FungiRegEx: A tool for patterns identification in Fungal Proteomic sequences using regular expressions

Victor Terron-Macias
Centro de Investigación en Matemáticas CIMAT, A.C.
Jezreel Mejía-Miranda
Centro de Investigación en Matemáticas CIMAT, A.C.
Miguel Canseco-Pérez
mcansec@ia.upchiapas.edu.mx
Universidad Politécnica de Chiapas
Mirna Muñoz-Mata
Centro de Investigación en Matemáticas CIMAT, A.C.
Miguel Terron-Hernández
Universidad Tecnológica de Tlaxcala

Article

Keywords:

Posted Date: January 18th, 2024

DOI: https://doi.org/10.21203/rs.3.rs-3852782/v1

Additional Declarations: No competing interests reported.

Abstract

In the context of genome-scale research, it is imperative to automatically analyze numerous species and sub-species to discern distinctive features present in multiple proteomes that contain specific sequences of interest since they provide specific properties. Complex sequences must be recognized within an organism's complete set of proteomes to accomplish this. This study introduces FungiRegEx, a user-friendly software for automatic genome-scale proteome analysis of fungi organisms, addressing the limitations of existing tools.

FungiRegEx utilizes real-time data retrieval of the different species from the JGI Mycocosm database without downloading any files. With a user-friendly GUI, the tool offers efficient regular expression searches across 2,402 fungal species from the JGI Mycocosm portal. Validation with the sequence AXSXG or effector RXRL demonstrates FungiRegEx's effectiveness in identifying user-defined patterns in the retrieved sequences. FungiRegEx accelerates result retrieval compared to manual processes, providing a console-free and programming-free experience; this tool allows customization, result filtering, and the possibility of saving the results for future research.

FungiRegEx offers a promising solution for researchers exploring specific sequences in the fungal proteomes. It combines speed, adaptability, and ease of use, displaying the results in a GUI and making it easy to read. Its architecture ensures optimized resource usage and deployment flexibility, allowing the customization of specific software parameters.

The tool's potential for future research and exploration is emphasized, providing a nuanced perspective on its practical use within the fungal genomics community.

Background

Understanding the characteristics of the species of the fungi phylogenetic tree requires the identification of certain sequences in the proteomes, which in turn may be correlated with the environment and its conditions[1]. The phylogenetic analysis provides a framework to develop research and identify multiple similarities and conservation zones. To identify and characterize proteins, a detailed analysis of proteomes is essential. Human experts can perform this task, but the analysis is challenging on a large scale.

A critical aspect of such automatic processes is the efficient traversal of proteomes. In the biological field, hard-coded algorithms mostly traverse phylogenetic trees, and some resources, such as grep[2], which is a text-processing program designed for regular pattern matching within the text, allow the search of regular expressions in a string; this string can be a proteome or any other type of text sequence; some requisites of grep are:

1. If the user runs Linux as the Operating System, this tool is included; if not, the user must acquire a similar tool for the respective Operating System that performs the same functions as grep.

2. Knowledge about how to use the bash to execute this tool from a terminal.
3. Knowledge about how to use the wildcards and syntax to perform the search. Note: This program has no GUI to display the results.

Another resource to find regular expressions is msgfdb2pepxml[3]; this resource is a library that converts the output from the MS-GFDB search engine to pepXML, uses regular expressions to recognize enzyme uses and cleavage rules, and supports PSI-MS, in order to execute this library requires:

1. Knowledge in Python programming language.
2. Knowledge of the syntax of the library to use it correctly.

Note

As this is a library, we do not have a GUI to present the results directly.

Another resource to find regular expressions is PhyloPattern[4], which is a library focused on identifying regular expressions in phylogenetic trees; this library is not focused on proteomes or any other biological sequence, also to execute this tool requires:

1. Prolog syntax knowledge.
2. Install in Prolog engine the library PhyloPattern.
3. Knowledge in Prolog programming language.

Note

knowledge another critical aspect is that this library does not provide a GUI directly.

Another resource is PatScan[5]; which is a program focused on searches for protein or nucleotide sequences of a pattern (regular expression). In order to execute this program the requisites are:

1. Compile the source files.
2. A FASTA file to perform the search on.
3. Knowledge in terminal use and syntax.

Another resource to identify repeats using regular expressions is Patscan[5] and PatMatch[6]. These programs do not automate searching for patterns within the sequences because they require the user to write the complete sequence in which they want to search for the pattern; entering all the sequences of a genome or proteome can take a long time for the size and amount of elements. PatMatch requires:

1. Access to this tool through their website.
2. Knowledge about the pattern syntax to perform the search.

Notes: [1] Focuses only on peptide and nucleotide sequences. [2] The length limitation to search for is less than 20 residues. [3] Only can process one sequence by time.

The process of searching for regular expressions is notably time-intensive if the sequences are introduced to any software manually.

However, the programs and libraries already mentioned could require specific knowledge, like commands, bash, computer, download of files, or programming knowledge; this seriously complicates the process of searching for regular expressions on a large scale for users without proficiency in using programs or libraries that help analyze biological sequences. Tools like those already mentioned above are examples of software that offers a simple pattern-matching system, which may or may not include a GUI, and its implementation could be challenging. Also, the information source needed to function must be trustworthy.

Fortunately, the Joint Genome Institute (JGI)[7] offers biological sequences like DNA or Protein/Nucleotide sequences with the certainty that all the information has been validated and is trustworthy.

In that context, software that can be easily integrated into automatic genome-scale processes to read and analyze proteomes on a large scale, detect matches, and save considerable time without downloading files is now needed.

In this way, we present FungiRegEx, a software that takes the available information from the JGI Mycocosm portal and performs a search into the proteome databases of the multiple species with the user-defined regular expression through its web scraper module integrated into the tool; also, it integrates a GUI with a user-friendly interface, to use it it is not necessary for the user to install any additional components, download additional files or have solid programming knowledge in any programming language.

This software helps to the recognition of repeated sequences, which holds substantial significance, as it offers valuable insights into the functional and evolutionary roles of diverse organisms[8,9], driving evolution, inducing variation, and regulating gene expression[10]. These patterns can be important for identifying certain protein functions or key structural regions. For

example, searching for protein sequences containing a specific pattern can help identify proteins that bind to certain ligands or have specific enzymatic activity[11]. Searching for repetitive patterns in protein sequences can also help to identify evolutionarily related proteins, which can provide information about the evolution of proteins and their functions over time[12].

Numerous software applications are accessible for the identification of various types of repeats. Nonetheless, no one has focused on FUNGI pattern detection through web scrapping; FungiRegEx does it. FungiRegEx employs a straightforward sequential search method to identify regular expressions directly from the protein sequences of FUNGI organisms. Diverging from the prevalent approach of employing a suffix tree or alignment matrix as a primary data structure, the algorithm introduced in this paper operates by directly identifying regular expressions within the protein sequence. As a result, this methodology exhibits efficiency in memory usage due to launching and running the scraper instances, boasts

enhanced comprehensibility and ease of implementation, and offers great speed in getting multiple sequences at the same time compared to if the process were carried out manually or using tools like PatMatch[6] that requires to introduce one by one sequence. Also, FungiRegEx does not require downloading any fasta or file.

Another relevant aspect of the tool is that it includes a GUI, which means the user does not need to have a strong knowledge of any programming language or commands to use it, compared to if grep or another tool that requires a bash interface were used. Also, the scrapper module is customizable to adapt it to the resources of the computer or server where it is executed (in case the user wants a greater or lesser number of scraper instances). Finally, this tool could be deployed on a server or a computer if the user wants to.

Various tools and resources, such as grep, msgfdb2pepxml, PhyloPattern, and PatScan, exist for searching regular expressions in biological sequences. However, these tools often require specific knowledge, limiting accessibility for users without programming proficiency. The process of manually introducing sequences to these tools is time-intensive and complex.

Addressing these challenges, FungiRegEx is introduced as a user-friendly software designed for automatic genome-scale proteome analysis. Integrated with a web scraper module, FungiRegEx efficiently searches user-defined regular expressions in the proteome databases of multiple fungal species sourced from the JGI Mycocosm database. Notably, FungiRegEx stands out by providing a GUI, eliminating the need for additional downloads or programming knowledge.

Materials and methods

Architecture of FungiRegEx

FungiRegEx front-end is based on React JS 17.0.2v, is a JavaScript library that is both available and open-source, designed for constructing interfaces[13], and Node JS 16.17v, serving as the back-end, is a JavaScript runtime constructed upon the V8 JavaScript engine[14], Chromium version 79.0.3945.117, an open-source browser project, is dedicated to creating a more secure, expeditious, and dependable means for users to engage with the web[15]. A collection of React JS components was created to execute these functions: provide an interactive GUI (see Fig. 1) to choose specific parameters to perform a search through the NodeJS back-end with the user-defined Regular Expression into proteomes, pick the species or species into fungi organisms to perform the search, visualize the results of the search, and download the results in CSV format if the user wants to, also the Node JS back-end was designed to perform these tasks: launch the scrapper instances into the JGI Mycocosm database, with the obtained information of every instance the regular expression is looked for into the proteome, to save RAM memory the backend reuses each instance of Chromium once they have obtained the information, in case of error an automatic restart of the instance is performed.

FungiRegEx works as follows: first, the user selects the type of search he wants to perform globally into all 2,402 different species of fungi or a specific species (**This means that new taxonomic additions will not be available in the software, unless that the user add it.**). Second, the user selects to scrap the regular expression into a range of identifiers or a list of identifiers in the database. Third, to start the search, the retrieved sequences of proteomes are scrapped into the Joint Genome Institute through the Node JS script and displayed in the table through React JS; the table can be ordered through alphabetical order of Specie or the number of matches into the proteome in ascending or descending order, the last column of the table displays the coincidences. Fourth, the results of the table can be downloaded in an output file in CSV format.

FungiRegEx is distributed as a compressed file in ZIP format. The source code is available for download at SourceForge and https://github.com/Maigolinox/fungiregex GitHub repository. Once the FungiRegEx directory has been uncompressed, it shows the FungiRegEx directory in which the results.csv file is empty, and package.json contains all the instructions to execute the web application correctly. The user must download each prerequisite from the platforms indicated in Table 1 and the documentation and uncompress and install them for the correct execution of FungiRegEx. The command to execute the front-end of FungiRegEx is *npm run start:frontend*, and the command to execute the back-end of FungiRegEx is *npm run start:backend*. Also, you can find more instructions in the documentation according to your operating system.

Validation of FungiRegEx results

FungiRegEx was challenged by querying multiple databases of various species available on the JGI Mycocosm website., the results were validated with the search of the sequence AXSXG as a regular expression, which is a pentapeptide of a lipase group that brings thermostability and resistance to solvents of an enzyme[16] that has been little described in fungi.

Similarly, the results that show FungiRegEx have been validated with the next considerations:

1. Specie: Saccharomyces cerevisiae (SacceM3836_1)[17–20].
2. AXSXG pentapeptide where X represents whatever amino acid.

To bring some results, we perform the search with the next parameters:

1. A.S.G, where . in regular expressions notation means whatever amino acid.
2. Search in a specific range: 1 to 2000. This means that FungiRegEx will launch the scrapper instances to retrieve the data in the JGI Mycocosm database from 2000 proteomes.

After running the search, the tool showed that of the 2000 scraped sequences, only one with identifier 1434 has that pentapeptide only once, while it also identified matches in 281 sequences with similarity. Figure 2 shows the results of the JGI Mycocosm database and Fig. 3 shows the match with the tool:

As mentioned, the tool also looks into the complete sequence for other similarities according to the regular expression (see Fig. 4). We hide the proteome column for the image size to show the results of FungiRegEx with the mentioned parameters.

In this way, the tool proves to be capable of finding the regular expression in the search proteome determined by the user, which may be of interest for subsequent studies.

A second use case executed to validate FungiRegEx functionality involves the search for effectors. Liping Liu et al.[22] identified different effectors, such as the RXLR, asserting that fungi, oomycetes, and bacteria release small secreted proteins crucial for symbiotic interaction and pathogenicity. Liping Liu investigated various effectors in different species, such as Mg3LysM (Mycosphaerella graminicola LysM), secreted by Mycosphaerella graminicola[23,24]. For this example, the JGI Mycocosm database of Mycosphaerella graminícola v2.0 will be used with the RXLR sequence, where X represents any amino acid. In regular expression language, the regular expression is R.LR.

Figure 5 shows the results of FungiRegEx hiding the proteome column due to the size of the proteomes to show the results in the figure.

As shown in Fig. 5, FungiRegEx can find the effector of interest in the proteome, showing the number of matches and the sequences. In addition to the above, it is identified that the species Mycosphaerella graminicola does indeed have this effector (RXRL), as stated by Liping Liu et al.[22].

Implementation

Regular expressions search for FUNGI organisms is based on finding exact repeats of length **k** along the amino acid chain. The regular expression can be as long as the user wants. If the protein sequence of length **k** is diminutive, the comparative process proceeds with greater expeditiousness. Once the regular expression is found in the protein sequence of the FUNGI organism, it is filtered to eliminate those that do not match.

Searching for regular expression matches

The application back-end begins its search by creating a regular expression object. Regular expressions are patterns used to search for character combinations in text strings. Regular expressions can contain various special characters and modifiers that define the pattern to search for[25].

The magnitude of the search range directly impacts the algorithm's processing time; in this way, smaller ranges are preferred for optimal efficiency.

If matches between regular expression and protein sequence are found

The aforementioned search process identifies the regular expression within the amino acid chain with a length of k, as illustrated in Fig. 3. As the detected repeats do not accurately reflect the true length of the repeating pattern, they need to be expanded to match the actual repeat length. In this paper, the chosen approach involves reading the characters located to the left or right of all repeats and storing them in an array.

If no matches between regular expression and protein sequence are found

If no matches are found, the algorithm will continue the search in another amino acid chain of the proteome. The search algorithm can progress in larger intervals without overlooking any repetitions.

The crucial factor in progressing with larger intervals lies in ensuring that the search algorithm never overlooks any matches.

Processing speed

An approximation of the speed will take the algorithm to process the regular expression given by the next mathematical formula.

$$time = \frac{number\ of\ ids}{number\ of\ \frac{pages}{second}}$$

Reducing the size of search intervals can improve processing speed. However, the algorithm's speed may decrease if the intervals become too small. This is because smaller intervals can cause the program to spend more time launching browser instances than acquiring information and performing the search for the regular expression.

With 200 puppeteer instances, 50,000 results are obtained in approximately 139 minutes, as can be seen in Fig. 7 with the estimated time that indicates the monitor (Depending on the resources of the computer, the computer can consult at least seven pages per second; this means that the 50,000 results can be consulted in only 12 minutes); it also clarifies that this depends on the available computer resources. It should be mentioned that just requesting the JGI Mycocosm database and getting a response on the webpage takes around 6.64 seconds, as shown in Fig. 8.

Results

The FungiRegEx tool was constructed to identify the regular expression in Fungal proteomes, the results are listed:

1. Efficiency in Proteomic Sequence Analysis: FungiRegEx is demonstrated as a tool that streamlines the process of analyzing proteomic sequences of 2,402 species available in the JGI Mycocosm portal.

2. Real-time Data Retrieval without FASTA Downloads: The FungiRegEx scrapper module eliminates the need to download any FASTA files for proteomic analysis. FungiRegEx dynamically requests data from the JGI Mycocosm website, ensuring real-time access to information.
3. Accelerated Results Retrieval: FungiRegEx exhibits a significant increase in result retrieval speed compared to manual proteomic data extraction on the JGI Mycocosm website.
4. Optimized Computational Resource Usage: FungiRegEx effectively utilizes the computational resources available on the user's computer, demonstrating efficiency in resource management.
5. Adaptability and Customization: Users can easily adapt FungiRegEx to run on any computer, allowing customization of the number of scrapper instances the program utilizes.
6. User-Friendly GUI Presentation: The software features a GUI that presents results without requiring users to code or possess specialized knowledge like the use of bash.
7. Platform Independence: FungiRegEx operates seamlessly without needing a specific operating system, providing users with flexibility in execution.
8. Local and Server Deployment Options: The tool can be launched locally or deployed on a server, giving users the autonomy to choose the preferred execution mode.
9. Efficient Result Filtering: FungiRegEx offers features for filtering results, facilitating the identification of specific sequences and enhancing result interpretation.
10. Console-Free User Experience: Users can interpret results without needing a console, enhancing accessibility compared to other tools that may require console interaction.
11. Simplified Search Syntax with User-Defined Parameters: FungiRegEx allows users to input any sequence of interest for searching across the entire proteome or within a specific range. This customization is particularly useful for identifying sequences in specific species, potentially contributing valuable insights for further research.
12. Potential for Future Research: The identification of the user-defined regular expression on certain proteomes allows the user to save those results for further researchs, as we could see in the validation section of this research with the AXSXG example sequence where that sequence was of interest due the properties that it provides. These identified sequences may or may not exhibit specific characteristics, paving the way for future research and exploration.
13. Details of the finding sequence: The tool indicates the number of sequences that match the regular expression and shows the exact match.

In conclusion, FungiRegEx enhances the efficiency of proteomic sequence analysis and provides a user-friendly, adaptable, and customizable tool with advanced features for result interpretation and exploration in subsequent research endeavors.

The programs listed in Table 1 accomplished the FungiRegEx function.

Table 1
Components of FungiRegEx that accomplish the functionality.

Program	Version
NodeJS	16.17.0v
ReactJS	18.0.0v
Chromium	79.0.3945.117v
Nodemon	2.0.20v
Axios	0.27.2v
ExpressJS	4.17.13v
Puppeteer cluster	0.23.0v
Regex Parser	2.2.11v
Cheerio	1.0.0-rc.12v
CORS	2.8.5v
DotEnv	16.0.1v
Flatted	3.2.6v

Discussion

In this section, we delve into a comprehensive discussion of the essential findings and limitations associated with FungiRegEx, a tool designed to streamline the analysis of proteomic sequences from the JGI Mycocosm database. The results highlight the tool's efficiency in handling a vast dataset of 2,402 Fungi species, offering accelerated results retrieval and optimized utilization of computational resources. Additionally, FungiRegEx introduces user-friendly features such as real-time data retrieval, adaptability for customization, and a graphical user interface (GUI), presenting a promising solution for researchers engaged in proteomic analysis. However, as with any technological advancement, the discussion also addresses certain limitations and considerations, providing valuable insights into the practical use of FungiRegEx. From speed considerations to potential IP blocking risks and task limitations, the ensuing dialogue aims to assess these aspects critically, offering researchers a nuanced perspective on the tool's capabilities and areas for potential improvement.

The results are presented in Table 2 and Table 3; the information was divided into two tables due to their size.

Table 2
Advantages and limitations of FungiRegEx.

Advantages	Limitations
FungiRegEx has demonstrated remarkable efficiency in analyzing proteomic sequences across the extensive dataset of 2,402 species on the JGI Mycocosm portal. This efficiency is crucial in accelerating research pro- cesses and enabling broader research.	Users are required to enter the characteristics for newly added Fungi species manually. Failure to do so re- stricts users to the initially registered 2,402 species, potentially limiting the software's applicability to fu- ture taxonomic additions.
Eliminating the need for FASTA file downloads through the FungiRegEx scrapper module signifies a significant advancement. Real-time data retrieval from the JGI Mycocosm website enhances the imme- diacy of access to the latest proteomic information, contributing to the timeliness of research outcomes.	To use the tool and get the available information from the website internet connection is needed, also the time to get the information depends of the user connection, limiting the speed to perform the search.
[1]The observed increase in result retrieval speed with FungiRegEx compared to manual extraction on the JGI Mycocosm website underscores the tool's potential to save researchers valuable time. This acceleration in the data retrieval process could have substantial implications for large-scale studies.	The speed of FungiRegEx is influenced by internet speed, available computational resources, and the num- ber of deployed scrapper instances. Balancing these factors is crucial, as fewer instances reduce resource consumption but extend task completion time. [2] Deploying many scrapper instances (over 100) poses the risk of temporary IP blocking by the JGI My- cocosm website. This limitation necessitates a careful balance to prevent potential issues with site access.
The effective utilization of computational resources by FungiRegEx highlights its efficiency in resource management. This optimization ensures that the tool operates smoothly, providing a seamless user experi- ence.	The resources that FungiRegEx uses must necessarily be configured by the user by changing the parameters of FungiRegEx.
The adaptability of FungiRegEx for use on any com- puter and the ability to customize the number of scrap- per instances enhances its versatility. Users can tailor the tool to suit their specific computational capabilities and requirements.	[1] The resources that FungiRegEx uses must neces- sarily be configured by the user. [2] FungiRegEx currently supports only one task at a time. If multiple users attempt simultaneous searches, the tool prioritizes the latest task, potentially deleting the previous user's task. This limitation may impact concurrent usage scenarios.

Advantages	Limitations
A user-friendly GUI in FungiRegEx simplifies result presentation, making it accessible to users without specialized coding knowledge. This feature promotes ease of use in the scientific community.	The results table of FungiRegEx shows the complete proteome in a single line, making it difficult to read the table in very long proteomes; for this reason, the GUI can be improved to facilitate the reading and presentation of the results.
The platform independence of FungiRegEx, coupled with the choice between local and server deployment, provides users with flexibility. Researchers can choose the execution mode that aligns with their preferences and available infrastructure.	While FungiRegEx offers flexibility in deployment, it is recommended for users to install the application locally. This recommendation emphasizes that Fun- giRegEx is single task.

Table 3
Advantages and limitations of FungiRegEx.

Advantages	Limitations
The tool's provision for user-defined parameters in the search syntax, including the ability to specify se- quences in certain species, enhances its applicability for diverse research scenarios.	The user needs to know how many proteomes to search for, that is, the length of the proteome of the species or species of interest.
As demonstrated in the validation section, identifying and filtering the user-defined regular expression into the results table of FungiRegEx for future investiga- tions highlights the tool's potential for ongoing and future research. This capability allows users to explore specific sequences of interest and their potential char- acteristics inside of the already obtained results.	The user must already have the sequence of interest that they wish to search for in the proteome of a certain species.
FungiRegEx makes it possible to store the results in CSV format.	FungiRegEx does not include an internal database to store the information; this means that if the user does not save the results, all the search of the regular ex- pression inside the scraped proteomes will be lost.
The tool's provision of information on the number of matching sequences and the exact match further enhances result transparency and aids researchers in refining their analyses.	Although the tool locates the sequences matching the regular expression in the proteome when listing the proteome in the results table, the matches do not have a differentiator that facilitates their location, such as changing the font color only in the matching part; this makes it challenging to locate the matching sequences in the proteome.

Declarations

Acknowledgements

None.

Funding

Dr. Miguel Angel Canseco Pérez supported the development of FungiRegEx with personal financial resources.

Availability of data and materials

The data used or analyzed during the current software tool are available from the corresponding author in the Joint Genome Institute database.

Ethics approval and consent to participate

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Consent for publication

Not applicable.

Authors' contributions

VMTM, MACP: Investigation, Web-Application Development, Writing, Resources, Project administration, Formal Analysis, Validation. JMM, MAMM, MTH: Writing —Review & editing

Availability and Requirements

- Project name: FungiRegEx.
- Project home page: https://sourceforge.net/projects/fungiregex/
- Operating system(s): Windows, Linux.
- Programming language: JavaScript.
- License: Creative Commons Attribution Non-Commercial License V2.0.
- Any restrictions to use by non-academics: None.

References

1. Lucia Muggia, K. S., Claudio G. Ametrano & Tesei, D. An overview of genomics, phylogenomics and proteomics approaches in ascomycota. *MDPI Life* **10**, 356, DOI: 10.3390/life10120356 (2020).
2. Bull, R., Trevors, A., Malton, A. & Godfrey, M. Semantic grep: regular expressions + relational abstraction. In *Ninth Working Conference on Reverse Engineering, 2002. Proceedings.*, 267–276, DOI: 10.1109/WCRE.2002.1173084 (2002).
3. Boris Nagaev, K. Y. & Palmblad, M. msgfdb2pepxml (2011).
4. Philippe Gouret, J. D. T. & Pontarotti, P. Phylopattern: regular expressions to identify complex patterns in phylogenetic trees. *BMC Bioinforma.* **10**, 298, DOI: 10.1186/1471-2105-10-298 (2009).
5. Dsouza M, O. R., Larsen N. Searching for patterns in genomic data. *Trends genet* **13**, 497–498, DOI: 10.1016/s0168-9525 (1997).

6. Yan T, B. T. M. L. W. D. W. S. C. J. R. S., Yoo D. Patmatch: a program for finding patterns in peptide and nucleotide sequences. *Nucleic Acids* **13**, 262–266, DOI: 10.1093/nar/gki368 (2005).
7. JGI, J. G. I. About us (2022).
8. Achaz G, N. P. R. E., Coissac E. Associations between inverted repeats and the structural evolution of bacterial genomes. *Genetics* **164**, 1279–1289, DOI: 10.1093/genetics/164.4.1279 (2003).
9. van Belkum A, v. A. L. V. H., Scherer S. Short-sequence dna repeats in prokaryotic genomes. *Microbiol Mol Biol Rev* **62**, 275–293, DOI: 10.1128/MMBR.62.2.275-293.1998 (1998).
10. Xingyu Liao, J. Z. H. L. X. X. B. Z., Wufei Zhu & Gao, X. Repetitive dna sequence detection and its role in the human genome. *Commun. biology* **6**, 954, DOI: 10.1038/s42003-023-05322-y (2023).
11. Daniel Barry Roche, D. A. B. & McGuffin, L. J. Proteins and their interacting partners: An introduction to protein-ligand binding site prediction methods. *Int. J. Mol. Sci.* **16**, DOI: 10.3390/ijms161226202 (2015).
12. Matthew Merski, J. L. J. S. S. D.-H. . M. W. G., Krzysztof Młynarczyk. Self-analysis of repeat proteins reveals evolutionarily conserved patterns. *BMC Bioinforma.* **21**, DOI: 10.1186/s12859-020-3493-y (2020).
13. Meta Platforms, F. O. S. Getting started, what is react and documentation (2020).
14. Foundation, O. Getting started, what is node js and documentation (2020).
15. LLC, G. Getting started, what is chromium and documentation (2020).
16. Denise Esther Gutiérrez-Domínguez, M. M. R.-A. J. N. A. T. I. I.-F. M. C.-P., Bartolomé Chí-Manzanero & Canto-Canché, B. Identification of a novel lipase with ahsmg pentapeptide in hypocreales and glomerellales filamentous fungi. *Int. J. Mol. Sci.* **23**, 9367, DOI: 10.3390/ijms23169367 (2022).
17. Cherry JM, A. C.-B. R. B. G.-C. E. C. K. C. M. D. S. E. S. F. D. H. J. H. B. K. K. K. C. M. S. N. R. P. J. S. M. S. M. W. S. W. E., Hong EL. New data and collaborations at the saccharomyces genome database: updated reference genome, alleles, and the alliance of genome resources. *Genetics* DOI: 10.1093/genetics/iyab224 (2022).
18. Stephen F. Altschul, A. A. S. J. Z. Z. Z. W. M., Thomas L. Madden & Lipman, D. J. Gapped blast and psi-blast: a new generation of protein database search programs, DOI: 10.1093/nar/25.17.3389 (1997).
19. Alejandro A. Schaffer, T. L. M. S. S. J. L. S. Y. I. W. E. V. K., L. Aravind & Altschul, S. F. Improving the accuracy of psi-blast protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res* DOI: 10.1093/nar/29.14.2994 (2001).
20. Edith D Wong, S. A. K. K. R. S. N. M. S. S. S. W. S. R. E. J. M. C., Stuart R Miyasato. Saccharomyces genome database update: server architecture, pan-genome nomenclature, and external resources. *Genetics* DOI: 10.1093/genetics/iyac191 (2023).
21. Steven D. Brown, C. M. J. A. C. A. A. A. S. A. S., Dawn M. Klingeman. Genome sequences of industrially relevant saccharomyces cerevisiae strain m3707, isolated from a sample of distillers

yeast and four haploid derivatives. *ASM Journals - Genome Anouncements* **1**, DOI: 10.1128/genomeA.00323-13 (2013).

22. Liping Liu, Q. J. R. P. R. O. W. Z., Le Xu & Wu, C. Arms race: diverse effector proteins with conserved motifs. *Plant Signal. & Behav.* **14**, 1557008, DOI: 10.1080/15592324.2018.1557008 (2019). PMID: 30621489, https://doi.org/10.1080/15592324.2018.1557008.
23. Marshall, R. *et al.* Analysis of Two in Planta Expressed LysM Effector Homologs from the Fungus Mycosphaerella graminicola Reveals Novel Functional Properties and Varying Contributions to Virulence on Wheat. *Plant Physiol.* **156**, 756–769, DOI: 10.1104/pp.111.176347 (2011).
24. Lee, W.-S., Rudd, J. J., Hammond-Kosack, K. E. & Kanyuka, K. Mycosphaerella graminicola lysm effector-mediated stealth pathogenesis subverts recognition through both cerk1 and cebip homologues in wheat. *Mol. Plant-Microbe Interactions* **27**, 236–243, DOI: 10.1094/MPMI-07-13-0201-R (2014).
25. web docs, M. Regular expressions (2023).

Figures

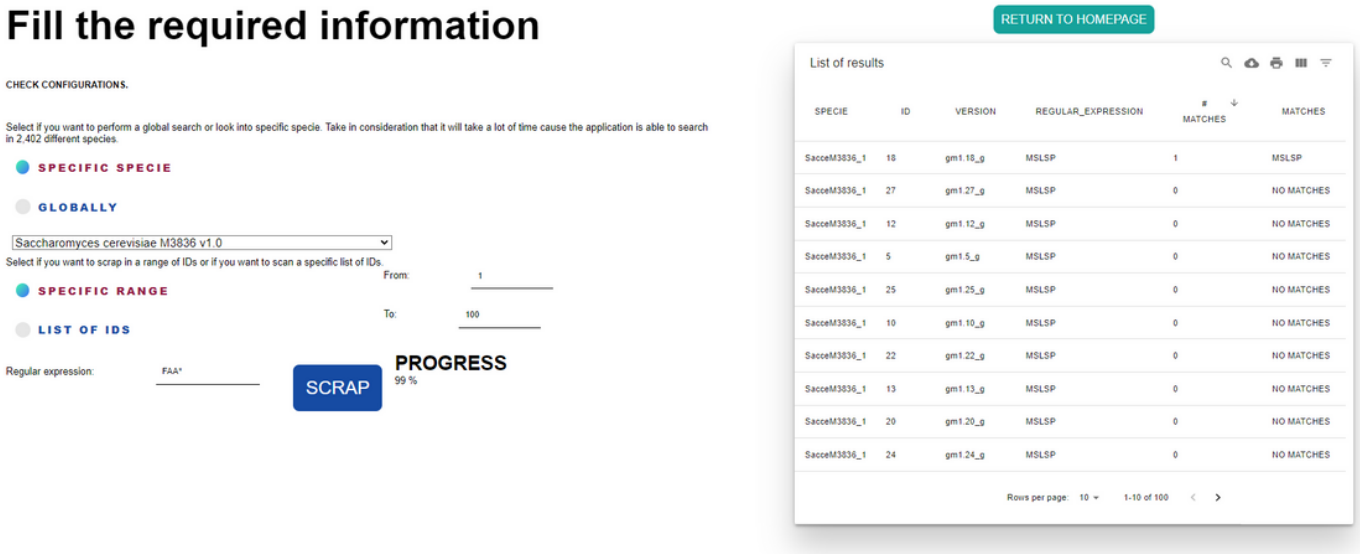

Figure 1

GUI of FungiRegEx.

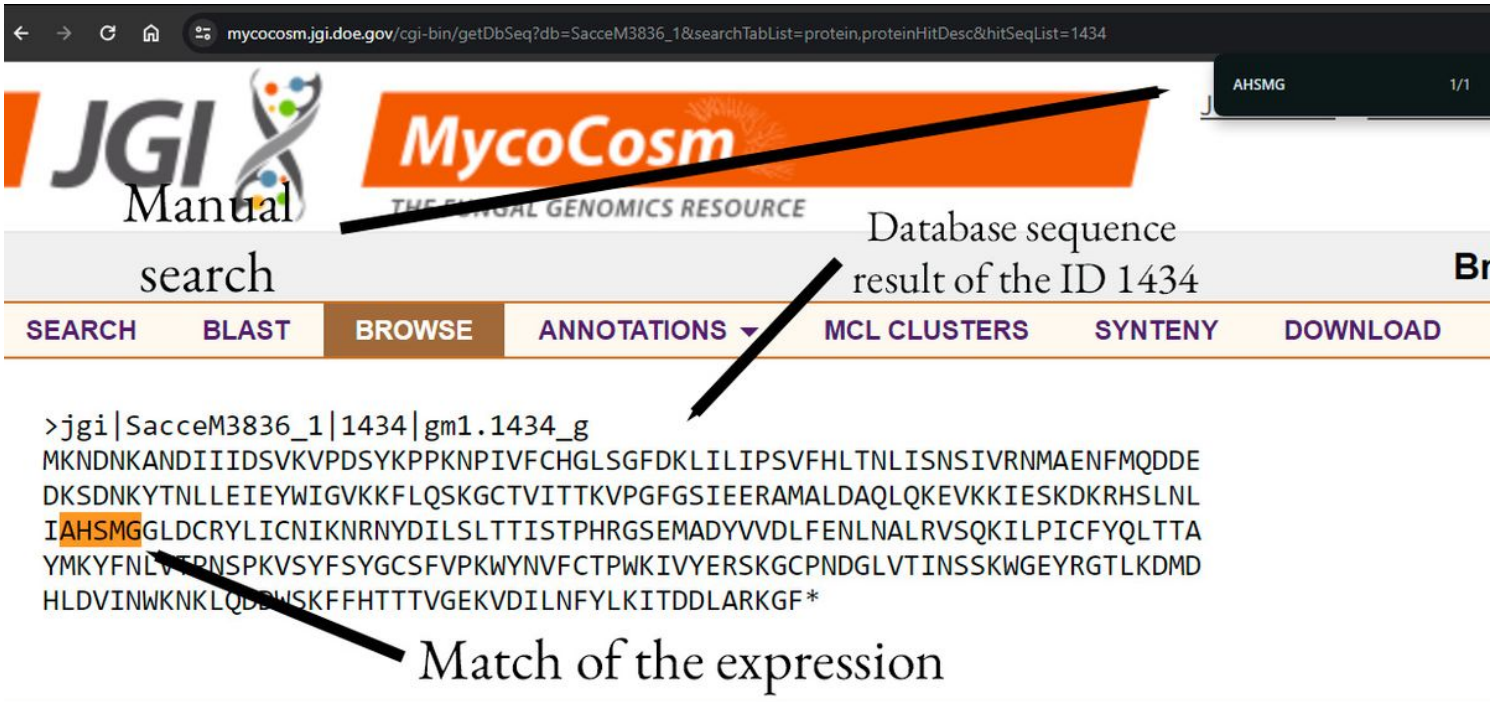

Figure 2

Retrieved sequence of protein Saccharomyces cerevisiae M3836 v1.0. with ID 1434.[21]

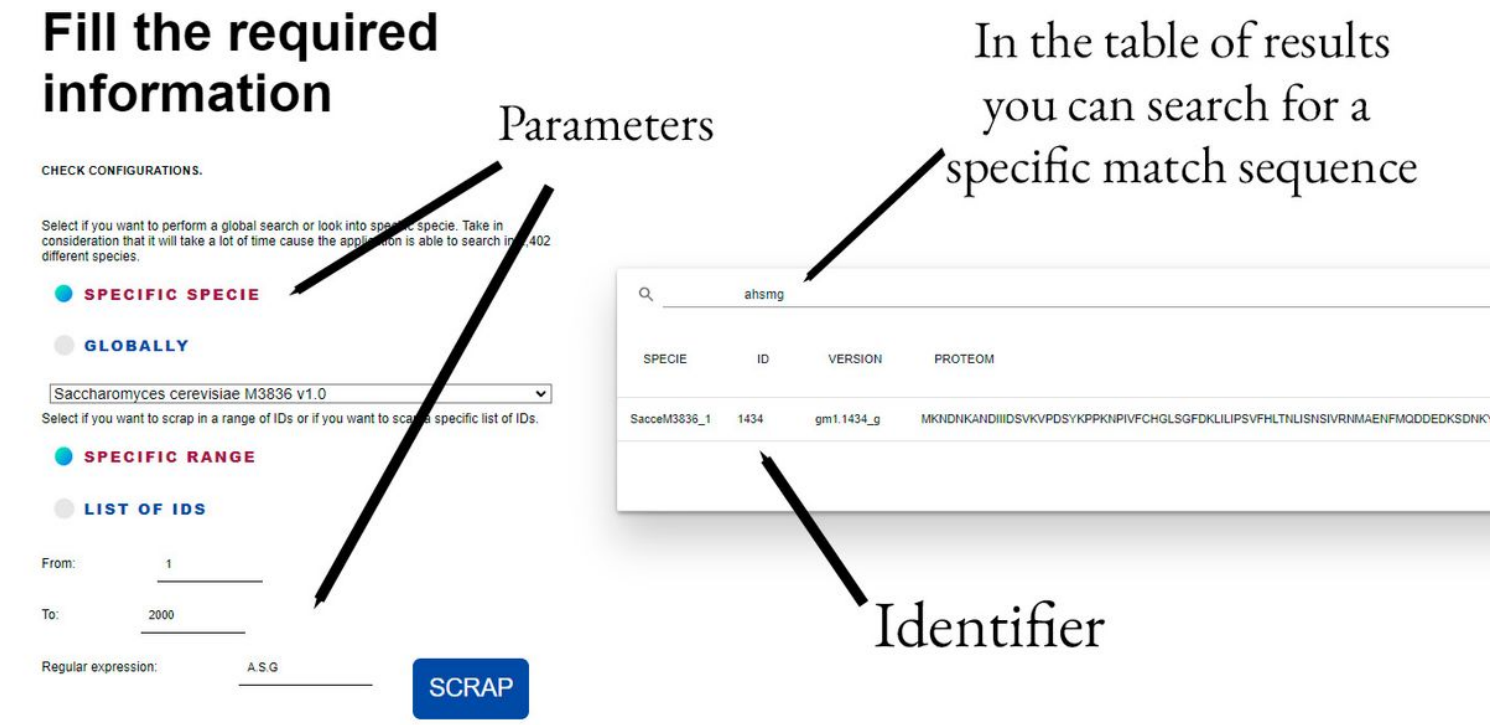

Figure 3

Results from FungiRegEx using A.S.G regular expression filtering by the specific Expression: AHSMG.

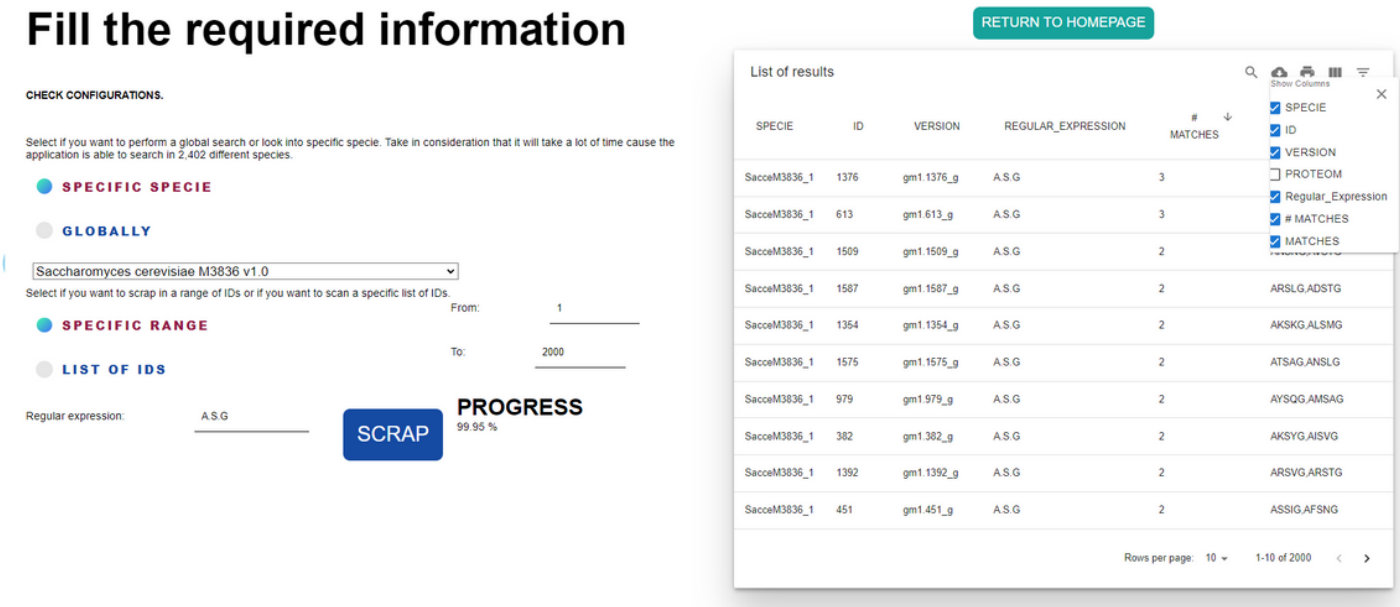

Figure 4

Results from FungiRegEx hiding the proteome column using A.S.G regular expression.

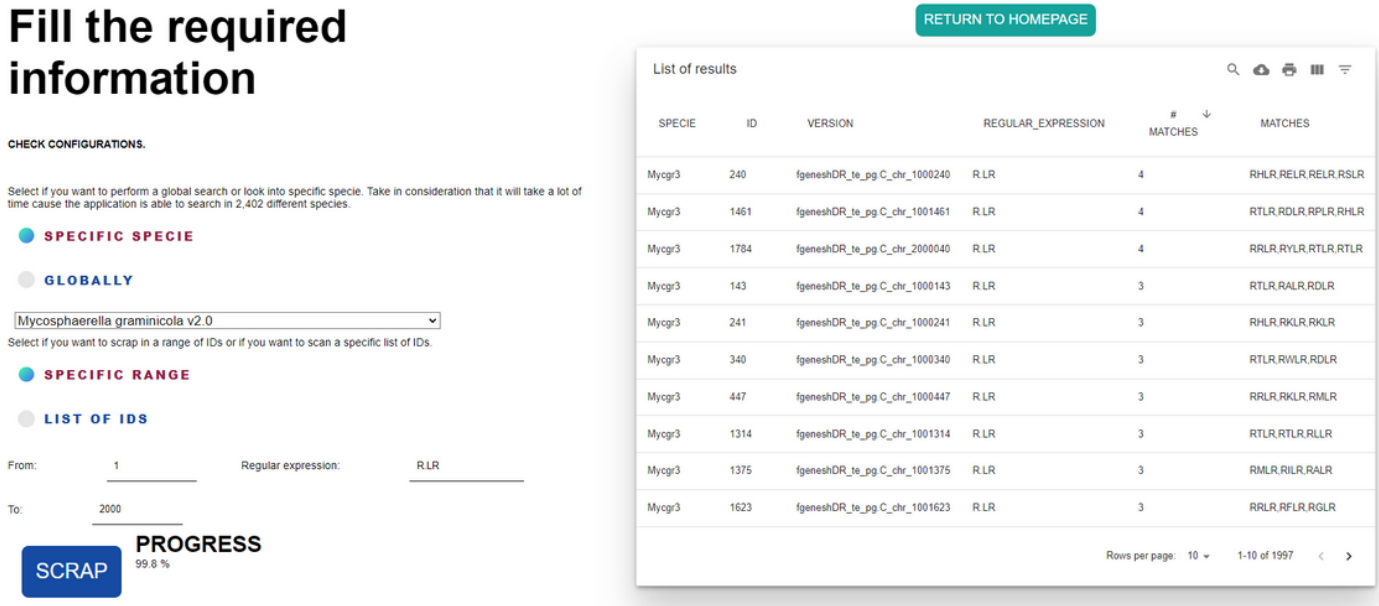

Figure 5

Results from FungiRegEx hiding the proteome column using R.LR effector regular expression.

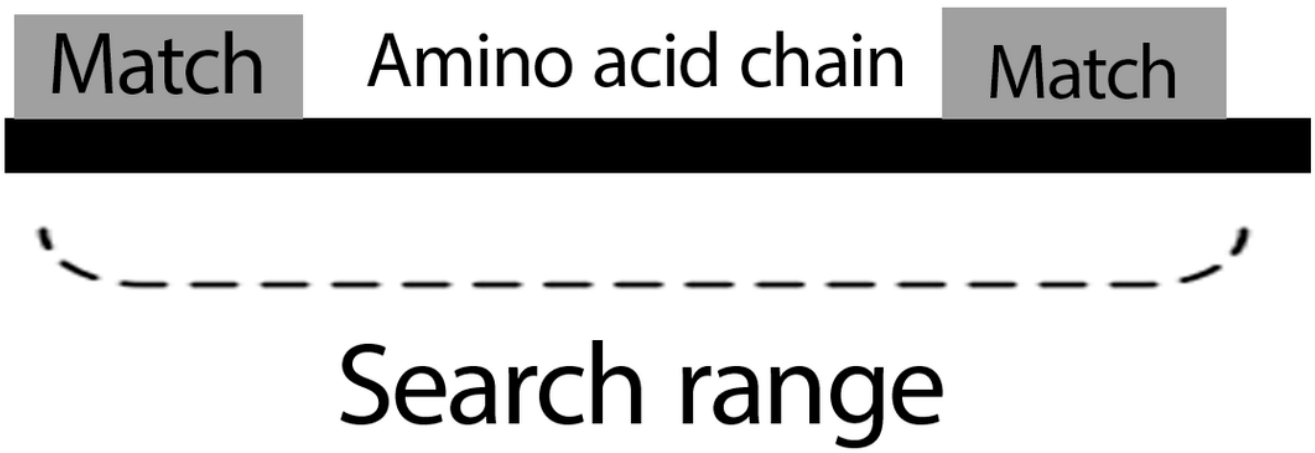

Figure 6

An occurrence of a match. Coincidences in the amino acid chain are detected by reading all and scanning the regular expression for k matches.

```
== Start:     2023-08-11 21:46:11.728
== Now:       2023-08-11 21:50:46.584 (running for 4.6 minutes)
== Progress:  1499 / 50000 (3.00%), errors: 841 (56.10%)
== Remaining: 2.5 hours (@ 5.45 pages/second)
== Sys. load: 26.8% CPU / 47.5% memory
== Workers:   200
```

Figure 7

Progress monitor and calculation of processing time. The Puppeteer cluster includes a tool that monitors the progress of data acquisition and the performance of each instance.

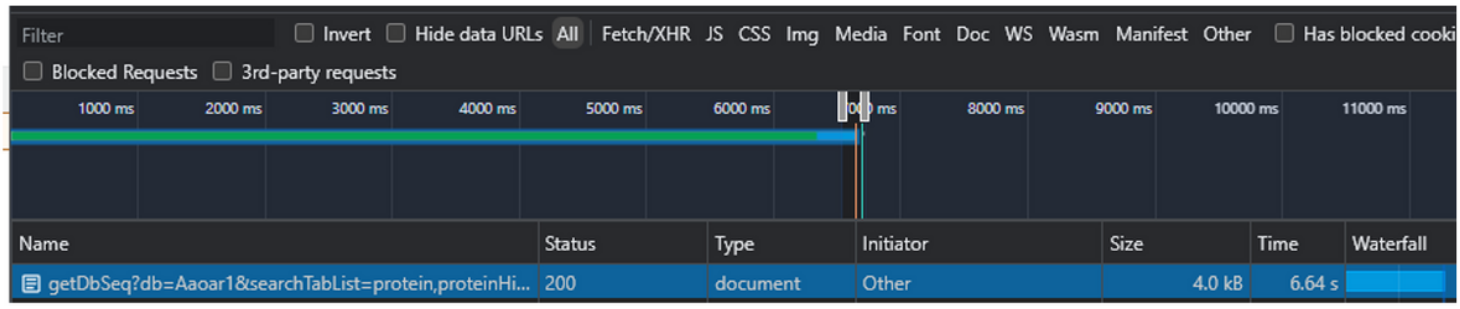

Figure 8

Network tool in web browser. You can see the time it takes to request the JGI Mycocosm database one by one server is 6.64 seconds; if the process were manual for 50,000 requests, it would take approximately 92 hours.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- AXSXGresults.csv
- RXLRresults.csv
- derechosAutorFungiRegEx.pdf