WHEN TRUST MATTERS

# The NASA and DNV Challenge on Optimization under Uncertainty

# Why Uncertainty Quantification (UQ) is important for both DNV and NASA

NASA and DNV both work with safety-critical systems that must operate reliably under extreme and uncertain conditions.

Both face challenges due to sparse or expensive-to-collect data, making Uncertainty Quantification (UQ) difficult, but even more important.

To address these common chellenges, NASA and DNV have jointly developed a cross-sector UQ challenge

# The NASA and DNV Challenge on Optimization under Uncertainty

NASA and DNV launched a joint challenge to the community to improve how we design and control complex systems—like spacecrafts or offshore structures—under uncertainty.

The challenge invited participants from across many different scientific and engineering fields.

**Challenge Goals**

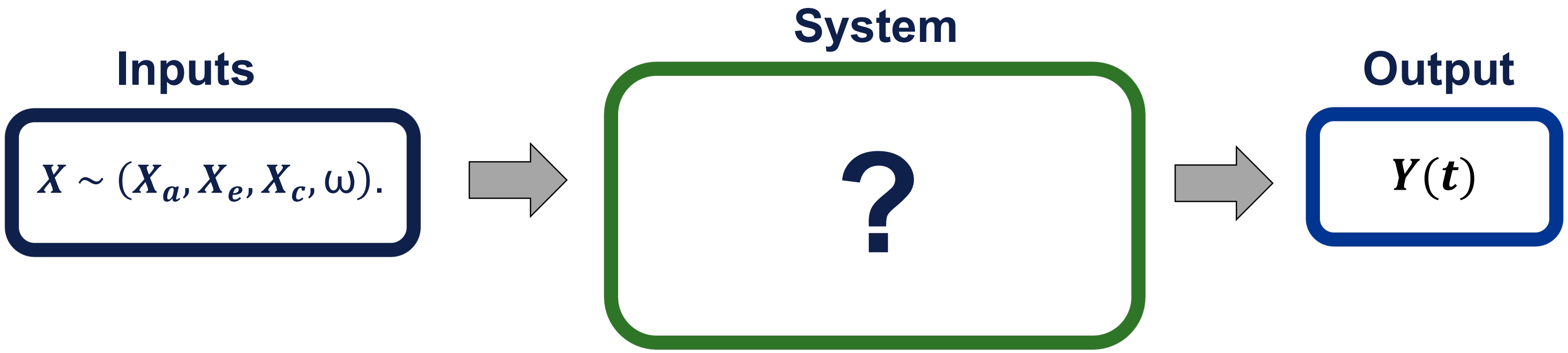
- Understand and manage two types of uncertainty:
  - *Aleatory* (random variability, e.g., weather)
  - *Epistemic* (lack of knowledge, e.g., unknown design parameters)

- Learn from *limited data* to understand system behavior and the *associated uncertainties*

- Optimize system performance while ensuring safety and reliability.

# A Domain-Agnostic Challenge Design

- The problem is intentionally defined in a **domain-agnostic** way to encourage the development of generalizable methods that are not tailored to any specific domain or physical context.
- Participants are only given **inputs and outputs**, with no knowledge of the underlying system.



# Input Variables:

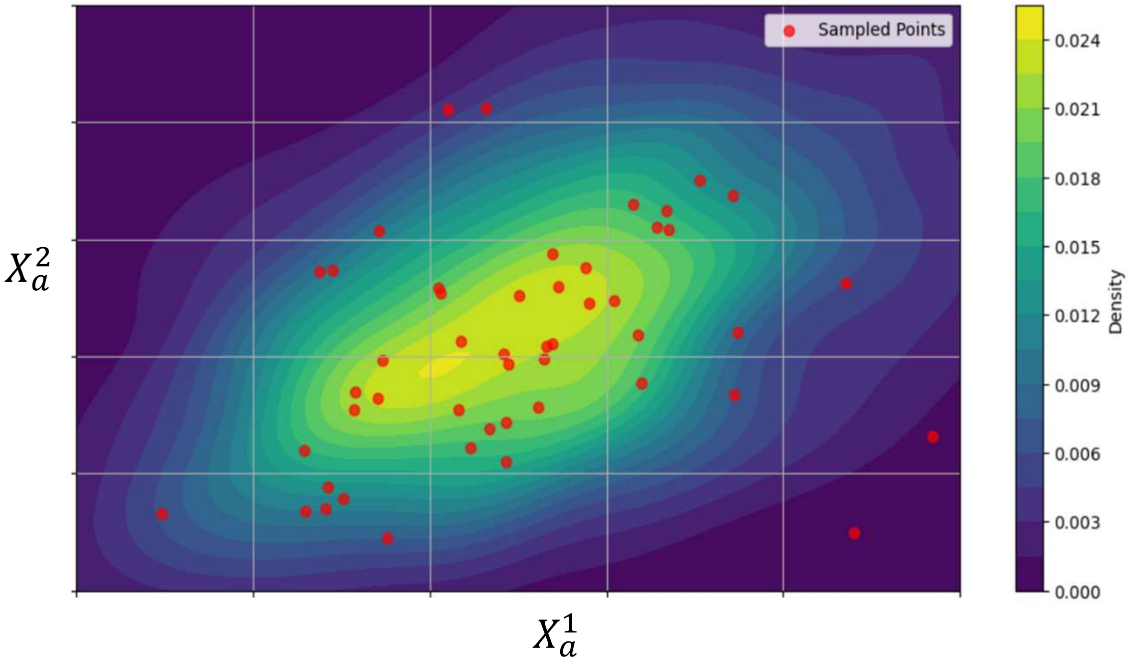
**Aleatory uncertainty,** $\boldsymbol{X}_a = [X_a^1, X_a^2]$:

Two variables given by joint distribution

**Epistemic uncertainty,** $\mathbf{X}_{\mathrm{e}} = [X_e^1, X_e^2, X_e^3]$

**Control parameters,** $\boldsymbol{X_c} = [X_c^1, X_c^2, X_c^3]$

$\boldsymbol{X}_a = [X_a^1, X_a^2]$:




Total input variables is then given by $\boldsymbol{X} \sim (\boldsymbol{X_a}, \boldsymbol{X_e}, \boldsymbol{X_c}, \omega)$,

with an additional aleatory variable ω representing the random seed.

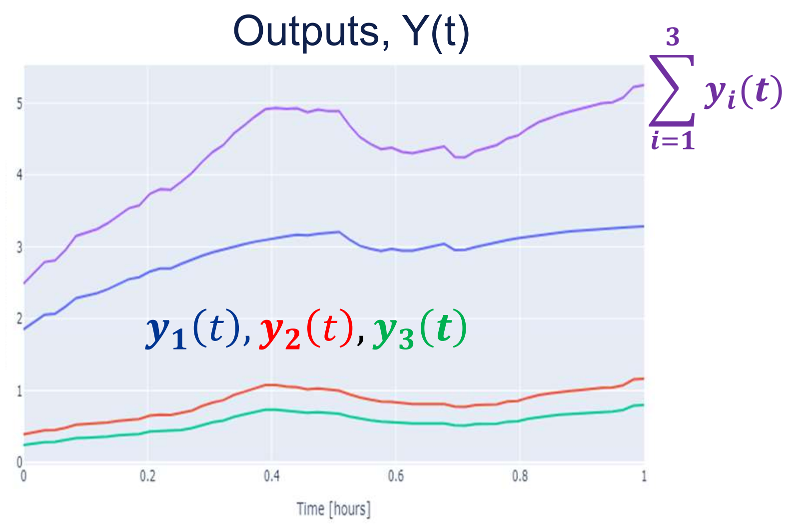
# Response, $Y(t)$

- **Input:** $\boldsymbol{X} \sim (\boldsymbol{X_a}, \boldsymbol{X_e}, \boldsymbol{X_c}, \omega)$.

- **Output:** The simulations will generate time series data $Y(t)$ for the outputs, which can be expressed as:

  $$\boldsymbol{Y}(X)(t) = \{\boldsymbol{y_1}(t), \boldsymbol{y_2}(t), \boldsymbol{y_3}(t), \boldsymbol{y_4}(t), \boldsymbol{y_5}(t), \boldsymbol{y_6}(t)\}$$

- **Dataset:** The participants received an initial dataset, $D$, consisting of $N$ samples: $D = \left\{X^i, Y^i\right\}_{i=1}^{N}$




# Available Simulation Models:

**Local model:** Infinite number of simulations

Run a local model (compiled binary executable) to calculate $Y(\boldsymbol{X})$ , where $\boldsymbol{X} = (\boldsymbol{X}_a, \boldsymbol{X}_e, \boldsymbol{X}_c, \omega)$.

| Inputs | | System | | Output |
|---|---|---|---|---|
| $X = (\boldsymbol{X}_a, \boldsymbol{X}_e, \boldsymbol{X}_c, \omega)$ | → | ? | → | $\boldsymbol{Y}(\boldsymbol{t})$ |

**Online model:** Limited number of simulations (*N* = 10)

Calculate $Y(\boldsymbol{X})$ , where $\boldsymbol{X} = \boldsymbol{X}_c$ . i.e., they can adjust the control parameters $\boldsymbol{X}_c$.

The aleatory variable $\boldsymbol{X}_a$ is sample from the **true distribution** and $\boldsymbol{X}_e$ set to its **correct values.**

| Inputs | | System | | Output |
|---|---|---|---|---|
| $\boldsymbol{X}_c$ | → | ? | → | $\boldsymbol{Y}(\boldsymbol{t})$ |

# Optimization Under Uncertainty

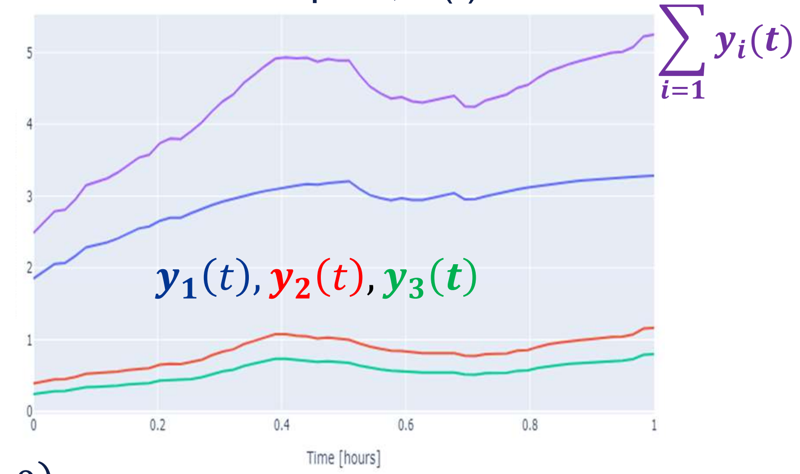
Objective Function to optimize: $J(X^i) = \sum_t (y_1^i(t) + y_2^i(t) + y_3^i(t))$

Limit state function: $g_j(Y(X)) = \mathrm{S_c^{\mathit{j}}} - y_\mathrm{j}$,

where $S_c$ represents maximum allowed value for the component $y_j, j \,\epsilon \{4,5,6\}$

Probability of breaking limit state constraint: $\mathrm{pof} = \sum_\mathrm{j} \mathrm{P}(g_j(Y(X)) < 0)$

Outputs, Y(t)

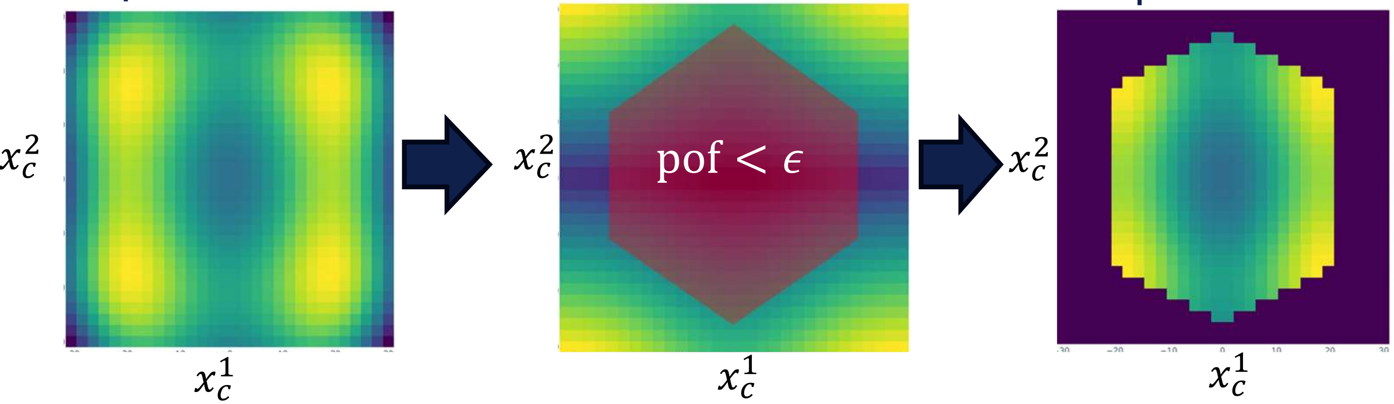




Unconstrained Optimization

Constrain mask: $\mathrm{pof} < \epsilon$ (e.g. $\epsilon = 10^{-3}$)

Constrained optimization



# Overview of Challenge Tasks:

**Problem 1: Uncertainty Model**

Create an uncertainty model for $x_a$ and $x_e$, given by the pair $(f_a, E)$, where $f_a$ is a probability density for $x_a$ and $E$ is an arbitrarily shaped set satisfying $x_e \epsilon E$

**Problem 2: Design Optimization**

- Performance-based design: $x_c$ that maximize objective function as $x_e$ varies in $E$
- Reliability-based design: $x_c$ that minimize the probability of failure as $x_e$ varies in $E$
- $\epsilon$-constrained design: $x_c$ that maximizes objective function as $x_e$ varies in $E$, under the constraint

$$\max_{x_e \epsilon E} \{\mathrm{pof}(x_e, x_c) \leq \epsilon\}$$



DNV