Understanding resilience optimization architectures with an optimization problem repository

Dr. Daniel Hulse

Robust Software Engineering

Intelligent Systems Division

NASA Ames Research Center

Hongyang Zhang

Dr. Christopher Hoyle

Design Engineering Lab

Dept. of Mech., Ind., & Mfg. Engineering

Oregon State University

IDETC/CIE 2021

August 17 – 20, 2021

47th Design Automation Conference (DAC)

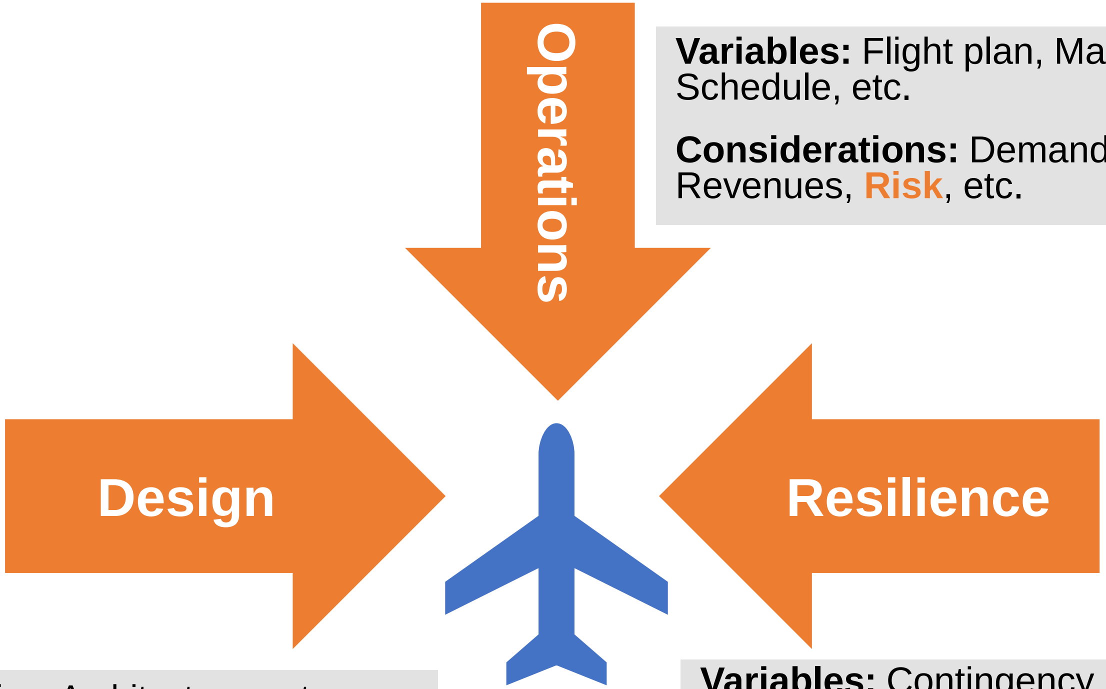
DAC-09: Design for Resilience and Failure Recovery

Virtual Presentation

DETC2021-70985

Oregon State University

Why Study Resilience Optimization?



Variables: Flight plan, Maintenance Schedule, etc.

Considerations: Demand, Usage, Revenues, **Risk**, etc.

Variables: Sizing, Architectures, etc.

Considerations: Performance, Efficiency, **Component Reliability**, etc.

Variables: Contingency Management, Fault Trim Adjustment, Emergency Procedures, etc.

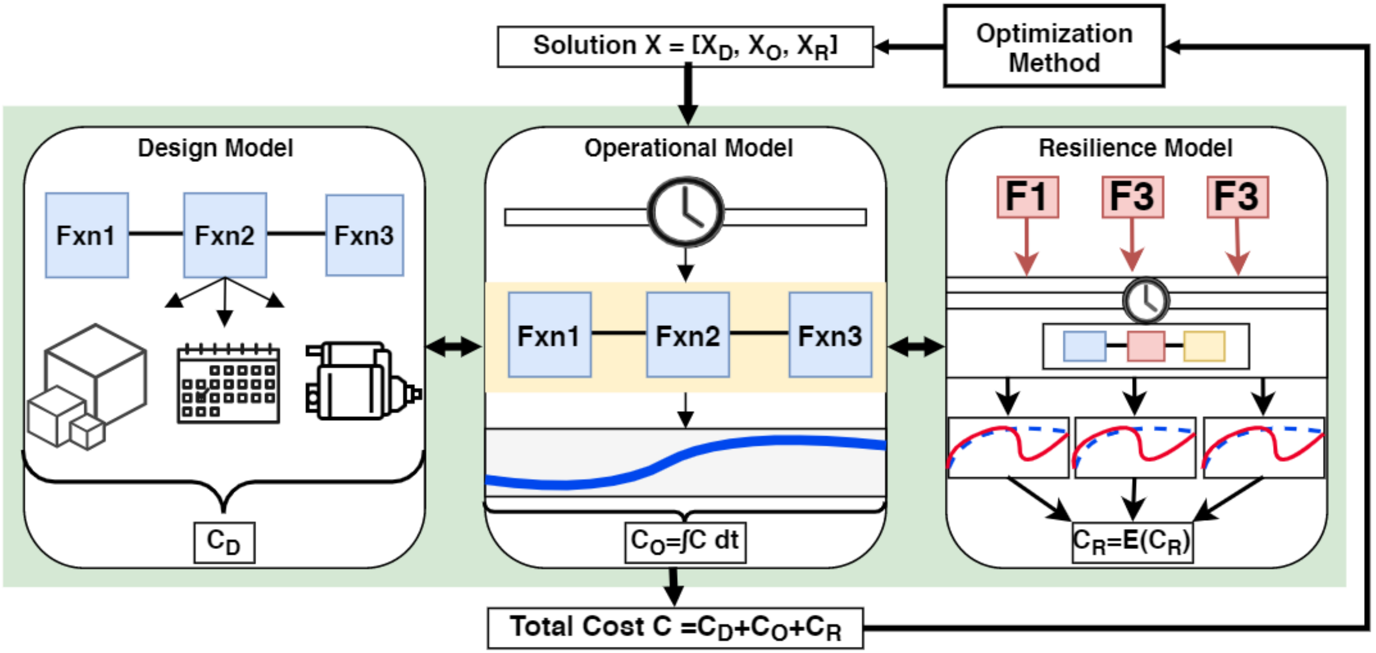
Considerations: **Safety, Damage, Flight Disruption**, etc.

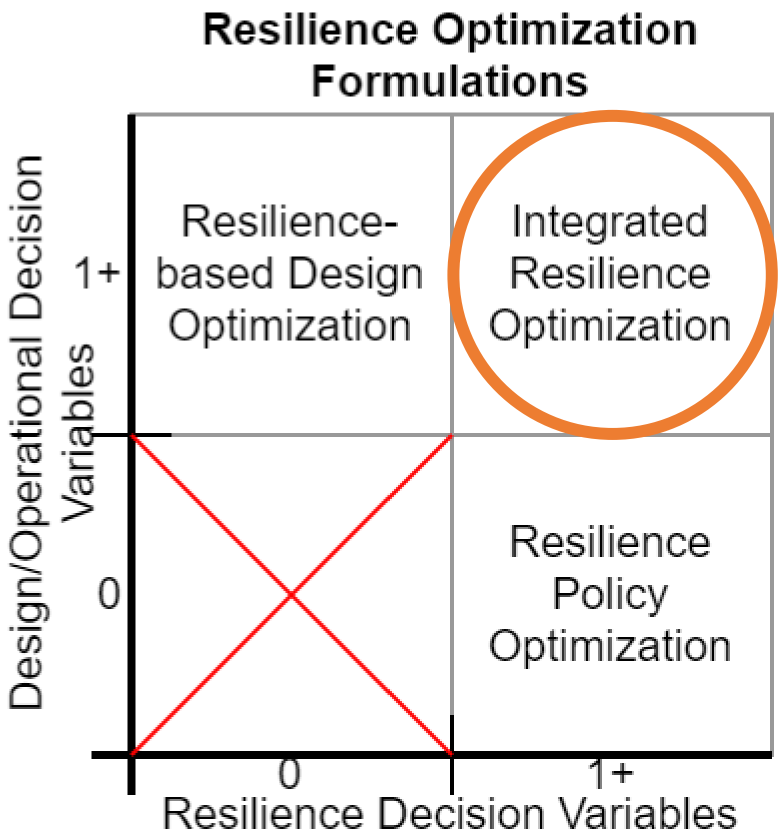
Design, **Operational**, and **Resilience-related** variables and objectives

Need to systematically **manage the complexities** to effectively explore the trade-space

Oregon State University

Previous Work





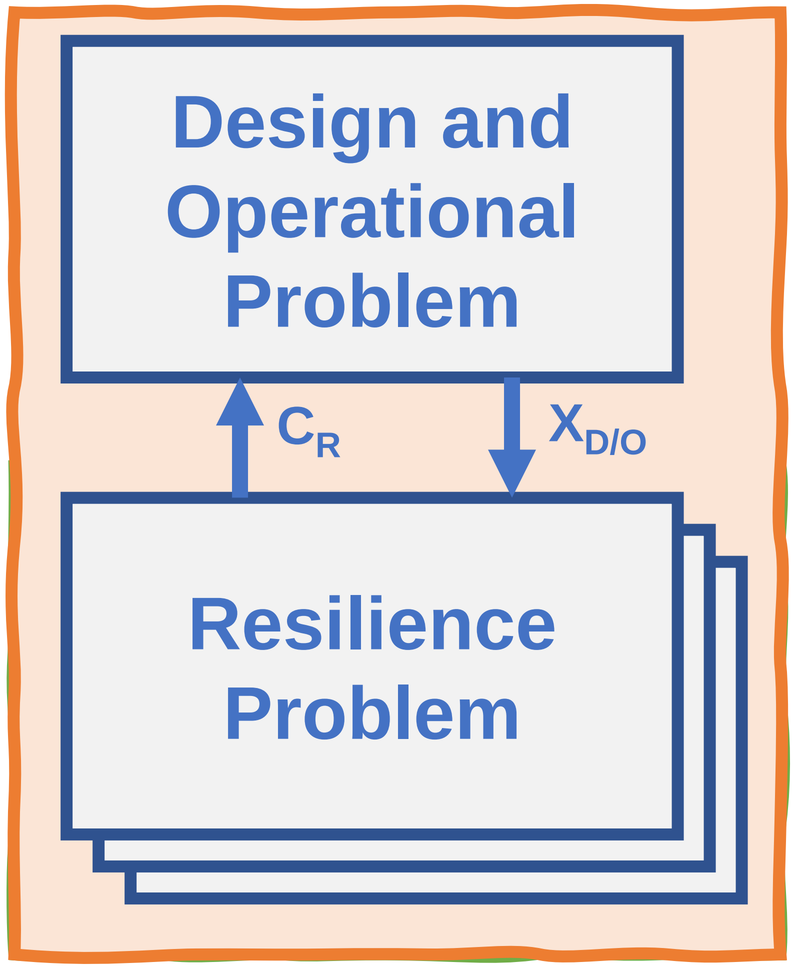
Previous work:

- Developed overall **Integrated Resilience Optimization** framework [1]
  - Combined optimization of Design, Operations, and Resilience variables and objectives
- Formulated different architectures and performed preliminary comparisons [1]

[1] Hulse, D., Biswas, A., Hoyle, C., Tumer, I. Y., Kulkarni, C., & Goebel, K. (2021). Exploring Architectures for Integrated Resilience Optimization. Journal of Aerospace Information Systems, 1-14.

Oregon State University

Previous Work: Architectures



Two types of decomposition:

- **Design/Resilience Levels**
  - All-in-one
  - Bilevel
  - Alternating
  - Sequential
- **Resilience Model Scenarios**
  - Monolithic resilience model
  - Scenario-independent resilience model
  - Grouped-Scenario Resilience model

Oregon State University

Why Develop a Repository?

Resilience Optimization Problem

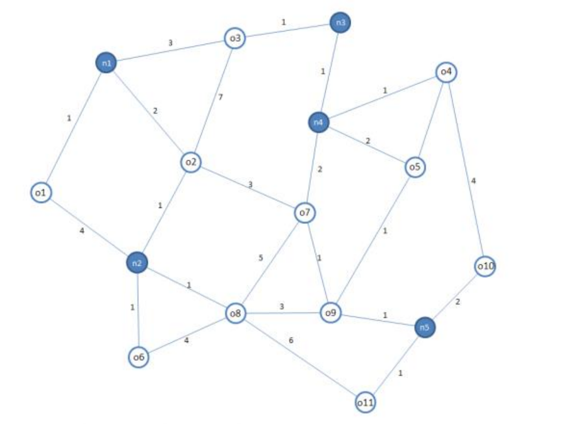
$$min_{\mathbf{x}}\, C_{D/O}(\mathbf{x}) + C_R(\mathbf{x})$$

$$where\ C_R = \sum_{s \in S} n * r_s * C_s(\mathbf{x})$$

- Small or large **number of scenarios**
- **Discrete or continuous** variables
- **Analytic or simulation** models

Different problem properties and types imply **one architecture may not fit all**

Travelling Salesman Problem



Lead to the development of Ant Colony Optimization, other algorithms.

MDO Test Suite

Table 2. Characteristics of Test Problems in MDO Test Suite

No.	Name	# of design variables (DV)	# of constraints	Notes	Status
1.1	Heart	8	8	algebraic eqs.	Done
1.2	Propane	10	10	algebraic eqs.	Done
2.1	Aircraft	10	2	empirical curve fits	Planned
2.2	Hub	many	many	parallel processing	Done
2.3	Electronic	8	3		Done
2.4	Speed	7	11	multilevel	Testing
2.5	Power	6	4		Done
2.7	Rule-based	5	5	discrete DV	Done
3.1	HSCT	44	300	GSE and database	Done
3.2	Space	163	41	Needs EAL	Done
3.4	Aerospike	15	5		Planned
3.6	Aerospike	15	5	Needs NASTRAN	Planned
3.7	FIDO 2	many	many		Planned
3.8	Damper	1507	11	integer DV	Done

Helped benchmark and develop new and existing MDO architectures.

A problem repository can help:

- **Develop** new approaches and **benchmark** existing ones
- Understand **which architecture to use** on a given new problem

Oregon State University

NASA

Repository Problems

Problem	Des. Vars	Res. Vars	Architecture	Decomposition	Algorithms Used	Model Type	Sim. Framework
Notional Example	4 (C)	2 (C)	AAO, Bilevel, Alt. (both)	Monolithic	Trust-Region	Equations	Stand-alone
Pandemic Management	N/A	6 (C)	AAO	Monolithic	Differential Evolution	Dynamic	Stand-alone
Cooling Tank	2 (C)	54 (D)	Bilevel, Alt. (with C_R)	Monolithic	Powell's (D)/EA (R)	Dynamic	fmdtools [51]
Drone	3 (D)	2 (D)	AAO, Bilevel, Seq. (no C_R)	Monolithic, Scenario-Set	Exhaustive Search	Dynamic	fmdtools [51]
EPS	14	N/A	AAO	Scenario-Set	Line search	Static	IBFM [52]
Monopropellant System	N/A	12 (D)	AAO	Monolithic	EA	Static	IBFM [52]

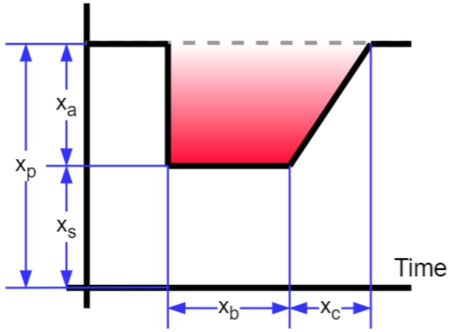
This work:

- **Collects 3 problems from previous work** (in [2])
  - **Monopropellant System:** First problem used to demonstrate resilience optimization
  - **EPS Problem:** Used to demonstrate resilience model decomposition strategy
  - **Drone Problem:** Used for initial comparison of IRO architectures in exhaustive search
- **Adds 3 new problems:**
  - **Notional Example:** Simple IRO problem not requiring a detailed simulation
  - **Pandemic Management:** Demonstrates a more complex lower-level —(in development)
  - **Cooling Tank:** Demonstrates a domain with different problem types at each level

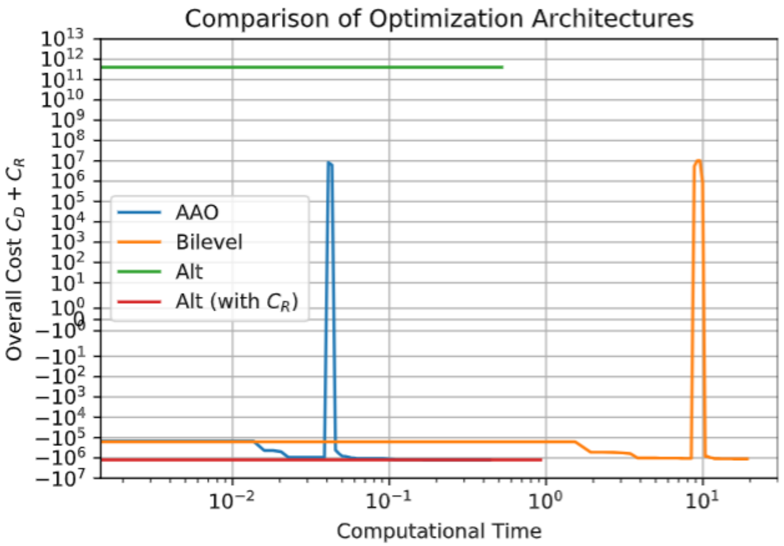
[2] Hulse, D. (2021). A Computational Framework for Resilience-Informed Design. Dissertation. Oregon State University.

Oregon State University

Architecture Comparisons



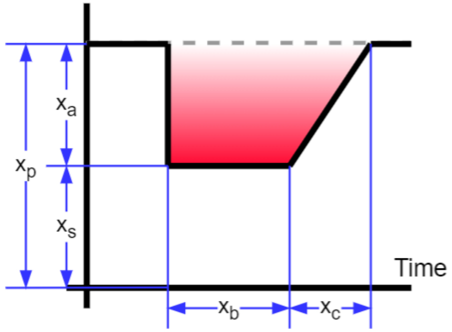
$$
\begin{aligned}
min_{\mathbf{x}}\ f(\mathbf{x}) =& C_D(x_p, x_a, x_r, x_s) + C_R(x_r, x_a, x_b, x_c) \quad (8)\\
s.t.\ C_D =& -a\sqrt{x_a+0.1} + b\sqrt{x_s+0.1} + c/\sqrt{x_r}\\
C_R =& x_r * n * (d * x_a * (x_b + x_c/2) + e/x_c + f/x_b)\\
h_{D1} =& x_p - (x_s + x_a) = 0\\
g_{D2} =& \frac{(x_p-1)^2}{c} - x_r < 0\\
\text{where}\quad & 100 > x_p > 0, 2 > x_a > 0, 100 > x_r > 10^-10,\\
& 2 > x_s > 0, 100 > x_b > 0, 100 > x_c > 0\\
\text{and}\quad & [a, b, c, n, d, e, f] = [1e6, 5e5, 100, 1e5, 50, 50, 20]
\end{aligned}
$$



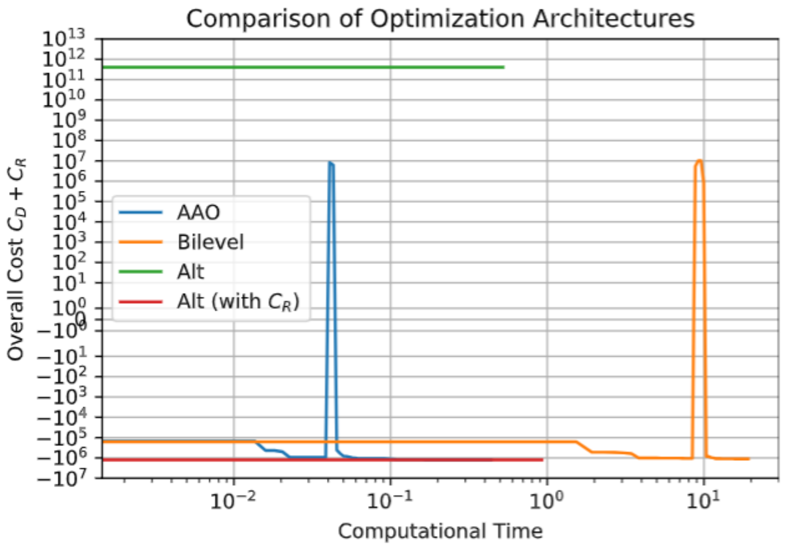
Notional Example (Cont. Trust Region)

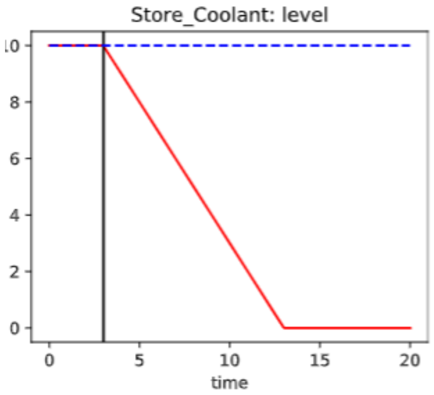
- **Bilevel:** orders of magnitude slower than AAO because each design gradient point requires a full lower-level re-optimization
- **Alternating:** most efficient but needs C_R in upper-level to be effective

Architecture Comparisons

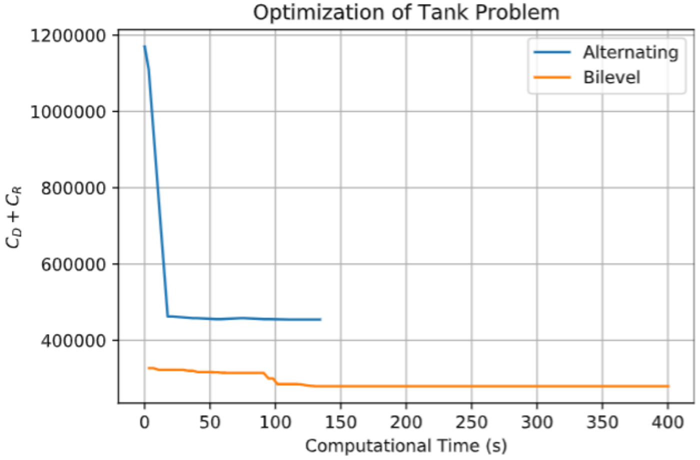


$$
\begin{aligned}
min_{\mathbf{x}}\, f(\mathbf{x}) =& C_D(x_p, x_a, x_r, x_s) + C_R(x_r, x_a, x_b, x_c) \qquad (8)\\
s.t.\; C_D =& -a\sqrt{x_a+0.1} + b\sqrt{x_s+0.1} + c/\sqrt{x_r}\\
C_R =& x_r * n * (d * x_a * (x_b + x_c/2) + e/x_c + f/x_b)\\
h_{D1} =& x_p - (x_s + x_a) = 0\\
g_{D2} =& \frac{(x_p-1)^2}{c} - x_r < 0\\
where\quad & 100 > x_p > 0, 2 > x_a > 0, 100 > x_r > 10^-10,\\
& 2 > x_s > 0, 100 > x_b > 0, 100 > x_c > 0\\
and\quad & [a,b,c,n,d,e,f] = [1e6, 5e5, 100, 1e5, 50, 50, 20]
\end{aligned}
$$





$$
\begin{aligned}
s.t.\; C_D =& 1000(x_T-10)^2 + 1000(x_T-10) + 10000x_I^2\\
C_R =& \sum_{s\in S} n * r_s * C_s(\mathbf{x_{ip}}, \mathbf{x_{op}})\\
g_R =& \sum_{f\in N} f \le 0\\
where\; & x_T \in (10-100), x_I \in (0-1),\\
& x_{ip,z} \in [-1,0,1], x_{op,z} \in [-1,0,1]
\end{aligned}
$$



<u>Notional Example</u> (Cont. Trust Region)

- **Bilevel:** orders of magnitude slower than AAO because each design gradient point requires a full lower-level re-optimization
- **Alternating:** most efficient but needs C_R in upper-level to be effective

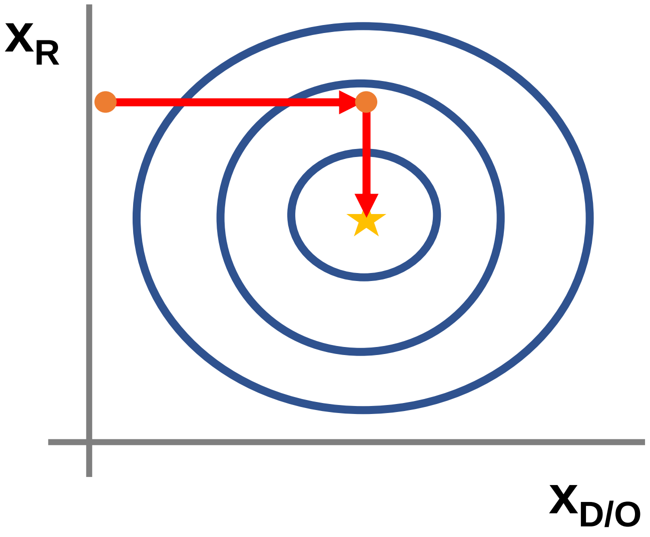
<u>Cooling Tank Example</u>

- Powell's Method in design model, evolutionary algorithm in resilience model
- Even with C_R in the upper-level, the **alternating approach is ineffective** compared to the bilevel architecture

Why do we have contrary results?

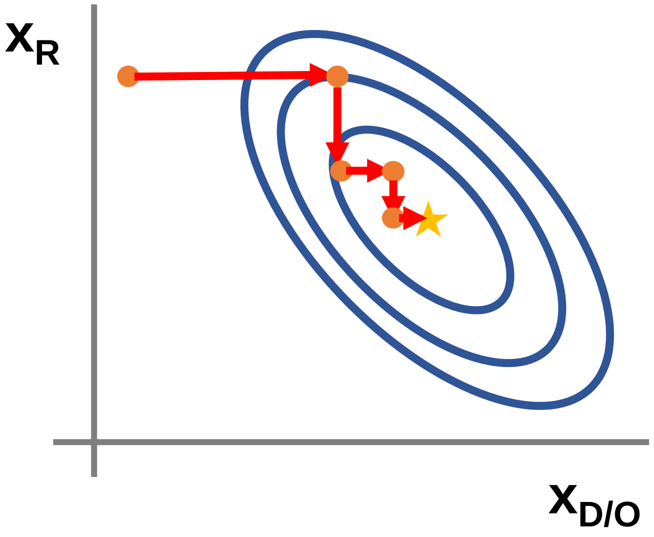
Differing levels of **Coupling:** the level to which design ($\mathbf{x}_{D/O}$) and resilience ($\mathbf{x}_R$) variables depend on each other.

Uncoupled: direct path from $x^*_{D/O}$ to x^*



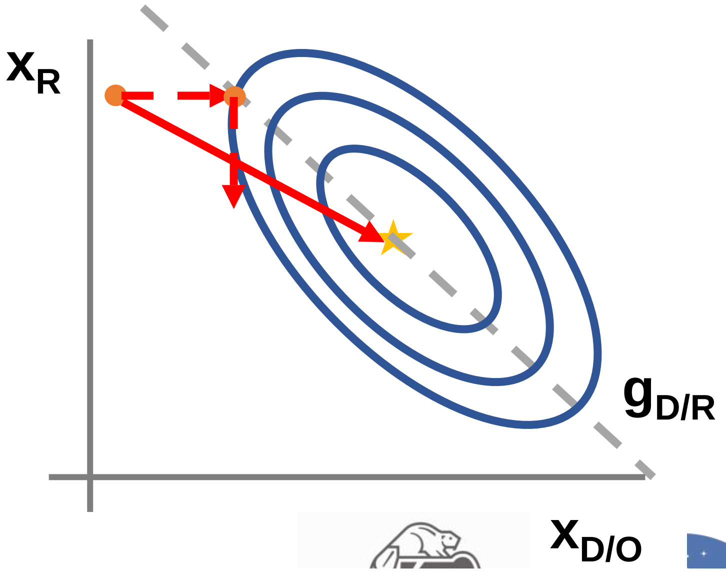
Enables **Sequential Approach**

Loosely coupled: unobstructed path that can be followed to x^* in alternating directions



Enables **Alternating Approach**

Fully coupled: joint steps $dx=[dx_{D/O},dx_R]$ must be taken to reach x^*



Solvable with **Bilevel** or All-at once Approaches

Overall Repository Theory/Findings

	Appropriate Architectures	
	Unaligned	**Aligned**
Fully Coupled	Bilevel, AAO	Bilevel, AAO
Loosely Coupled	Alternating (with C_R), Bilevel, AAO	Alternating, Bilevel, AAO
Uncoupled	Sequential (with C_R), Bilevel, AAO	Sequential, Bilevel, AAO

The applicability of Design/Resilience Level Decomposition Architectures depends on **couplings** between levels

Within Alternating and Sequential architectures, the use of a C_R in the upper level depends on the **alignment** of the Design and Resilience problems.

Independent ↑ ... ↓ Coupled

Resilience Problem Coupling	Appropriate Solution Approach
Scenario Independence	Two-stage approach
Independent Scenario Sets	Lower-level decomposition
Fully Coupled Scenarios	Monolithic lower-level

The applicability of scenario-based decomposition depends on the **couplings** between scenarios (i.e., whether a resilience variables map directly to scenarios/sets or not)

Oregon State University

NASA

Conclusions, Limitations, Future Work

Conclusions

- Developed a resilience optimization problem **repository**
- Compared optimization architectures for Integrated Resilience Optimization
- Applications help us understand **when optimization architectures apply to given problem formulations**

Limitations:

- Still developing pandemic problem
- Does not cover all **previous resilience optimization approaches/formulations** (e.g., two stage, etc…)

Future Work:

- Include and develop **more problems/formulations**
- Study other problem/architecture attributes (e.g., resilience model execution **parallelism**)

Links

Paper Link

ti.arc.nasa.gov/publications/20210010232/download

Repository Link

github.com/DesignEngrLab/resil_opt_examples

Contact

Daniel Hulse	daniel.e.hulse@nasa.gov
Hongyang Zhang	zhangho2@oregonstate.edu
Christopher Hoyle	chris.hoyle@oregonstate.edu