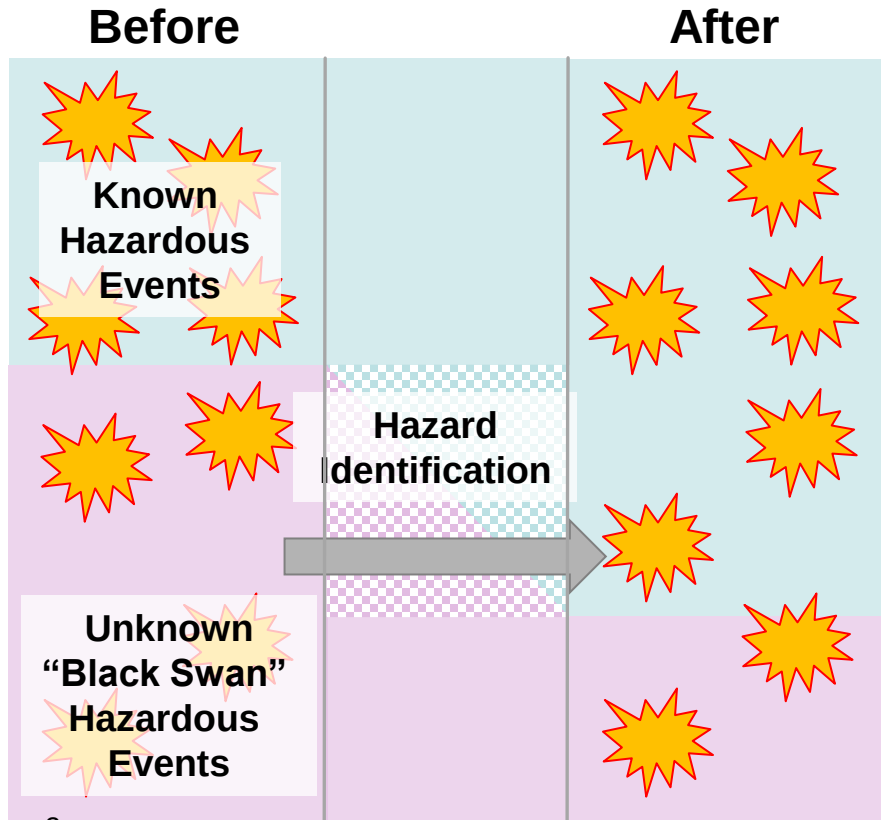


Uncovering Hazards Using Multi-Objective Optimization to Explore the Faulty State-Space

Inga Girshfeld*, **Daniel Hulse**, Lukman Irshad[‡]
NASA Ames Research Center (*USRA ‡KBR)
AIAA Scitech Forum, 23-27 January, 2023

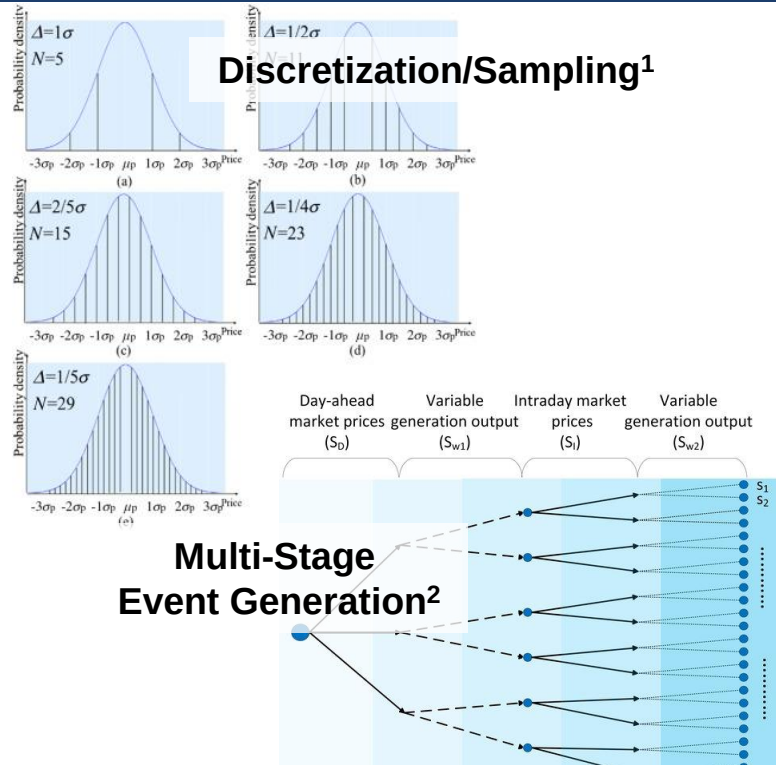
This material is a work of the U.S. Government and is not subject to copyright protection in the United States.

Motivation: Scenario Generation for Design



- Hazard Identification
 - **Reduces black swan risk** by uncovering otherwise unknown potential hazardous events
 - Difficulties:
 - Tedious, manual process
 - What do you do if:
 - There is no data?
 - Designer knowledge is incomplete?
- What if we could use computational search to find modes for us?
 - Opportunity: **find unforeseen modes**

Previous Work in Scenario Generation



- Formally defined in **stochastic programming**:
 - Methods to represent probability distribution with a few samples
- Methods extended to “less formal” settings
 - **Human-in-the-loop simulation**—represent situations to participants
 - **Software testing**—to verify function/performance requirements
- Ongoing research in **autonomous systems**
 - Verify that autopilot can respond in wide variety of hazardous situations normally expected of human

Diagram 1: Zang, P., Sun, G., Zhao, Y., Luo, Y., & Yuan, X. (2020). Stochastic optimization based on a novel scenario generation method for midstream and downstream petrochemical supply chain. Chinese Journal of Chemical Engineering, 28(3), 815-823.

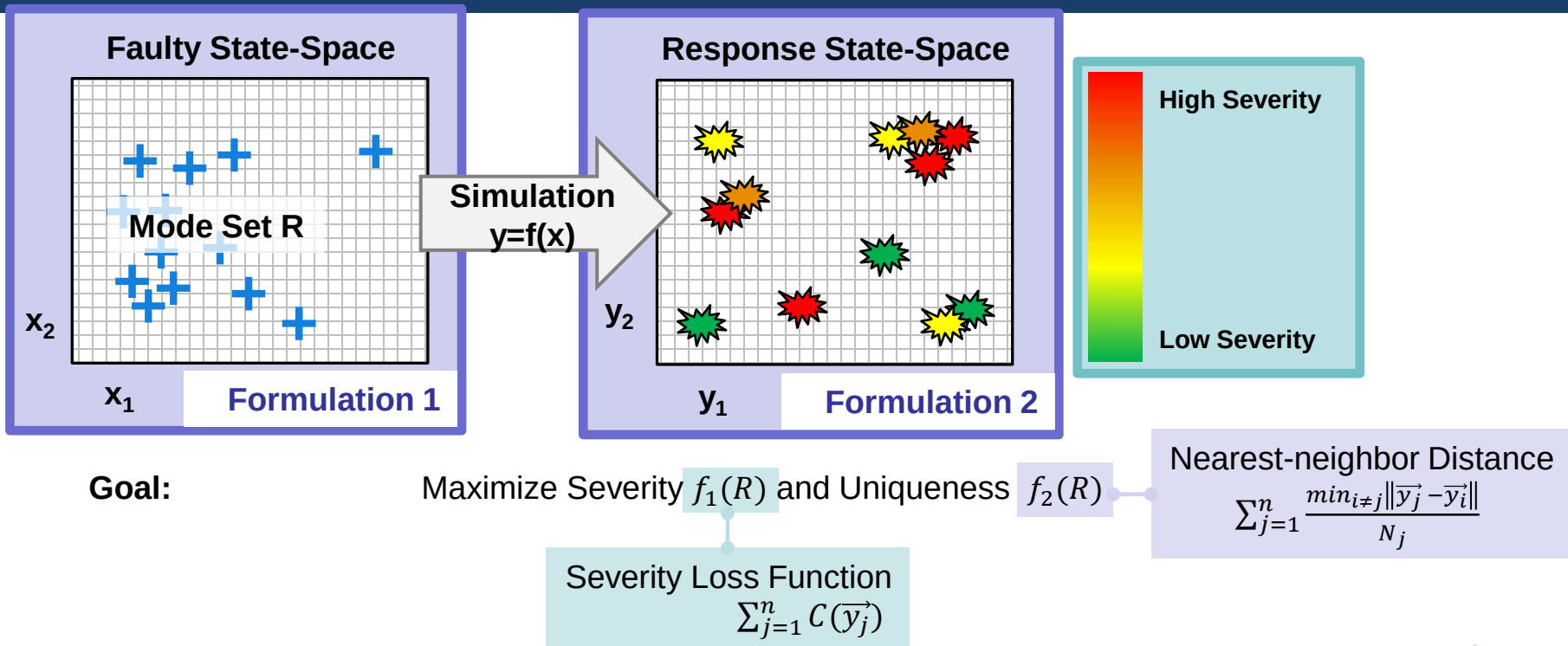
– **Diagram 2:** Han, X., Kardakos, E. G., & Hug, G. (2017, December). Trading strategy for decentralized energy resources in sequential electricity markets: A swiss case study. In 2017 IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia) (pp. 1-7). IEEE.

Previous Work in Scenario Generation

- Autonomous systems approaches
 - Data-driven approaches: Get scenarios from accident data
 - Knowledge-based generation: Get scenarios from experts
 - **Adversarial generation: Search state-space to find worst-cases**
- **Key gap:** Finding not just the worst, but a unique set
 - Oliveira et al. (2022) introduced the concept of searching with a **diversity-based genetic algorithm** optimizing uniqueness and severity.
 - We extend this framework with a **Cooperative Coevolutionary Algorithm** and **study the effect of weighting factors** in the formulation

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2022). A diversity-based genetic algorithm for scenario generation. *European Journal of Operational Research*, 299(3), 1128-1141.

Problem Formulation

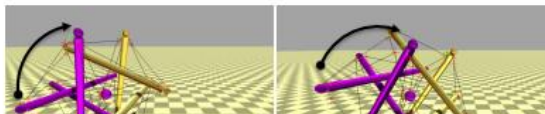


5

With respect to:

Set of Modes $R = \{[x_1 \dots x_s]_1 \dots [x_1 \dots x_s]_n\}$

Cooperative Coevolution Overview



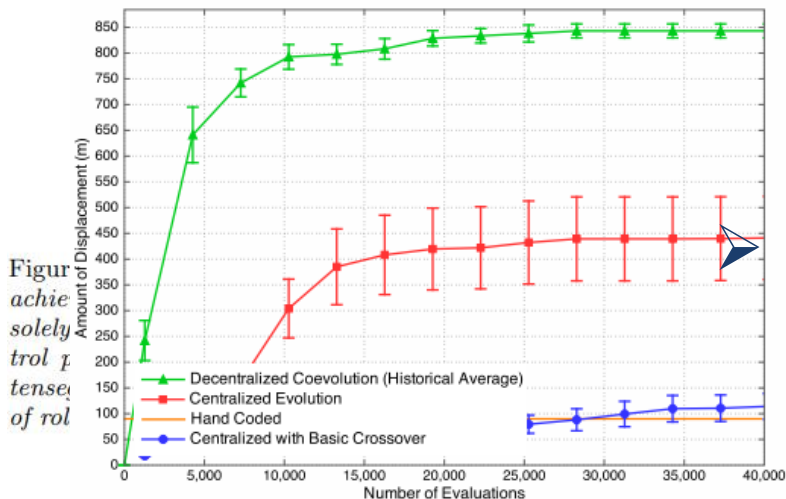
➤ Used in optimization of multiagent systems

➤ Goal: coordinate agents to maximize overall reward

We think this is analogous to selecting this set of modes

➤ Independent “local” severity objective

➤ Coupled “global” uniqueness objective



Isken, A., Agogino, A., SunSpiral, V., & Tumer, K. (2013, July). Controlling tensegrity robots through evolution. In Proceedings of the 15th annual conference on Genetic and evolutionary computation (pp. 1293-1300).

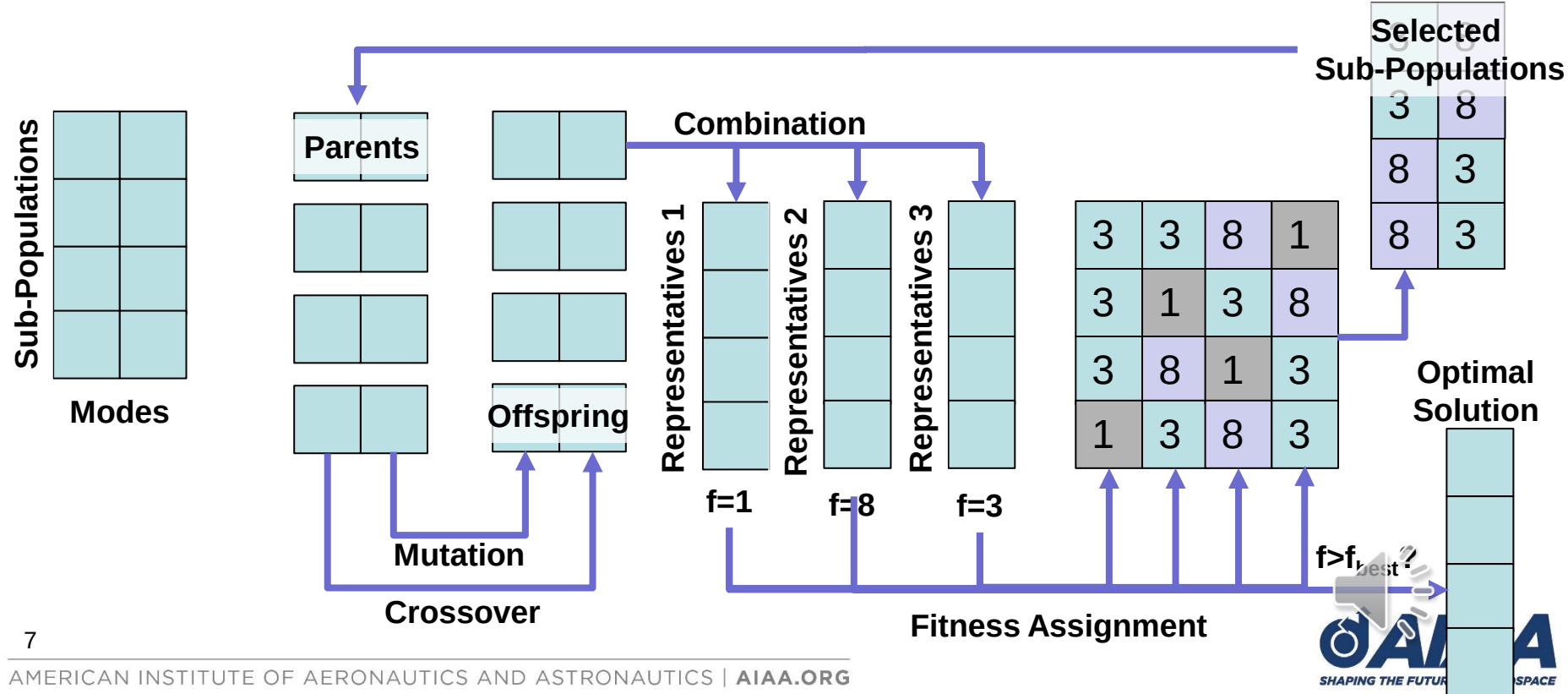
Cooperative Coevolution - Method

1: Instantiation

2: Permutation

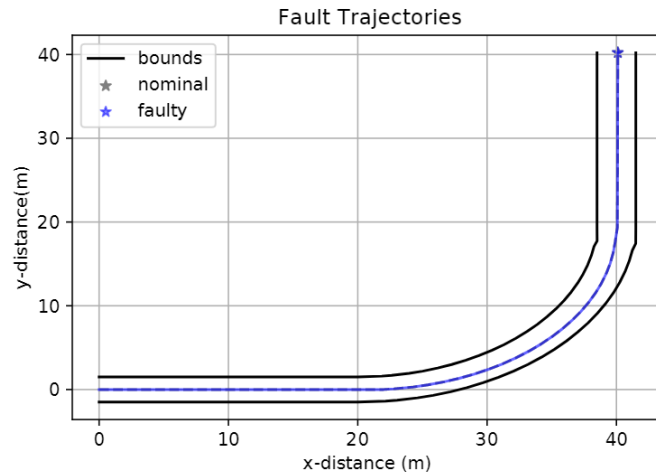
3: Evaluation

4: Selection

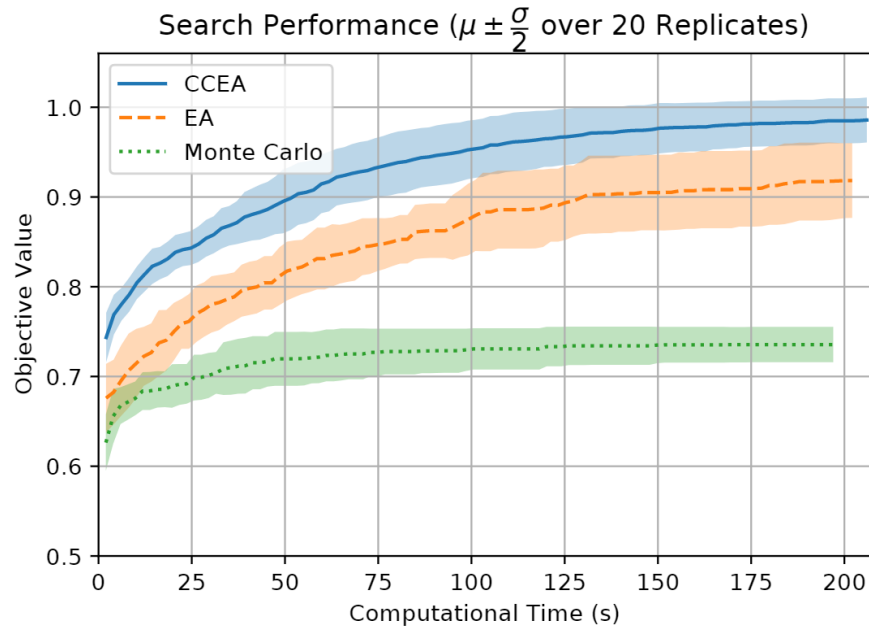


Case Study Demonstration

- Autonomous rover
 - Follows line markings to end-point
 - Subject to disturbances:
 - Friction
 - Transfer
 - Drift
 - Severities: Deviation from line markings
 - Variety Formulations
 - Input State-Space Distance (nearest-neighbor)
 - Output End-location Distance (nearest-neighbor)



Study 1: Effectiveness of CCEA



- Compared with EA and random search
- CCEA gives a more efficient search because of the separation of mode and set optimization

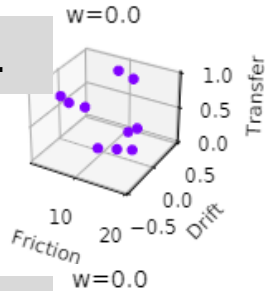
Study 2: Effect of Formulation and Weights

- Formulations
 - Input space distance
 - End-location distance
- Weights
 - 0 – all variety
 - 0.25 – preference variety
 - 0.5 – equal weighting
 - 0.75 – preference severity
 - 1 – all severity

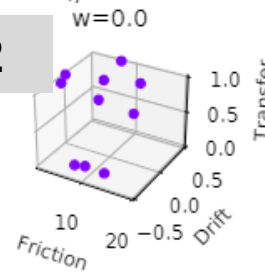
Formulation Comparison

Input Space

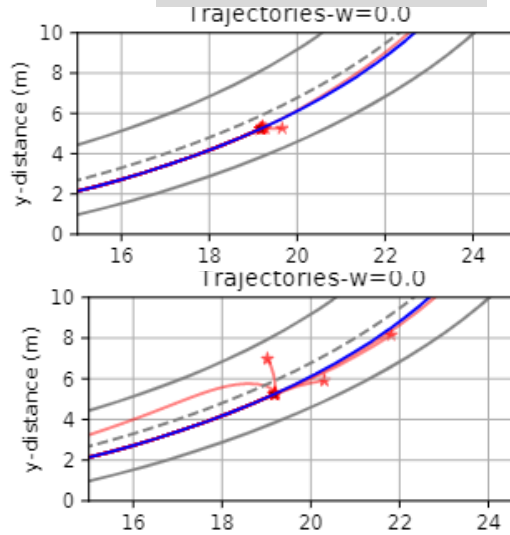
Form. 1



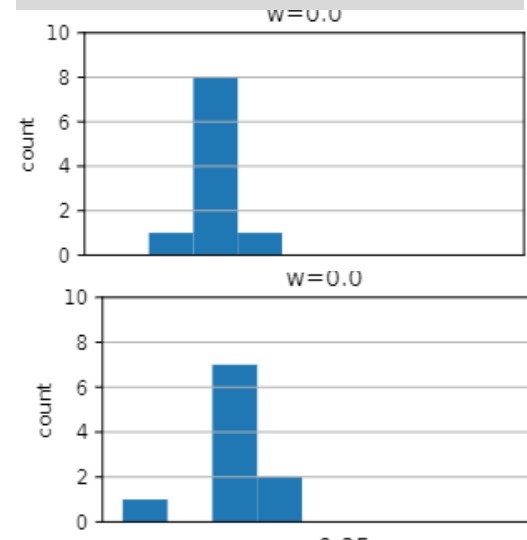
Form. 2



Trajectories

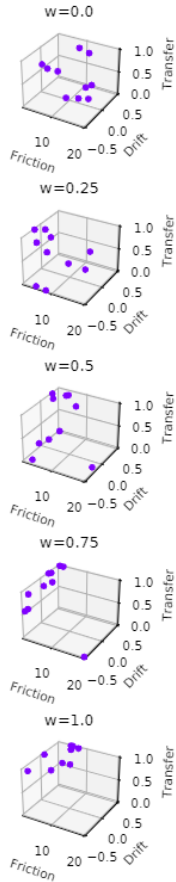


Severity distribution

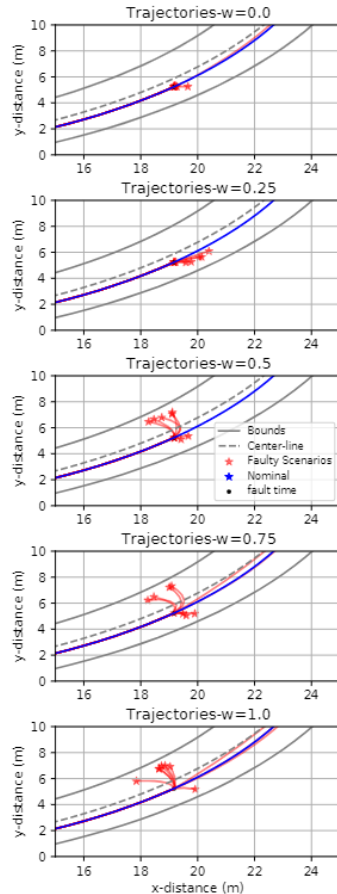


- Input-space formulation covers space but trajectories are less interesting
- Output-space formulation finds large variety of trajectories without sacrificing input-space variety

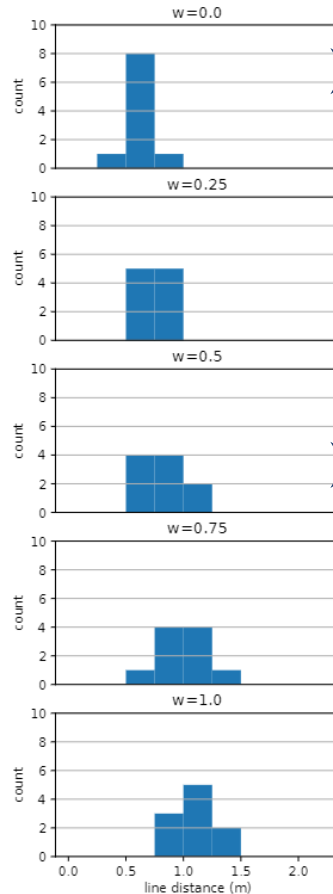
Effect of Weights- Formulation 1



(a) Faulty State-Space.



(b) Resulting Trajectories.



(c) Resulting Line Distances.

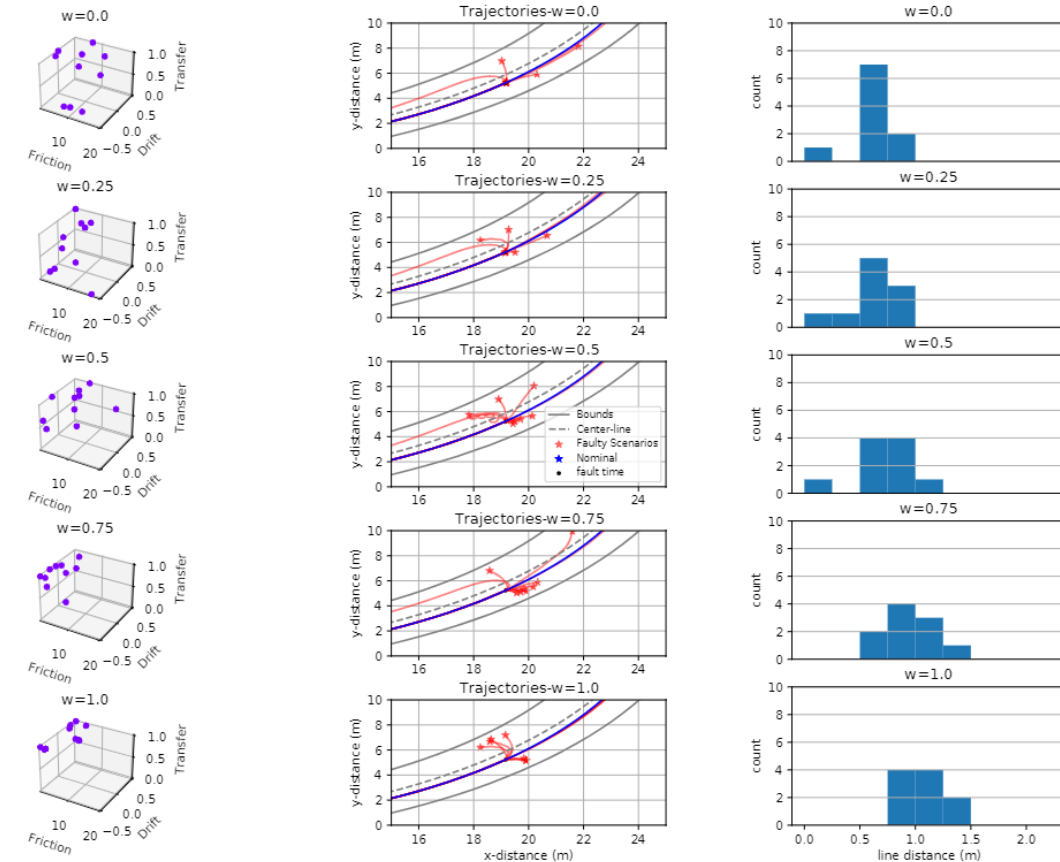
➤ Increasing weight leads to less fault-space variety

➤ Emergence of fault-space clusters

➤ Also increases # of high-severity modes

➤ End-location further from start point

Effect of Weights – Formulation 2



(a) Faulty State-Space.

(b) Resulting Trajectories.

(c) Resulting Line Distances.

- Increasing weight leads to less end-state variety
- Convergence toward left or right
- Equal-weight solution gives good balance

Discussion

- Formulation 2 (output states) gives more seems to give more interesting results than Formulation 1 (input states)
 - Both may be worthwhile—output states for **exploring possible consequences** and input states for **exploring possible modes**
 - What about a combined approach?

Conclusions

- Formulating scenario generation as a multiobjective optimization of variety and severity can help **identify more modes** than just the worst case
- CCEA performs better than a conventional EA on this problem
- It can be challenging to find the right balance (weightings) between objectives to give interesting results
 - Future work should explore using non-dominated sorting rather than the weighted sum approach



AMERICAN INSTITUTE OF
AERONAUTICS AND ASTRONAUTICS