with the extracted interactions alone, it is unclear whether the interaction between SLCO1B1 and fluvastatin should precede or follow the interaction between CYP3A4 and fluvastatin. In the synthesis of pathways, ordering of the interactions is essential, as a pathway is a series of interactions, in which the consequence of an interaction affects the subsequent interactions. For instance, the CYP and UGT enzymes in Figure 1 (left) are able to metabolize fluvastatin in the liver cells only if the drug is distributed by the drug transporters SLCO1B1, SLCO2B1 and SLCO1B3. In other words, distribution of the drug in the liver cells is a prerequisite for the drug metabolizing interaction to take place. In our approach, such kind of preconditions and postconditions of interactions are encoded as logic rules so that the reasoning component in step 2 assigns an ordering to the interactions extracted in step 1. We describe the details of each of the steps in the rest of the section.

3.1. *Fact and interaction extraction from knowledge bases*

The first stage of pathway synthesis is to identify and recognize the facts involved in the pharmacokinetic pathways of the drug of interest. As the pharmacokinetics of drug behaves differently depending on its dosage form, it is essential to obtain such information in order to synthesize pathways correctly. Drug metabolism can take place in different organs, and a drug that is known to be metabolized in the liver cells would have a different pathway from a drug that is metabolized in other cells such as the intestinal cells. DrugBank [22] is a rich resource to obtain such kind of facts about drugs. The field “dosage form” in DrugBank provides information such as whether a drug is taken orally or intravenously, and the metabolism of a drug can be obtained from the field “biotransformation”. With DrugBank, logic facts in the form of `is_taken(Drug, Method)` and `metabolism(Drug, Organ)` are generated, in which `Drug` is the name of drug of interest, `Method` is either `orally` or `intravenously` and `Organ` can be `liver` or `intestine`. Using the drug fluvastatin as an example, the facts are written as logic facts `is_taken(fluvastatin, orally)` and `metabolism(fluvastatin, liver)`.

To construct pharmacokinetic pathways, it is important to identify the interactions between drug and enzymes. Several resources are used to obtain interactions between drugs and enzymes. The DrugBank knowledge base provides the metabolizing enzymes as well, but the list of enzymes for each drug is not comprehensive as only the main enzymes are included. Other than DrugBank, PharmGKB [5] is another rich resource that provides information

about genes, drugs and diseases. An extensive list of interactions between drugs and proteins can be found in PharmGKB relationships. The interactions are categorized into several types namely "pharmacokinetics" (PK), "pharmacodynamics" (PD), "molecular and cellular functional assays" (FA) and "clinical outcome" (CO). For the purpose of the pharmacokinetic pathway synthesis, the interactions labeled as pharmacokinetics ("PK") are utilized. However, obtaining the pharmacokinetic interactions is not sufficient for pathway synthesis, as it is important to realize whether the proteins involved in the interactions are enzymes or transporters. Such kind of information can be inferred from the Gene Ontology (GO) annotations [23].

GO is a hierarchy of controlled vocabulary that includes three independent ontologies for biological process, molecular function and cellular component. Standardized terms in GO describe roles of genes and gene products in any organism. Curators annotate the functions of proteins by assigning GO terms, and such annotation is known as GO annotations. The terms "metabolic process" (GO:0008152) and "transporter activity" (GO:0005215) from the GO "biological process" and "molecular function" sub-ontologies can be utilized to identify enzymes and proteins. Given a drug-protein interaction, a protein is considered as an enzyme if the protein is annotated under the subcategories of the term "metabolic process" according to the GO annotation. Similarly, a protein annotated in one of the subcategories of the term "transporter activity" is regarded as a transporter. Since the protein is obtained from a drug-protein interaction, the transporter is regarded as a drug transporter. With PharmGKB and GO, logic facts in the form of `enzyme(Protein)`, `metabolizes(Enzyme, Drug)`, `transporter(Protein)` and `distributes(Transporter, Drug)`are generated. Using the drug fluvastatin as an example, the facts and interactions are written into these logic facts: `enzyme(cyp3a4)`, `metabolizes(cyp3a4, fluvastatin)`, `transporter(slco1b1)` and `distributes(slco1b1, fluvastatin)`.

3.2. *Automated text extraction of facts and interactions*

While the sources provide an extensive amount of information for the synthesis of pharmacokinetic pathways, much of the information is resided in the biomedical literature. In particular, drug transporters, drug-enzyme metabolic relations and the metabolites produced as a result of drug-enzyme metabolic relations can be found in Medline abstracts. Extraction of such facts and interactions requires the utilization of syntactic and lexical clues from sentences. One way of extracting metabolic relations between the target drug and enzymes from text can be performed based on cooccurrences of drug names, gene/protein names and the word "metabolized". However, using cooccurrences is not sufficient for the precision needed in pathway synthesis. Suppose the target drug for extraction is fluvastatin, the relations `metabolizes(CYP2C9, fluvastatin)` and `metabolizes(CYP3A4, fluvastatin)` are extracted from the sentence "*Fluvastatin is metabolized by CYP2C9, while simvastatin, lovastatin and atorvastatin are metabolized by cytochrome P450 3A4 (CYP3A4).*" (PMID:16714062) based on cooccurrences, in which the relation `metabolizes(CYP3A4, fluvastatin)` is an incorrect relation according to the sentence. By utilizing the syntactic pattern that a drug-enzyme metabolic relation is feasible if the word "metabolized" and the gene/protein mention appear in the same verb phrase, then only the correct relation `metabolizes(CYP2C9, fluvastatin)` can be extracted from the sentence. This example shows that it is important to extract drug-enzyme metabolic relations with the use of syntactic patterns. With the diverse extraction needs in the synthesis of pharmacokinetic pathways, it is not feasible to develop individual extraction systems for each specific extraction goal. This implies the need of a flexible system that is designed for generic extraction.

Our PTQL framework [24, 25] provides the flexibility to perform such kind of diverse extraction. A sample list of logic facts that are generated through PTQL extraction is described in Table 1. The central piece of our extraction framework is the parse tree database, which is composed of the syntactic structures for each of the sentences in the entire collection of Medline abstracts. Each of the Medline abstracts is represented as a hierarchical representation with the syntactic and semantic information, which includes the recognition of biological entities. BANNER [26] and MetaMap [27] were used for gene/protein mentions and drug names, while gene/protein mentions are normalized by GNAT [28] for their standardized form in order to avoid name ambiguity. By storing the syntactic and semantic information in the database, performing extraction becomes a matter of writing extraction queries. Since standard relational database queries such as SQL are not ideal for expressing queries that involve linguistic patterns, we

developed a query language called *parse tree query language* (*PTQL*) that are used to express linguistic patterns for extraction. While PTQL queries are expressive, writing the queries manually for extraction is time-consuming and potentially limits the recall of the system. As an alternative, our PTQL framework provides a component that is capable of generating PTQL queries from keyword-based queries. For instance, in the extraction of drug-enzyme metabolic relations, we simply issue the keyword-based query

```
<type:DRUG> and <class:metabolism> and <type:PROTEIN>
```

where `<type:DRUG>` and `<type:PROTEIN>` correspond to matching any mentions of drug name and gene/protein names, and `<class:metabolism>` corresponds to lexical variants of the word "metabolism", which includes "metabolize", "metabolized" and "metabolizes". Using a small corpus of Medline abstracts, the component first retrieves sentences that are relevant to the keyword-based query as in a typical search engine. By utilizing the parse tree database, grammatically similar sentences are retrieved and their common grammatical patterns are utilized in forming PTQL queries automatically. The resulting PTQL queries include the necessary syntactic patterns and they are applied to the full parse tree database that includes all Medline abstracts for extraction. This component is used for the extraction of (i) drug-enzyme metabolic relations; (ii) proteins responsible for drug elimination; (iii) protein expression in liver and intestinal cells. Extraction of these kinds of relations is essential in the synthesis of pharmacokinetic pathways, as existing sources such as DrugBank and PharmGKB lack such information or in a format that is not easily readable by computers.

• *Protein expression in liver and intestinal cells.* In the synthesis of pharmacokinetic pathways, it is essential to find out whether a protein is expressed in the liver or intestinal cells. The following keyword-based query is used for the extraction of protein expression in liver cells:

```
<type:PROTEIN> and <class:liver>
```

The keywords "hepatic" and "liver" are used to represent `<class:liver>`. Similarly, `<class:intestine>` is used for the extraction of protein expression in intestinal cells, in which "intestinal", "gastrointestinal" are used for `<class:intestine>`. With the keyword-based queries, the generated PTQL queries retrieve the sentences and relations as shown in Table 1.

• *Proteins responsible for drug elimination.* Among the interactions between the target drug and its drug transporters, it is necessary to find out the roles of each of the drug transporters, as drug transporters are known to be involved in various roles such as drug distribution, absorption and elimination. Finding the exact roles of transporter is essential for the assignment of the ordering of the interactions. For instance, if a drug transporter is responsible for drug elimination, we know that the drug transporter cannot distribute the drug until the drug is metabolized. Using the keyword-based query

```
<type:PROTEIN> and <class:elimination> and <type:DRUG>
```

where `<class:elimination>` is represented by the keywords "elimination" and "excretion", an example of sentences that indicate the role of drug elimination for a drug transporter is shown in Table 1.

• *Drug-metabolites relations.* For the extraction of relations, typically named entity recognizers are applied to recognize the entities involved in the target relations. However, in the case of metabolites, the lack of dictionaries or training data means that we need to rely on lexical hints for the identification of metabolites. For example, the sentence "The antioxidative effects of the metabolites of fluvastatin (M2, M3, M4 and M7) …" indicates that M2, M3, M4 and M7 are the metabolites of fluvastatin based on the lexical hints "*metabolites of fluvastatin*".

Table 1 – Sample logic facts and evidence sentences for each type of relations that are extracted by our PTQL framework. Note that the protein names have been replaced with their official gene symbols in the logic facts.

Logic facts	Evidence sentences
`metabolizes(CYP3A4, fluvastatin)`	Fluvastatin is metabolized by CYP2C9 (PMID:16714062)
`is_expressed(ABCC2, liver)`	These decreases in hepatic Mrp2 may contribute to cholestasis (PMID:17959626)
`is_expressed(SLC15A1, intestine)`	PPARalpha plays critical roles in intestinal PEPT1 expression. (PMID:16751172)
`eliminates(ABCB1)`	Colchicine is also a substrate of P-glycoprotein, a transporter involved in cellular efflux and elimination of numerous drugs. (PMID:15494379)
`metabolite(desmethyldesipramine, desipramine)`	desipramine in rats may be attributed not only to the inhibition of the norepinephrine transporter by desipramine but also to the inhibition of serotonin transporter by the active metabolite desmethyldesipramine. (PMID: 17850785)

3.3. *Ordering of interactions through reasoning*

Once the relevant facts and interactions are acquired from knowledge bases and Medline abstracts, the last step is to utilize the facts and interactions to generate pharmacokinetic pathways. As the interactions themselves do not reveal any kind of ordering, the goal is to represent the fundamental behavior and properties of pharmacokinetics so that the representation can be utilized to assign ordering of the interactions through reasoning. Implementation of the reasoning component requires a language that is ideal in specifying what kind of reasoning to be performed rather than how the reasoning is performed. This is analogous to declarative programming language such as SQL, in which the users specify what is intended to be found rather than how the search mechanism of the database system should be performed to answer the queries. AnsProlog [29, 30] is a declarative language that is useful for reasoning, as well as capable for reasoning with incomplete information. We first describe how AnsProlog is applied to the representation of pharmacokinetic properties.

The core idea of the representation of pharmacokinetics is to encode the *pre-* and *post-conditions* of interactions, also known as the executability and direct effects of actions. *Timepoints* are used to define the logical ordering of the interactions. Interaction I_1 occurs before interaction I_2 if I_1 is assigned with a timepoint that is smaller than the timepoint for I_2. Using the interaction that involves the generation of metabolites as an example, the pre-condition of such generation is that the target drug has to be metabolized. The post-condition of the interaction is the production of metabolites. Such mechanism is represented by the following AnsProlog logic rules:

```
o(converts(D, M),Loc, T) :- h(metabolized(D),T),
metabolites(D, M), metabolism(D, Loc), not
h(converted(D),T).

h(converted(D),T+1) :- o(converts(D, M), Loc, T),
metabolites(D,M).
```

The first logic rule states that the pre-conditions for the action *converts(D, M)* occur at timepoint *T* in location *Loc* (which can be either the liver or intestinal cell), denoted as `o(converts(D,M),Loc,T)`. For instance, `o(converts(fluvastatin,m2),liver,3)` indicates that the drug fluvastatin is converted into the metabolite M2 in the liver cells at timepoint 3. The following are the pre-conditions, which are specified to the right of the "if" symbol `:-` in the rule, for the action *converts(D, M)*:

- the drug *D* has been metabolized at timepoint *T*, denoted as `h(metabolized(D),T)`;
- metabolite *M* is known to be a metabolite of *D*, denoted as `metabolites(D, M)`;
- metabolism of *D* is known to take place in *Loc*, denoted as `metabolism(D, Loc)`;
- it is not known that *D* has been converted into metabolites in the previous timepoints, denoted as `not h(converted(D),T).`

Notice that `metabolism(D, Loc)` and `metabolites(D, M)` are logic facts that are obtained from the extraction of knowledge bases and text. The second logic rule states the post-condition of the action *converts(D, M)*, which is to indicate the drug *D* is converted in the next timepoint *T*+1 when the action occurs at timepoint *T*. With the timepoints, we can observe that fluvstatin is converted into metabolites with `h(converted(fluvastatin),4)` and this conversion occurs as a result of the action `o(converts(fluvastatin, m2), liver, 3).`

Another example of pharmacokinetic behavior is that a drug can only be metabolized by some enzymes if the drug is distributed to the appropriate location in the liver by a drug transporter. In addition, elimination of the target drug can only take place when the drug is metabolized and metabolites are produced. Our logical representation also includes the fact that an orally-taken drug is transported to the intestines, and drugs that are taken intravenously are transported to the liver. To mimic the behavior of typical pharmacokinetic pathways, we include logic rules to ensure that all interactions involved in the intestinal cells have to occur ahead of the interactions in the liver cells. By encoding rules that represent the pharmacokinetic behavior of drugs, interactions are assigned with timepoints to indicate the ordering of the interactions in the pharmacokinetic pathway. With the logic rules and facts, the model, known as *answer sets* in AnsProlog, is computed by an answer set solver called clasp [31]. The resulting answer sets correspond to the pharmacokinetic pathway of the target drug.

Our approach for pathway synthesis can be summarized in the following steps:

1. Given the drug of interest, knowledge bases that include DrugBank, PharmGKB and the Gene Ontology annotations are utilized to acquire information such as interactions between drug and proteins, as well as the enzymes and drug transporters involved in the interactions.
2. To complement the information fetched from manually curated knowledge bases, information such as protein expression in liver/intestinal cells, the roles of drug transporters, drug metabolites are extracted from Medline abstracts using our PTQL extraction framework.
3. With the generic logical representation of pharmacokinetic pathways, facts and interactions acquired in steps (i) and (ii) are utilized so that the pharmacokinetic interactions are assigned with timepoints to reveal their ordering in the resulting pharmacokinetic pathway.

4. Synthesis of pharmacokinetic pathways

In this section, we illustrate our approach with the synthesis of pharmacokinetic pathways for two drugs: repaglinide and parvastatin. The pathways of another 18 drugs are synthesized by our system and presented as supplementary material in our website: http://www.kbpathway.org/. The pharmacokinetic pathways of these drugs have been manually annotated and made available in PharmGKB. Our system provides the output in the form of logic facts, as well as GPML files that can be visualized in Cytoscape [32] with the Cerebral plug-in [33], which takes advantages of the protein cellular locations in generating the layout of the pathways. We evaluate the performance of our system based on the performance of the extraction.

4.1. *Repaglinide pharmacokinetic pathway*

The logical representation of the pathway generated by our system is shown in Table 2. The model indicates that repaglinide is administered orally in the initial step, represented as `h(is_taken(repaglinide,orally),0)`. The drug consumption leads to the presence of the drug in the intestinal cells (`h(is_present(repaglinide, intestine),1)`), and the drug is transported to the liver cells through the bloodstream (`o(transports(repaglinide,intestine),liver,1)`). The drug repaglinide becomes present in the liver cells (`h(is_present(repaglinide,liver),2)`) and it is distributed by the hepatic drug transporter SLCO1B1 (`o(distributes(slco1b1,repaglinide),liver,2)`). Metabolism of repaglinide by the enzymes CYP3A4 and CYP2C8 (`o(metabolizes(cyp3a4,repaglinide),liver,2)`, `o(metabolizes(cyp2c8, repaglinide),liver,2)`) occurs after the distribution, and repaglinide becomes metabolized (`h(metabolized(repaglinide,liver),3)`). As a result of the drug metabolism, metabolites M1 and M4 are generated (`o(converts(repaglinide,m1),liver,3)`, `o(converts(repaglinide,m4),liver,3)`). The last timepoint indicates that repaglinide is no longer present in the liver, represented by `-h(is_present( repaglinide,liver),4)`, in which the symbol "`-`" corresponds to negation.

Table 2 – The output of the logical representation of the pharmacokinetic pathway of repaglinide generated by our system.

Timepoint	Events
0	h(is_taken(repaglinide,orally),0).
1	h(is_present(repaglinide,intestine),1). o(transports(repaglinide,intestine),liver,1)
2	-h(is_present(repaglinide,intestine),2). h(is_present(repaglinide,liver),2). o(distributes(slco1b1,repaglinide),liver,2). o(metabolizes(cyp3a4,repaglinide),liver,2). o(metabolizes(cyp2c8,repaglinide),liver,2).
3	h(metabolized(repaglinide,liver),3). o(converts(repaglinide,m1),liver,3). o(converts(repaglinide,m4),liver,3).
4	-h(is_present(repaglinide,liver),4).

The manually annotated pathway and the version synthesized by our system are shown in Figure 3. One distinctive difference is that our current approach is not capable of finding which of the enzymes are responsible for which metabolites. Here we assume that the enzymes CYP3A4 and CYP2C8 are responsible for the metabolism of repaglinide, and metabolites M1 and M4 are generated in the process. In the next illustration, we show that our automated pathway synthesis approach is capable of uncovering components that are not described in manually annotated pathways.

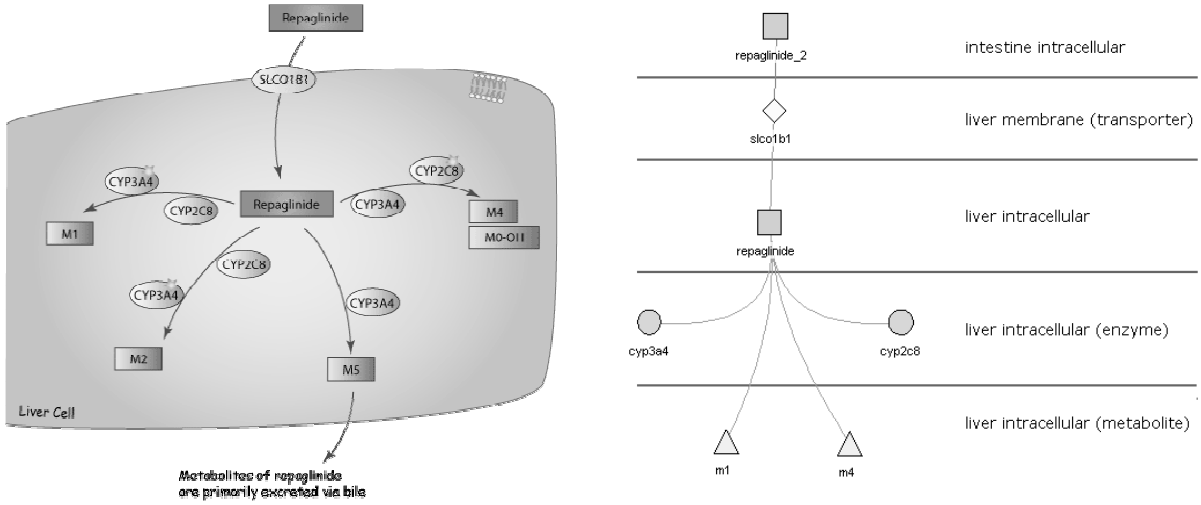


Figure 3 – (left) The manually curated pharmacokinetic pathway of the drug repaglinide from PharmGKB; (right) the pharmacokinetic pathway of repaglinide synthesized by our system.

4.2. *Pravastatin pharmacokinetic pathway*

Here we demonstrate that our method is capable of producing extra elements in the pathways compared to the manually annotated pathways. We use the synthesis of pravastatin pharmacokinetic pathway as an example. The manually annotated pathway in Figure 4 (left) lacks the information that states which enzymes are responsible for metabolism and what metabolites are generated as a result of the metabolism. Such information is included in our synthesized version of the pathway. As shown in Figure 4 (right), enzymes CYP3A4, CYP2C8 and CYP2C9 metabolize the drug pravastatin, and it is metabolized into the metabolites SN-38, 3alpha-hydroxy-iso-pravastatin and 3'alpha-iso-pravastatin. The drug-enzyme metabolic relations are originated from PMID: 17178259 according to the relationships in PharmGKB corresponding evidences. The resulting metabolites are extracted by our PTQL extraction framework, and the evidence sentences in Table 3 indicate the correctness of these facts and interactions.

Table 3 – Evidence sentences for the metabolites (underlined) of pravastatin extracted by our PTQL extraction framework.

PMID	Evidence	Correctness
16027406	Plasma concentrations of *pravastatin* and its active *metabolite*, 3alpha-hydroxy-iso-pravastatin, were measured, and pharmacokinetics was assessed.	Correct
10490896	In addition, as in the liver, *pravastatin* was metabolized in the small intestine by sulfation and subsequent degradation to its main *metabolite* 3'alpha-iso-pravastatin.	Correct
16515396	These genetic variants have been shown to lead to altered pharmacokinetics of OATP1B1 substrates, mainly *pravastatin*, but also the irinotecan metabolite SN-38	Incorrect

4.3. *Evaluation and analysis*

We evaluate the performance of our pathway synthesis method by finding how many of the interactions can be recovered (i.e. the coverage) with respect to 20 pharmacokinetic pathways available in PharmGBK. Table 4 shows the coverage when each of the sources DrugBank, PharmGBK relations and PTQL extraction is utilized for pathway synthesis. While the use of PharmGKB relations achieves the best coverage of the three sources with 47.27%, the coverage for the three sources combined results in 56.97%. We further manually evaluated the extraction performance by our PTQL framework. As DrugBank and PharmGKB do not provide information about drug metabolites, it is essential for the PTQL framework to be capable of extracting metabolites. Another important evaluation criterion is to determine if such approach in pathway synthesis can uncover more information than manually annotated pathways. Table 5 indicates that our extraction framework achieves a high precision of 84.0% and 82.72% for the extraction of enzymes and transporters, and metabolites. In particular, among the enzymes, transporters and metabolites uncovered by our method, our system produces 24 extra enzymes and transporters as well as 48 metabolites that are correct but not included in the 20 manually annotated pathways. The high quality of

extraction suggests that using PTQL framework for extraction is a valuable source to complement the existing knowledge bases for pathway synthesis.

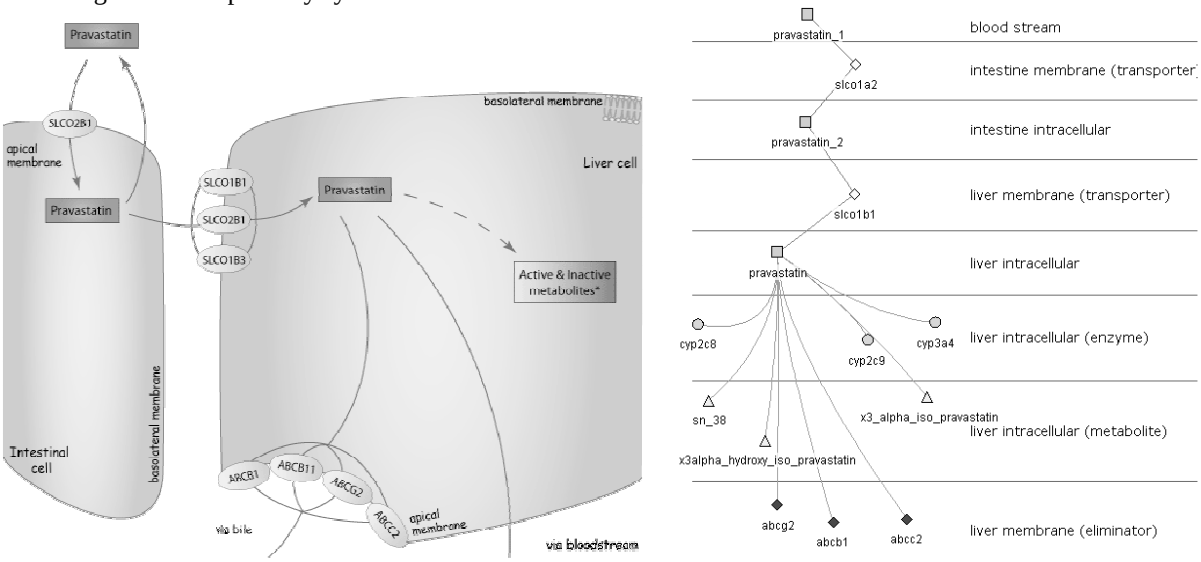


Figure 4 – (left) The manually curated pharmacokinetic pathway of the drug pravastatin from PharmGKB; (right) the pharmacokinetic pathway of pravastatin synthesized by our system.

Table 4 – Coverage of each of the sources in the pharmacokinetic pathways for the 20 manually annotated pharmacokinetic pathways in PharmGKB.

Sources	Coverage
DrugBank	20.61%
PTQL extraction	34.23%
PharmGKB	47.27%
All	56.97%

Table 5 – Precision and recall for PTQL extraction of enzymes and transporters, as well as metabolites. The ones that are not in the PharmGKB annotated pathways are considered as "Extra", and their correctness is evaluated.

	Precision	Recall	Extra (Precision)
Enzymes and transporters	84.00%	34.23%	19/31 = 61.29%
Metabolites	82.72%	65.05%	48/62 = 77.42%

We further analyzed the correctness of the placement of the interactions in the synthesized pathways compared to 20 pharmacokinetic pathways available in PharmGKB. Our synthesized pathways are generally consistent with the annotated pathways. However, our synthesized pathways do not include information that specifies which enzymes are responsible for the production of a particular drug metabolite. Such limitation is due to the fact that metabolites are not found in DrugBank and PharmGKB relationships, and our current text extraction is limited to the extraction from individual sentences. Drug-enzyme-metabolite relations can rarely be found within individual sentences.

Among the 20 pharmacokinetic pathways we evaluate, 2 of them are taken intravenously according to DrugBank, namely imipramine and irinotecan. In our logical representation of pharmacokinetics, we assume that no interactions occur in the intestine for drugs that are taken intravenously, unlike the orally-taken drugs that require absorption in the intestine. The imipramine pathway in PharmGKB only shows the interactions in liver cells, but the irinotecan pathway includes interactions in both the intestinal and liver cells, with the interactions in the liver cells appear ahead of the ones in the intestinal cells. Our current modeling of pharmacokinetic properties does not capture this behavior. In terms of metabolism, 17 of the drugs we evaluate are known to be metabolized in the liver cells according to PharmGKB, so by default we assume the metabolism of the other 3 drugs, namely atorvastatin, repaglinide, rosuvastatin, takes place in the liver cells. This assumption is valid except for the atorvastatin, lovastatin and simvastatin pathways that indicate the drugs are metabolized in both the intestinal and liver cells. In our current modeling, we assume that drug metabolism and drug distribution for elimination take place in the same cell. This is

not the case for the drugs clopidogrel, fluvastatin and pravastatin, in which the manually annotated pathways suggest that drug transporters take the drug from intracellular to extracellular in the intestinal cells, even metabolism of these drugs occur in the liver cells. Our current model also does not capture the transformation of a metabolite to another through enzymes, as suggested by the pathways for phenytoin and tamoxifen.

5. Conclusion

The study of pharmacokinetics is essential in identifying the effectiveness of drugs in the systemic circulation. In particular, variability in drug response is largely influenced by genetics. In this paper, we extend our previous work in synthesizing biological networks [25] by including a reasoning component for the synthesis of pharmacokinetic pathways. The use of reasoning distinguishes our approach from existing methods in generating networks of interactions so that ordering of interactions can be assigned through reasoning. Such ordering is critical for the representation of pathways, in which the effects of interactions trigger the subsequent interactions. Our results show that our approach is capable of synthesizing pharmacokinetic pathways in high quality and identifying components that are not in the manually annotated pathways. With the partial pathways generated by our approach, curators can utilize the synthesized pathways as a first step of curation and add their findings to expand the pathway annotation. Through the synthesized pharmacokinetic pathways, drug designers can examine the impact of drug response due to the genetic variations of the gene products involved in the pathways. Identifying relations between drug response and genetic variations is a critical step in realizing personalized medicines.

For future work, we will implement a web-based version of our approach so that pharmacokinetic pathways can be created based on the drugs specified by the users. Announcements will be made on our website at http://www.kbpathway.org when the web-based version of the implementation becomes publicly available. We also plan to expand our work to handle close-loop interactions, which cannot be captured in our current approach. Information such as drug-drug interactions will be included to identify drugs that inhibit or induce enzymes responsible for the metabolism of other drugs. Such information can be useful to drug designers as well as physicians to learn the potential side-effects of drugs due to drug-drug interactions. We also plan to apply our approach to other kinds of pathways, such as pharmacodynamics and signaling pathways.

Acknowledgements

We would like to thank the comments made by the anonymous reviewers. We would also like to acknowledge the support of these research grants: NSF 0412000, SFAZ CAA 0131-07 and SFAZ CAA 0289-08.

References

1. G. Joshi-Tope, M. Gillespie, I. Vastrik, et al. Reactome: a knowledgebase of biological pathways. *Nucl. Acids Res.*, **33**, suppl 1, D428-432 (2005).
2. M. Kanehisa, S. Goto, S. Kawashima, et al. The KEGG resource for deciphering the genome. *Nucleic Acids Res. Database Issue*, **32**, D277-80 (2004).
3. P. Romero, J. Wagg, M. L. Green, et al. Computational prediction of human metabolic pathways from the complete human genome. *Genome Biol.*, **6**, 1, R2 (2005).
4. H. Mi, N. Guo, A. Kejariwal, et al. PANTHER version 6: protein sequence and function evolution data with expanded representation of biological pathways. *Nucleic Acids Res.*, **35**, D247 (2007).
5. T. E. Klein, J. T. Chang, M. K. Cho, et al. Integrating genotype and phenotype information: an overview of the PharmGKB project. *Pharmacogenomics Journal*, **1**, 3, 167 (2001).
6. P. Uetz, L. Giot, G. Cagney, et al. A comprehensive analysis of protein--protein interactions in Saccharomyces cerevisiae. *Nature*, **403**, 6770, 623-627 (2000).
7. D. Rajagopalan and P. Agarwal. Inferring pathways from gene lists using a literature-derived network of biological relationships. *Bioinformatics*, **21**, 6, 788-793 (2005).

8. C. Friedman, P. Kra, H. Yu, et al. GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, **17 Suppl 1**, S74-82 (2001).
9. J. C. Park, H. S. Kim and J. J. Kim. Bidirectional incremental parsing for automatic pathway identification with combinatory categorial grammar. *Pac. Symp. Biocomputing*, 396-407 (2001).
10. M. Theobald, N. Shah and J. Shrager. Extraction of Conditional Probabilities of the Relationships Between Drugs, Diseases, and Genes from PubMed Guided by Relationships in PharmGKB. *AMIA* (2009).
11. G. A. Viswanathan, J. Seto, S. Patil, et al. Getting started in biological pathway construction and analysis. *PLoS Computational Biology*, **4**, 2 (2008).
12. A. Funahashi, M. Morohashi, H. Kitano, et al. CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. *Biosilico*, **1**, 5, 159-162 (2003).
13. T. Aittokallio and B. Schwikowski. Graph-based methods for analysing networks in cell biology. *Brief Bioinform*, **7**, 3, 243-255 (2006).
14. X. Guo and A. J. Hartemink. Domain-oriented edge-based alignment of protein interaction networks. *Bioinformatics*, **25**, 12, 240-1246 (2009).
15. R. Sharan, S. Suthram, R. M. Kelley, et al. Conserved patterns of protein interaction in multiple species. *Proc. Natl. Acad. Sci. U. S. A.*, **102**, 6, 1974-1979 (2005).
16. M. L. Mavrovouniotis, G. Stephanopoulos and G. Stephanopoulos. Computer-aided synthesis of biochemical pathways. *Biotechnol. Bioeng.*, **36**, 1119-1132 (1990).
17. N. Tran, C. Baral, V. J. Nagaraj, et al. Knowledge-based framework for hypothesis formation in biochemical networks. *Bioinformatics*, **21**, suppl 2, ii213-219 (2005).
18. K. Oda, J. D. Kim, T. Ohta, et al. New challenges for text mining: mapping between text and manually curated pathways. *BMC Bioinformatics*, **9 Suppl 3**, S5 (2008).
19. Z. A. Sharif. Pharmacokinetics, metabolism, and drug-drug interactions of atypical antipsychotics in special populations. *J. Clin. Psychiatry*, **5**, 22-25 (2003).
20. N. Mizuno, T. Niwa, Y. Yotsumoto, et al. Impact of Drug Transporter Studies on Drug Discovery and Development. *Pharmacol. Rev.*, **55**, 3, 425-461 (2003).
21. D. Greenblatt, L. von Moltke, J. Harmatz, et al. Pharmacokinetics, pharmacodynamics, and drug disposition. *Neuropsychopharmacology: the Fifth Generation of Progress*, 507–24 (2002).
22. D. S. Wishart, C. Knox, A. C. Guo, et al. DrugBank: a comprehensive resource for in silico drug discovery and exploration. *Nucl. Acids Res.*, **34**, suppl 1, D668-672 (2006).
23. E. Camon, D. Barrell, V. Lee, et al. The Gene Ontology Annotation (GOA) Database - An integrated resource of GO annotations to the UniProt Knowledgebase. *Silico Biology*, **4**, 1, 5-6 (2004).
24. P. H. Tu, C. Baral, Y. Chen, et al. Generalized text extraction from molecular biology text using parse tree database querying. *Arizona State University*, **TR-08-004** (2008).
25. L. Tari, J. Hakenberg, G. Gonzalez, et al. Querying parse tree database of Medline text to synthesize user-specific biomolecular networks. *Pacific Symposium on Biocomputing (PSB'09)*. (2009).
26. R. Leaman and G. Gonzalez. BANNER: An executable survery of advances in biomedical named entity recognition. *Pacific Symposium on Biocomputing (PSB'08)*, 652-663 (2008).
27. Aronson, A. R. *MetaMap: Mapping Text to the UMLS Metathesaurus.* (2006).
28. J. Hakenberg, C. Plake, R. Leaman, et al. Inter-species normalization of gene mentions with GNAT. (2008).
29. M. Gelfond and V. Lifschitz. The Stable Model Semantics for Logic Programs. *International Symposium on Logic Programming*, 1070-1080 (1988).
30. M. Gelfond and V. Lifschitz. Classical Negation in logic programs and disjunctive databases. *New Generation Computing*, 365-387 (1991).
31. M. Gebser, B. Kaufmann, A. Neumann, et al. clasp: A Conflict-Driven Answer Set Solver. *Proceedings of the Ninth International Conf. on Logic Programming and Nonmonotonic Reasoning (LPNMR'07)*, 260-265 (2007).
32. P. Shannon, A. Markiel, O. Ozier, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, **13**, 11, 2498-2504 (2003).
33. A. Barsky, J. L. Gardy, R. E. W. Hancock, et al. Cerebral: a Cytoscape plugin for layout of and interaction with biological networks using subcellular localization annotation. *Bioinformatics*, **23**, 8, 1040-1042 (2007).

IN SILICO BIOLOGY

CHRISANTHA T. FERNANDO
Department of Informatics, University of Sussex
Falmer, Brighton, BN1 9RH, United Kingdom

Mathematical Biology, National Institute for Medical Research
Mill Hill, London, NW7 1AA, United Kingdom

Mathematical Collegium Budapest (Institute for Advanced Study),
Szentháromság u. 2, H-1014, Budapest, Hungary

RICHARD A. GOLDSTEIN
Mathematical Biology, National Institute for Medical Research
Mill Hill, London, NW7 1AA, United Kingdom

PHIL HUSBANDS
Department of Informatics, University of Sussex
Falmer, Brighton, BN1 9RH, United Kingdom

DOV J. STEKEL
School of Biosciences, University of Nottingham, Sutton Bonington Campus
Nottingham, LE12 5RD, United Kingdom

Rather than studying existent living systems, we can increasingly produce computer models that capture the salient aspects of life. This provides us with unprecedented opportunities to examine, manipulate, and explore biological phenomena, allowing us to investigate some of the deepest issues in biology.

1. Introduction

It's life, Jim, but not as we know it [1].

The organisms that we study today are the outcome of a specific, historically contingent process of billions of years of evolution. Over its history, biology has mostly involved the description and classification of observed life forms. We have made great progress in that time in expanding the scale of our observations, looking at a much wider range of organisms using increasingly precise and detailed measurement techniques. Especially with the increasingly data rich experimental techniques of genetics and molecular biology, we have coupled these investigations with mathematical modelling and computer simulations, developing our understanding of increasingly large and complex biological systems, elucidating the functions of these systems, and proposing hypotheses about their evolution.

Despite these advances, there are severe limitations on our current techniques. Firstly, there is much information about living systems that is either practically or inherently difficult or impossible to obtain. We cannot measure systems with arbitrary precision. Many measurement techniques either look at bulk properties of large ensembles of components, or alternatively, measure the distinct properties of a small number of these components, rather than allowing us to look in a comprehensive way at distributions (including outliers). Measurement methods explore a limited range of timescales, and cannot be arbitrarily extended to very slow or very fast processes. It is generally impossible to study a system without the examination itself perturbing the object of study. Studying living systems are particularly difficult, as the requirements of life are incompatible with many investigative techniques.

Conversely, there are many manipulations that we might wish to make but cannot. We cannot arbitrarily change environmental conditions. We cannot change the properties of the physical components that make up living systems.

We cannot delete species from food webs. We cannot investigate the role of stochastic fluctuations by removing them from the system.

Evolutionary investigations offer their own difficulties. We cannot directly explore the process that gave rise to current organisms, running evolution under controlled and monitored conditions on the time scales for which observed organisms have evolved. We cannot know the relative importance of 'Chance and Necessity' [2] in explaining observed biology. We cannot explore avenues that evolution might have taken under different circumstances; we cannot 'play the tape twice' [3].

Finally, it is difficult to deduce general principles from the single example represented by the current set of evolutionarily related organisms. We cannot experimentally determine whether universal features found in all living organisms represent absolute constraints, common responses to a common situation, or a randomly chosen alternative maintained through a common genetic descent. We cannot address questions about what would have happened if it had arisen under different circumstances.

Rather than analysing specific living systems, we can try to create artificial systems that have properties characteristic of natural living systems [4]. Experimentalists have had some success investigating fundamental issues in evolutionary biology through, for instance, directed in vivo evolution [5, 6]. Progress has also been made by either building up cells from non-living components [7, 8] or stripping existent cells of all but the most essential ingredients [9]. The experimental challenges of this work are daunting, and there remain many limitations to the possibilities that can be explored.

An alternative is *in silico* biology, creating computational simulations of systems that embody important properties of biological systems, such as homeostasis, reproduction, evolution, etc. This approach allows us to explore, investigate, and manipulate the workings of such systems without being restricted to the particular examples nature has made available to us. We can explore the processes of Darwinian evolution by studying digital 'organisms' on true evolutionary timescales not possible in the laboratory [10], including their origin [11-15], evolution [16-18], development [19, 20], behaviour [21], and interactions [22, 23]. The aim of these simulations is often the discovery of general or fundamental principles and organization of life, rather than purely the elucidation of the workings of a specific example.

In silico biology represents a movement of biology from an analytical to a synthetic framework as we create artificial systems that exhibit biological behaviour. This shift has three consequences. Firstly, we can greatly expand our subjects of investigation beyond those that are naturally available. In this way, we can go beyond the idiosyncratic examples provided by nature, allowing us to generate a set that is both more varied and more systematic. Secondly, it is by attempting to create artificial systems that embody many of the phenomena of life that we most rigorously test our understanding of biology, investigate what is sufficient, what is necessary, what is understood, what is missing. Finally, in addition to providing a deeper theoretical foundation for biology, these activities can provide important tools for computer science as evidenced by the success of various evolutionary computational methods.

2. Work in *in silico* biology

The first investigations of in silico biology were the introduction of automata by John von Neumann in the late 40s. . Since then, this field has grown to include many topics, including

- Evolution of diversity and complexity. Evolution by natural selection has produced a biosphere with more than 10 million species and organisms ranging in size by 10 orders of magnitude. Yet although evolution underpins all of biology, we do not yet understand how and why the observed diversity and complexity arises. A major contribution of in silico Biology is the capacity to explore our understanding of evolution by testing what factors are necessary to observe the emergence of complex life forms. Specifically, we are interested in understanding how it is possible for complex traits to emerge in an open-ended manner [24], or indeed why complexity emerges at all [25]. Particularly important is garnering an understanding of how and why the major transitions in evolution, such as the evolution of multicellularity or of sexual reproduction, have arisen [26]. Because these cannot be repeated, in silico techniques are particularly valuable.

- The evolution of evolvability: Selective pressures exist at different levels, including genetic, individual, kinship group, lineage and species group. In silico biology can help us to understand the relationship between these levels of selection [27]. Of particular interest is understanding whether the capacity for evolution is itself a property that evolves [28-30], and, if so, under what circumstances [17].
- Biological Networks: What governs the properties of networks of dynamical elements, such as those that occur in biochemical and gene-regulatory networks [31, 32]? How to they evolve [33]? Why are some network architectures much more common than would be expected at random [34, 35]? How do we understand the robustness and modularity observed in such networks [36]? How does gene-expression produce morphology [37]? How can we analyse biological networks [38]?
- Co-evolution: How do we understand the relationship between hosts and pathogens [39]? How has this relationship determined our evolutionary process? How does this affect our expectations about newly emergent diseases?
- Learning and Evolution: How does learning and the ability to learn affect the evolutionary process [40-44]? How are collections of relatively simple organisms (such as an ant colony or wasp hive) able to exhibit intelligent behaviour far in advance of that of the individuals? Can bacteria undertake associative learning? [45-47].
- The Evolution of Communication and Signalling: How does communication arise [48, 49]? What are the necessary conditions for its emergence [50-52]?
- The Origin of Life: What are the most basic features required for something to be 'living' [14]? How did life arise [26]? How did nucleotides arise [13]? What was the function of the first replicase ribozyme [53]?
- Emergence: One theme throughout in silico biology is the important process by which complex behaviour can result through the interactions of a large number of relatively simple components. How can we understand emergence [54]?

3. Things as they are, things as they are not

You see things; and you say 'Why?' But I dream things that never were; and I say 'Why not?' [55]

The question that the Serpent asks Eve in Shaw's play makes a distinction between 'Why' and 'Why not' questions. But in reality, these questions are often the same. If we wish to understand why things are the way they are, we need to look at alternatives and consider the reasons that these alternatives did not occur. Unfortunately, the world has only provided us with a limited set of highly related living organisms. *In silico* biology can provide further alternatives, giving us increased understanding regarding the conditions under which such alternatives might have arisen, and why the currently observed organisms arose instead.

References

1. *Star Trek*, G. Roddenberry, creator.
2. J. Monod, *Chance and Necessity: An Essay on the Natural Philosophy of Modern Biology*, Vintage, London (1972).
3. W. Fontana and L. W. Buss, *Proc. Natl. Acad. Sci. USA,* **91**, 757 (1994).
4. R. E. Lenski, C. Ofria, T. C. Collier and C. Adami, *Nature* **400**, 661 (1999).
5. M. Travisano, J. A. Mongold, A. F. Bennett and R. E. Lenski, *Science,* **267**, 87 (1995).
6. J. A. G. M. De Visser and D. E. Rozen, *J. Evol. Biol.,* **18**, 779 (2005).
7. C. Fernando, M. Santos and E. Szathmáry, *Top. Curr. Chem. ,* **259**, 167 (2005).
8. P. L. Luisi, *The Emergence of Life: from Chemical Origin to Synthetic Biology*, Cambridge University Press, Cambridge(2006).
9. A. C. Forster and G. M. Church, *Mol. Syst. Biol. ,* **2** (2006).
10. E. Szathmáry, *Philos. Trans. R. Soc. London. B. Biol. Sci. ,* **361**, 1761 (2006).
11. E. Szathmary and J. Maynard Smith, *J. Theor. Biol.,* **187**, 555 (1997).
12. C. Fernando and J. Rowe, *J. Theor. Biol.* **247**, 152 (2007).
13. C. Fernando and J. Rowe, *Biosys.,* **91**, 355 (2008).
14. T. Gánti, *The Principles of Life*, Oxford University Press, Oxford, UK (2003).

4
15. R. Lathe, *Icarus,* **168**, 18 (2003).
16. C. Adami, *Introduction to Artificial Life*, Springer (1999).
17. M. Lynch, *Proc. Natl. Acad. Sci. USA* **104**, 8597 (2007).
18. J. P. Crutchfield and P. Schuster, *Evolutionary Dynamics*, OUP (2002).
19. K. Stanley and R. Miikkulainen, *Artificial Life,* **9**, 93 (2003).
20. I. Salazar-Ciudad and J. Jernvall, *Evol. Devel.* **6**, 6 (2004).
21. U. Alon, *An Introduction to Systems Biology: Design Principles of Biological Circuits*, Chapman and Hall, (2006).
22. J. Hofbauer and K. Sigmund, *Evolutionary games and population dynamics*, Cambridge University Press, Cambridge (1998).
23. R. Axelrod, *The Evolution of Cooperation*, Basic Books (1985).
24. M. A. Bedau, *Artificial Life,* **4**, 125 (1998).
25. C. Adami, C. Ofria and T. C. Collier, *Proc Natl Acad Sci U S A.,* **97**, 4463 (2000).
26. J. Maynard Smith and E. Szathmáry, *The Major Transitions in Evolution*, Oxford University Press, Oxford (1995).
27. S. Okasha, *Evolution and the levels of selection*, Oxford University Press, Oxford (2006).
28. M. Pigliucci, *Nat. Rev. Genet.,* **9**, 75 (2008).
29. A. G. Jones, S. J. Arnold and R. Bürger, *Evolution,* **61**, 727 (2007).
30. W. Zhu and S. Freeland, *J. Theor. Biol.,* **239**, 63 (2006).
31. R. Thomas, *Int. J. Dev. Biol. ,* **42**, 479 (1998).
32. T. Lenser, T. Hinze, B. Ibrahim and P. Dittrich, in *Proceedings EvoBio* Vol. LNCS 4447, Marchiori, E., Moore, J. H., and Rajapakse, J. C., Eds., Springer Verlag, Valencia, pp. 132 (2007).
33. M. M. Babu, S. A. Teichmann and L. Aravind, *J Mol Biol,* **358**, 614 (2006).
34. N. Kashtan and U. Alon, *Proc Natl Acad Sci USA,* **102**, 13773 (2005).
35. O. Sporns and R. Kotter, *PLoS Biology,* **2**, e369 (2004).
36. A. Wagner, *Robustness and Evolvability in Living Systems*, Princeton University Press, Princeton (2007).
37. G. B. Muller, *Nat. Rev. Genet.,* **8**, 943 (2007).
38. J. Bongard and H. Lipson, *Proc Natl Acad Sci U S A.,* **104**, 9943 (2007).
39. M. Nowak, *Evolutionary Dynamics: Exploring the Equations of Life*, Harvard University Press, Cambridge (2006).
40. P. Adams, *J Theor Biol,* **195**, 419 (1998).
41. M. J. Baldwin, *Psychol. Rev.,* **16**, 207 (1909).
42. M. J. Baldwin, *Am. Natur.,* **30**, 441 (1896).
43. M. J. Baldwin, *Psycholog. Rev.,* **5**, 4 (1898).
44. G. E. Hinton and S. J. Nowlan, in *Adaptive Individuals in Evolving Populations. Models and Algorithms*, Addison-Wesley Longman, 447 (1987).
45. I. Tagkopoulos, Y.-C. Liu and S. Tavazoie, *Science,* **320**, 1313 (2008).
46. C. Fernando, A. M. L. Liekens, L. E. H. Bingle, C. Beck, T. Lenser, D. J. Stekel and J. E. Rowe, *J. Roy. Soc. Interface* **6**:463 (2009).
47. N. Gandhi, G. Ashkenasy and E. Tannenbaum, *J. Theor. Biol.,* **249**, 58 (2007).
48. H. Brighton, in *Theoretical and Applied Linguistics*, Vol. PhD, The University of Edinburgh, Edinburgh (2003).
49. L. Steels, R. van Trijp and P. Wellens, in *ECAL07*, Almeida e Costa, F. a. R., L.M. and Costa, E. and Harvey, I, Ed., Springer Verlag. , pp. 425 (2007).
50. L. Steels, *Trends in Cognitive Sciences,* **10**, 347 (2006).
51. M. Oliphant and J. Batali, *The newsletter of the Center for Research in Language,* **11**, 1 (1997).
52. E. Szathmary and S. Szamado, *Nature,* **456**, 40 (2008).
53. C. Fernando, G. Von Kiedrowski and E. Szathmary, *J. Mol. Evol.,* **64**, 572 (2007).
54. S. McGregor and C. Fernando, *J. Artif. Life,* **11**, 459 (2005).
55. G. B. Shaw, *Back to Methuselah,* (1921).

GENOMIC STANDARDS CONSORTIUM WORKSHOP: METAGENOMICS, METADATA AND METAANALYSIS (M3)

PETER STERK
NERC Centre for Ecology and Hydrology, Oxford, OX1 3SR, United Kingdom

LYNETTE HIRSCHMAN
Information Technology Center, The MITRE Corporation, 202 Burlington Road Bedford, MA 01730, USA

DAWN FIELD
NERC Centre for Ecology and Hydrology, Oxford, OX1 3SR, United Kingdom

JOHN WOOLEY
University of California San Diego, 9500 Gilman Drive La Jolla, CA 92093, USA

The M3 workshop has, as its primary focus, the rapidly growing area of metagenomics, including the metadata standards and the meta-analysis approaches needed to organize, process and interpret metagenomics data. The PSB Workshop builds on the first M3 meeting, a Special Interest Group (SIG) meeting at ISMB 2009, organized by the Genomics Standards Consortium.

1. M3: Metagenomics, Metadata, MetaAnalysis

1.1. *Background*

There are now thousands of genomes and metagenomes available for study (see the Genomes Online Database (http://www.genomesonline.org/) [1]. Interest in improved sampling of diverse environments (e.g. ocean, soil, sediment, and a range of hosts) combined with advances in the development and application of ultra-high throughput sequencing methods will vastly accelerate the pace at which new metagenomes are generated. For example, in 2007, the Global Ocean Survey published scientific analyses of 41 metagenomes [2], and as of October 2008, the submission of user-generated metagenomes to the public MG-RAST annotation server surpassed 1300 [3]. We have entered an era of 'mega-sequencing projects' that now include funded projects like the Genomic Encyclopedia of Bacteria and Archaea (GEBA) project and the Human Microbiome Project, with many more equally visionary projects on the horizon.

While a genome represents the full genetic (DNA) complement of a single organism, metagenomes represent the DNA of an entire community of organisms. Metagenomes are partial samples of complex and largely unknown communities that can often only be poorly assembled. Genome and metagenomes are now also being complemented with studies of metatranscriptomes (community transcript profiles) and metaproteomes (community protein profiles). The integrative study of these datasets including those from multi-omic experiments of the same biological samples, bring with them the demand for new computational approaches. These data hold the promise of unparalleled insights into fundamental questions across a range of fields including evolution, ecology, environmental biology, health and medicine. Advances stem from improved understandings of the combinations, abundances and functions of the organisms in these communities and their genes and pathways. We are just starting to exploit these new technologies to understand the microbial world, their role in climate and biogeochemical processes and potential for bioenergy sources, and their implications for human health. The data sets will transform our knowledge of microbial evolution and ecology, since we have only scratched the surface in terms of sampling

natural microbial diversity in terms of space and time. At the same time, the emerging science of metagenomics offers insight into eukaryotes by way of the roles of their microbiota, both mutualistic and pathogenic. The rapid pace of genomic and metagenomic sequencing projects [4], which now include studies of microbiomes, will only increase as the use of ultra-high-throughput sequencing methods becomes more common place. Therefore, the role of standards becomes even more vital to scientific progress and data sharing. It is clear that we need new standards to capture additional contextual data as well as tools to support its use in downstream computational analyses.

1.2. *The PSB M3 Workshop*

The M3 Workshop at PSB 2010 builds directly on the past GSC workshops and the ISMB M3 SIG. Its focus is on comparative studies of (meta)genomes that bring these sequences into "context" (i.e., by geolocation, habitat, organism phenotype, etc). For example, a seminal paper, illustrating the power of this approach, has recently been published in PNAS [5]. It reports a study aimed at elucidating the relationships between metabolic pathways and environmental parameters in microbial communities using the data and metadata from the Global Ocean Survey (GOS), an earlier landmark paper in the history of the field of metagenomics [2]. The kick-off of the Human Microbiome Project and the resulting data sets will open enormous new possibilities for integration and analysis of metagenomic data sets in context.

The agenda of this M3 workshop has been designed to cover the growing intersection of science and standards. The workshop combines talks selected from abstract submissions, and a panel discussion to give a "voice" to the community. Building such community-driven consensus, in the form of standards that support and accelerate scientific discovery in biology, is of growing importance. This is especially true given the rapid growth of experimental data, most notably including both genomic and metagenomic sequences.

1.3. *The Genomic Standards Consortium*

The establishment of the Genomic Standards Consortium in late 2005 and its growing membership and activities attests to the growing interest in this area and the willingness of a wider range of researchers to become involved in this area of work. The GSC has largely been an activity centered in Europe to date (the UK) with strong involvement from the US. We feel it is essential to encourage the involvement of researchers across Asia, especially given the growing investments in genomic technologies in this area of the world. The PSB offers an ideal opportunity to engage bioinformaticians from around the world, and notably, to begin discussions with leading scientists from Asia.

The Genomic Standards Consortium (GSC) is organizing the M3 workshops as part of its goal to create richer descriptions of our collection of genomes and metagenomes through the development of standards and tools for supporting compliance and exchange of contextual information [6]. Established in September 2005, this international community includes representatives from the International Nucleotide Sequence Database Collaboration (INSDC), major genome sequencing centers, bioinformatics centers and a range of research institutions.

The GSC has been responsible for promulgating the MIGS/MIMS standard (Minimal Information about Genomic/Metagenomics Sequences), and, at the latest GSC meeting in September 2009, a new standard MIENS (Minimal Information about Environment Sequences). These standards are being incorporated into the INSDC (International Nucleotide Sequence Database Collaboration) as part of a new “structured comment field”. This development will be explored in a panel session that will be part of the workshop, involving representatives from EBI, Genbank and DDBJ. The GSC has also launched a new electronic journal SIGS (Standards in Genomic Sciences (http://standardsingenomics.org/) in order to provide an open-access publication for the rapid dissemination of both genome and metagenome reports compliant with the MIGS/MIMS standards; the first issue (July 2009) contains reports on seven sequenced bacterial genomes.

2. M3 Workshop Structure

The workshop goal is to attract experimentalists and computational researchers making "next-generation" use of contextual metadata. The workshop is divided into two parts – a set of contributed talks that highlight specific research activities, and a panel of leaders in the metagenomics community who will discuss the broad issues related to generation of metagenomics data, metadata standards and tools to support the metaanalysis. In addition, the workshop includes a poster session to highlight recent advances related to the M3 goals and GSC activities.

The contributed talks describe comparative metagenomic studies that demonstrate the power provided by data curated (e.g., habitat or host) and measured (e.g. geographic location, salinity, temperature, or pH) using appropriate metadata standards. Likewise, we have welcomed studies describing new approaches, tools, databases, standards, ontologies or substantial new sets of curated metadata that aid in the integration and inter-operability of disparate datasets. We have also included discussion of research focused on capture and organization of metadata, for example through text-mining and ontology development, that enables new understanding of the interaction of organisms in their ecological context.

The talks will cover the three "M"s:

Metagenomics

- *Using 100 years of data to contextualize metagenomics in the Western English Channel*. Jack Gilbert, Plymouth Marine Laboratory, UK
- *Metagenomics reveals functional shifts in the bovine rumen microbiota composition with propionate intake.* Michael E. Sparks, Animal and Natural Resources Institute, USDA, Agricultural Research Service, Beltsville, USA

Metadata

- *Gemina: Ontology and Metadata Standards Development provide core of Infectious Pathogen Surveillance and Geospatial Tool.* Lynn Schriml, University of Maryland School of Medicine, Baltimore, USA

MetaAnalysis

- *Comparative Microbial Genomics of resistance genes in Staphylococcus aureus*. Anja Stausgaard, The Technical University of Denmark, 2800 Kgs. Lyngby, Denmark
- *More accurate taxonomic assignment of short reads*. Gabriel Valiente, Algorithms, Bioinformatics, Complexity and Formal Methods Research Group, Technical University of Catalonia, E-08034 Barcelona, Spain

The panel will include GSC board members and metagenomic data producers, organizers from the main (meta)genomic databases, and tool producers. The central theme for discussion is "unifying access to our current collection of genomes and metagenomes."

Acknowledgments

The organizers gratefully acknowledge the support from the NSF grant RCN4GSC, DBI-0840989. LH has also been supported in part by NSF IIS 0844419: SGER for Utility and Usability of Text Mining for Biological Curation. PS is supported by NERC grant (NE/E007325/1) to DF.

References

1. K. Liolios, K. Mavromatis, N. Tavernarakis and N. C. Kyrpides, Nucleic Acids Res **36**, D475-479 (2008).
2. D. B. Rusch, A. L. Halpern, G. Sutton, K. B. Heidelberg, S. Williamson, S. Yooseph, D. Wu, J. A. Eisen, J. M. Hoffman, K. Remington, K. Beeson, B. Tran, H. Smith, H. Baden-Tillson, C. Stewart, J. Thorpe, J. Freeman, C. Andrews-Pfannkoch, J. E. Venter, K. Li, S. Kravitz, J. F. Heidelberg, T. Utterback, Y. H. Rogers, L. I. Falcon,

4
V. Souza, G. Bonilla-Rosso, L. E. Eguiarte, D. M. Karl, S. Sathyendranath, T. Platt, E. Bermingham, V. Gallardo, G. Tamayo-Castillo, M. R. Ferrari, R. L. Strausberg, K. Nealson, R. Friedman, M. Frazier and J. C. Venter, PLoS Biol **5** (3), e77 (2007).

3. F. Meyer, D. Paarmann, M. D'Souza, R. Olson, E. M. Glass, M. Kubal, T. Paczian, A. Rodriguez, R. Stevens, A. Wilke, J. Wilkening and R. A. Edwards, BMC Bioinformatics **9**, 386 (2008).
4. D. Field, G. Garrity, T. Gray, N. Morrison, J. Selengut, P. Sterk, T. Tatusova, N. Thomson, M. J. Allen, S. V. Angiuoli, M. Ashburner, N. Axelrod, S. Baldauf, S. Ballard, J. Boore, G. Cochrane, J. Cole, P. Dawyndt, P. De Vos, C. DePamphilis, R. Edwards, N. Faruque, R. Feldman, J. Gilbert, P. Gilna, F. O. Glockner, P. Goldstein, R. Guralnick, D. Haft, D. Hancock, H. Hermjakob, C. Hertz-Fowler, P. Hugenholtz, I. Joint, L. Kagan, M. Kane, J. Kennedy, G. Kowalchuk, R. Kottmann, E. Kolker, S. Kravitz, N. Kyrpides, J. Leebens-Mack, S. E. Lewis, K. Li, A. L. Lister, P. Lord, N. Maltsev, V. Markowitz, J. Martiny, B. Methe, I. Mizrachi, R. Moxon, K. Nelson, J. Parkhill, L. Proctor, O. White, S. A. Sansone, A. Spiers, R. Stevens, P. Swift, C. Taylor, Y. Tateno, A. Tett, S. Turner, D. Ussery, B. Vaughan, N. Ward, T. Whetzel, I. San Gil, G. Wilson and A. Wipat, Nat Biotechnol **26** (5), 541-547 (2008). J. Raes, K. U. Foerstner and P. Bork, Curr Opin Microbiol **10** (5), 490-498 (2007).
5. J. Raes, K. U. Foerstner and P. Bork, Curr Opin Microbiol **10** (5), 490-498 (2007).
6. D. Field, G. M. Garrity, S. A. Sansone, P. Sterk, T. Gray, N. Kyrpides, L. Hirschman, F. O. Glockner, R. Kottmann, S. Angiuoli, O. White, P. Dawyndt, N. Thomson, I. S. Gil, N. Morrison, T. Tatusova, I. Mizrachi, R. Vaughan, G. Cochrane, L. Kagan, S. Murphy and L. Schriml, OMICS **12** (2), 109-113 (2008).

EXTRACTION OF GENOTYPE-PHENOTYPE-DRUG RELATIONSHIPS FROM TEXT: FROM ENTITY RECOGNITION TO BIOINFORMATICS APPLICATION

ADRIEN COULET[1,2], NIGAM SHAH[2], LAWRENCE HUNTER[4], CHITTA BARRAL[5], RUSS B. ALTMAN[1,3]

1.Department of Genetics, 2.Department of Medicine, 3.Department of Bioengineering, Stanford University, Stanford, CA 94305, USA

4.Center for Computational Pharmacology, University of Colorado Denver School of Medicine, Aurora, CO 80045, USA

5. Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287, USA

E-mail: coulet@stanford.edu

Advances in concept recognition and natural language parsing have led to the development of various tools that enable the identification of biomedical entities and relationships between them in text. The aim of the *Genotype-Phenotype-Drug Relationship Extraction from Text* workshop (or *GPD-Rx* workshop) is to examine the current state of art and discuss the next steps for making the extraction of relationships between biomedical entities integral to the curation and knowledge management workflow in Pharmacogenomics. The workshop will focus particularly on the extraction of Genotype-Phenotype, Genotype-Drug, and Phenotype-Drug relationships that are of interest to Pharmacogenomics. Extracting and structuring such text-mined relationships is a key to support the evaluation and the validation of multiple hypotheses that emerge from high throughput translational studies spanning multiple measurement modalities. In order to advance this agenda, it is essential that existing relationship extraction methods be compared to one another and that a community wide benchmark corpus emerges; against which future methods can be compared. The workshop aims to bring together researchers working on the automatic or semi-automatic extraction of relationships between biomedical entities from research literature in order to identify the key groups interested in creating such a benchmark.

Keywords: NLP; Pharmacogenomics; Entity Recognition; Event Extraction; Genotype-Phenotype-Drug Relationships

1. Introduction

Research in the BioNLP community, such as BioCreative II[1] and the BioNLP Shared Task'09,[2] have led to the development of efficient BioNLP methods for entity recognition and event extraction. The aim of the *GPD-Rx* workshop is to discuss how results of previous shared tasks can be adapted and improved in order to efficiently provide a detailed representation of complex pharmacogenomic processes described in the literature. The extraction of a structured and fine-grained representation is a key to evaluate and validate hypothesis that emerge from translational studies. The objective of the *GPD-Rx* workshop is thus to advance in this direction by identifying key groups and propose corpus, standard vocabularies, knowledge representation language and evaluation methods that would enable the comparison and the interoperability of future results.

2. Entity Recognition

Entity recognition or named entity recognition is the task of identifying, in free text, words that mention a known entity. Most of the efforts aimed at extracting relationships between entities start with this fundamental task in order to identify entities to be related.

Entity recognition has been extensively studied in the biomedical domain with varying results. Some of the proposed methods are generic and can identify any kind of entity that is part of a dictionary provided as a reference to the system.[3,4] Other methods are specialized in the recognition of specific kinds of entities such as genes/proteins,[5] genomic variations,[6,7] diseases,[8] or drugs.[9] Machine learning approaches are commonly integrated with entity recognition methods to improve their results.[10]

The first goal of the *GPD-Rx* workshop is to discuss issues in the recognition of entities relevant to pharmacogenomics.

3. Extraction of Relationships between Entities

The second goal of the workshop is to discuss the application, in pharmacogenomics, of methods that extract relationships between relevant entities (*e.g.* genomic variation, phenotype, drug).

One simple approach is based on the hypothesis that two entities which are frequently mentioned together are associated. Entity recognition methods have been applied to search for the co-occurrences of entities with the goal of discovering associated ones.[11] This approach has been applied for the construction of gene networks[12] or the guidance of biomedical curation.[13] In such co-occurance driven approaches, associations have a higher chance to be true when the co-occurrence of entities is observed in a small amount of text (*e.g.* a sentence), and a lower chance to be true when observed in larger amounts (*e.g.* a full section).

The development of natural language parsers have led to a second approach that enables, by providing the grammatical structure of sentences, the extraction of relationships (or events) mentioned in the text. The importance of learning protein-protein interactions in biology has motivated many researchers to use parsers to extract such relations with a high accuracy. The work of Fundel *et al.*,[14] of Rebholz-Schuhmann *et al.*,[15] of Hunter *et al.*,[16] and of Miyao *et al.*[17] illustrate the latest research in extracting biomedical relationships from text.

Similar approaches have already been developed for the extraction of Genotype-Phenotype-Drug relationships.[18–21] The *GPD-Rx* workshop aims at identifying issues specific to this task and to using the output of such efforts. For example, the comparison of extracted relationships, to determine agreement or to point out a contradiction, is a key to make extracted relationships actionable.

4. Standards

We belive that BioNLP groups focused on relationship extraction tasks would have a mutual interest in using shared standards to facilitate the comparison and the interoperability of their results. The main ones are:

- the use of unique identifier for entities involved in relationships,
- the use of a common knowledge representation language for the description of relationships,
- evaluation methods for the extraction of relationships,
- shared text corpora and vocabularies of entity names and vocabularies of relationship type,
- set of gold standard relationships.

The workshop aims to stimulate discussion for identifying, sharing and wide-spread use of such standards when applying text-mining in the realm of pharmacogenomics.

Acknowledgments

We would like to thank the PSB 2010 organizers and particularly Tiffany Murray for helping us in the organization of the *GPD-Rx* workshop.

References

1. Lynette Hirschman, Martin Krallinger, John Wilbur and Alfonso Valencia, Editors. 'The BioCreative II - Critical Assessment for Information Extraction in Biology Challenge', *Genome Biology*, **9**(S2), (2008).
2. Jun'ichi Tsujii, Editor. *Proceedings of the BioNLP 2009 Workshop Companion Volume for Shared Task*, (2009).
3. Alan R. Aronson, 'Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program'. *Proceedings of the AMIA Symposium*, pp. 17–21, (2001).
4. Manhong Dai, Nigam H. Shah, Wei Xuan, Mark A. Musen, Stanley J. Watson, Brian D. Athey and Fan Meng, 'An Efficient Solution for Mapping Free Text to Ontology Terms', *Proceedings of the AMIA Summit on Translational Bioinformatics*, (2008).
5. Larry Smith, Lorraine K Tanabe, Rie Johnson nee Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, *et al.*, 'Overview of BioCreative II gene mention recognition', *Genome Biology*, **9**(S2), (2008).
6. J. Gregory Caporaso, William A. Baumgartner Jr., David A. Randolph, K. Bretonnel Cohen and Lawrence Hunter, 'MutationFinder: a high-performance system for extracting point mutation mentions from text', *Bioinformatics*, **23**(14), pp. 1862–1865, (2007).

7. Christopher J.O. Baker and Dietrich Rebholz-Schuhmann, Editors. 'Proceedings of the European Conference on Computational Biology (ECCB) 2008 Workshop: Annotations, interpretation and management of mutations (AIMM)', *Bioinformatics*, **10**(S8), (2009).
8. Rong Xu, Kaustubh Supekar, Alex Morgan, Amar Das and Alan M. Garber, 'Unsupervised Method for Automatic Construction of a Disease Dictionary from a Large Free Text Collection', *Proceedings of the AMIA Symposium*, (2008).
9. Isabel Segura-Bedmar, Paloma Martnez and Mara Segura-Bedmar, 'Drug name recognition and classification in biomedical texts: A case study outlining approaches underpinning automated systems', *Drug Discovery Today*,**13**(17-18), pp. 816–823 (2008).
10. Robert Leaman and Graciela Gonzalez, 'Banner: An executable survey of advances in biomedical named entity recognition', in *Pacific Symposium on Biocomputing*, pp. 652–663, (2008).
11. Graciela Gonzalez, Juan C. Uribe, Luis Tari, Colleen Brophy and Chitta Baral, 'Mining gene-disease relationships from biomedical literature: Weighting proteinprotein interactions and connectivity', in *Pacific Symposium on Biocomputing*, pp. 28–39, (2007).
12. Tor-Kristian Jenssen, Astrid Laegreid, Jan Komorowski and Eivind Hovig, 'A literature network of human genes for high-throughput analysis of gene expression', *Nature Genetics*, **28**(1), pp. 21–8, (2001).
13. Yael Garten and Russ B. Altman, 'Pharmspresso: a text mining tool for extraction of pharmacogenomic concepts and relationships from full text', *BMC Bioinformatics*, **10**(Suppl 2), S6, (2009).
14. Katrin Fundel, Robert Küffner and Ralf Zimmer, 'Relex - relation extraction using dependency parse trees', *Bioinformatics*, **23**(3), 365–371, (2007).
15. Dietrich Rebholz-Schuhmann, Harald Kirsch, Miguel Arregui, Sylvain Gaudan, Mark Riethoven and Peter Stoehr, 'Ebimed - text crunching to gather facts for proteins from medline', *Bioinformatics*, **23**(2), 237–244, (2007).
16. Lawrence Hunter, Zhiyong Lu, James Firby, William A. Baumgartner Jr, Helen L. Johnson, Philip V. Ogren and K. Bretonnel Cohen, 'Opendmap: An open-source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-type-specific gene expression', *BMC Bioinformatics*, **9**(78), (2009).
17. Yusuke Miyao, Kenji Sagae, Rune Saetre, Takuya Matsuzaki and Jun'ichi Tsujii, 'Evaluating contributions of natural language parsers to proteinprotein interaction extraction', *Bioinformatics*, **25**(3), 394–400, (2009).
18. Thomas C. Rindflesch, Lorraine Tanabe, John N. Weinstein and Lawrence Hunter, ' EDGAR: extraction of drugs, genes and relations from the biomedical literature', in *Pacific Symposium on Biocomputing*, pp. 517-528, (2000).
19. Caroline B. Ahlers, Marcelo Fiszman, Dina Demner-Fushman, François-Michel Lang and Thomas C. Rindflesch, 'Extracting semantic predications from medline citations for pharmacogenomics', in *Pacific Symposium on Biocomputing*, pp. 209–220, (2007).
20. Luis Tari, Jörg Hakenberg, Graciela Gonzalez and Chitta Baral, 'Querying parse tree database of medline text to synthesize user-specific biomolecular networks', in *Pacific Symposium on Biocomputing*, pp. 87–98, (2009).
21. Luis Tari, Saadat Anwar, Shanshan Liang, Jörg Hakenberg and Chitta Baral 'Synthesis of Pharmacokinetic Pathways Through Knowledge Acquisition and Automated Reasoning', in *Pacific Symposium on Biocomputing*, (2010).