

From Deterministic to Probabilistic

- Why do probabilistic and where to start

Aakash Bangalore Satish, Sang-ri Yi

University of California, Berkeley

NSF NHERI Network



NATURAL HAZARDS ENGINEERING RESEARCH INFRASTRUCTURE (NHERI)



For more information, visit the
NHERI DesignSafe website: DesignSafe-ci.org



SimCenter

Goal: Develop an **open-source framework** and **extensible software tools** for researchers in **natural hazards engineering**

Transformational software that supports **uncertainty quantification (UQ)**, **performance-based engineering**, **community resilience**

“There is **uncertainty inherent in all aspects of earthquake engineering** that **needs to be addressed** on an ongoing basis with transformative research, process and code development, and focused implementation programs.”

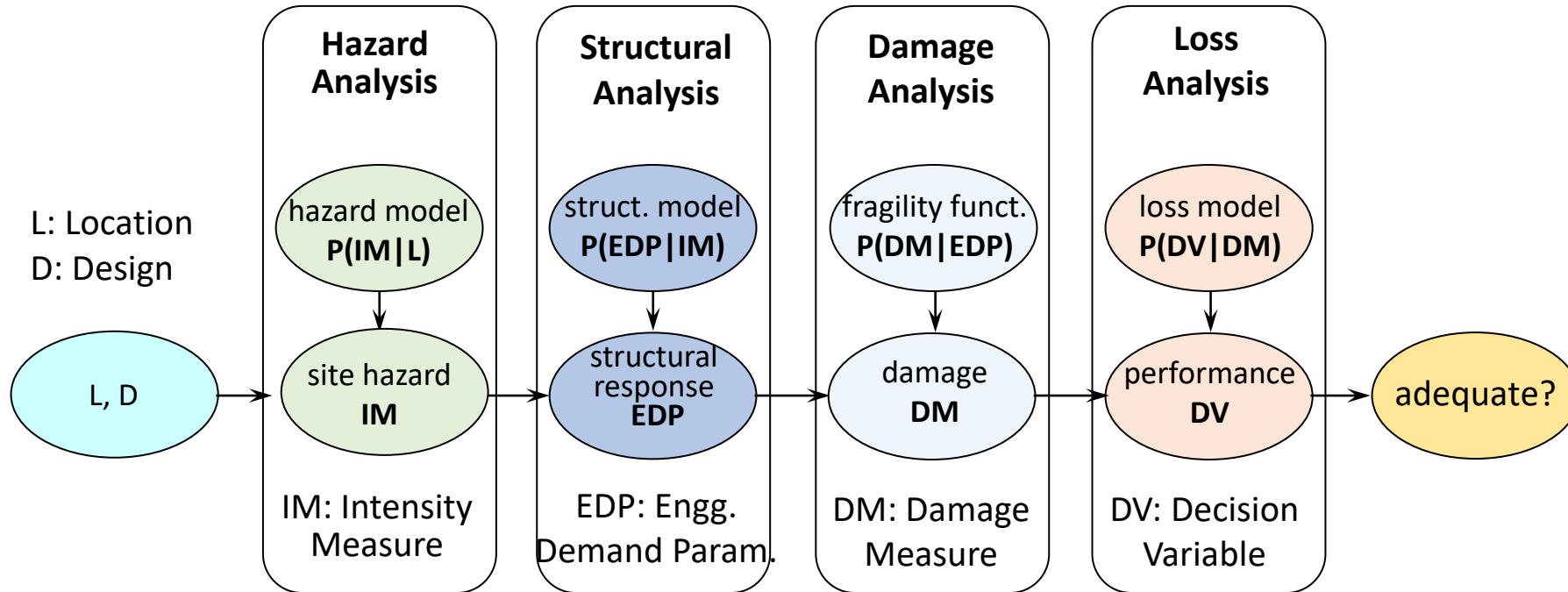
- National Research Council 2011. *Grand Challenges in Earthquake Engineering Research: A Community Workshop Report*. Washington, DC: The National Academies Press. <https://doi.org/10.17226/13167>



SimCenter's scientific workflow

Why UQ workflow?

PBE Framework (Adapted from Porter 2003)



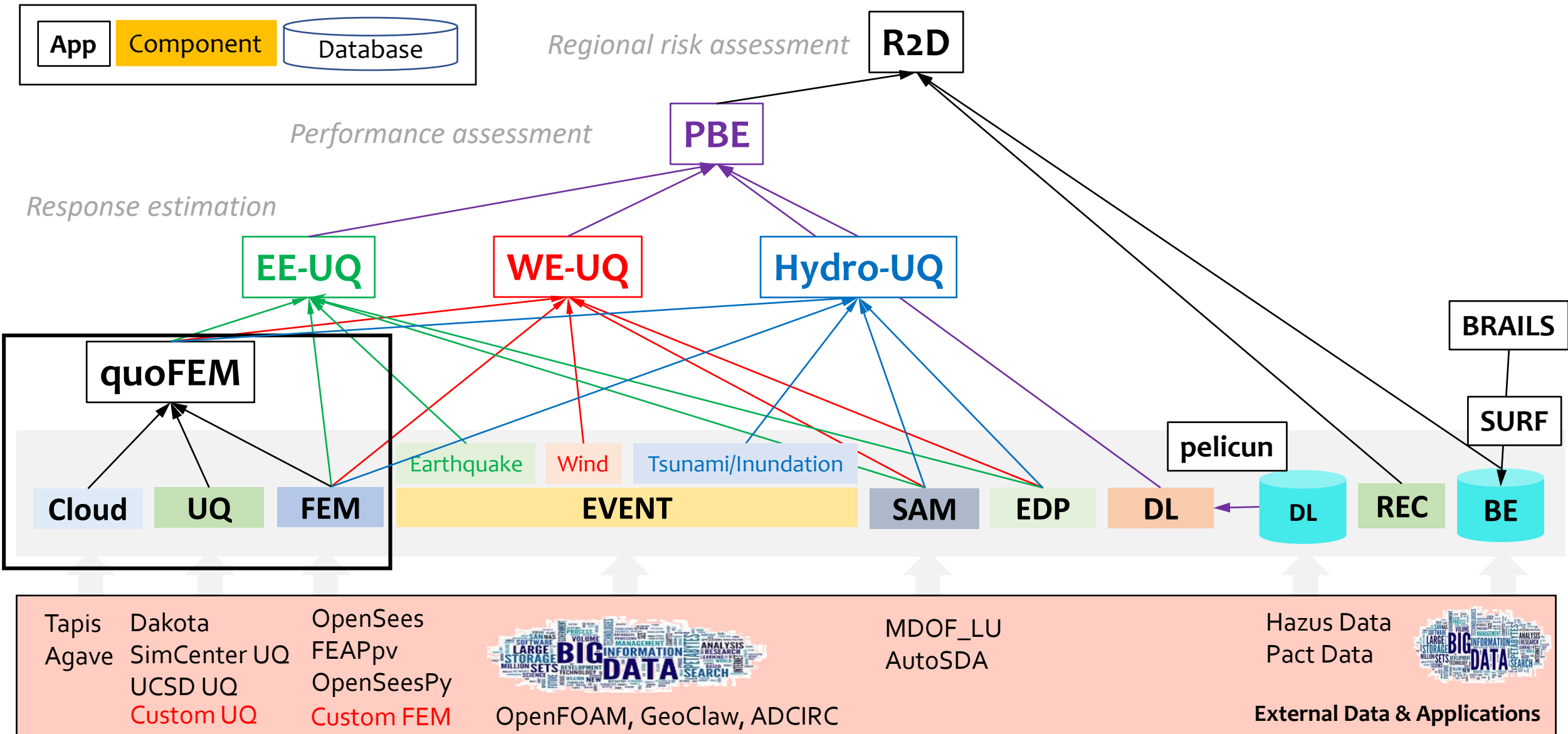
Uncertainty from several sources, including:

- Inherent variability
- Model parameter uncertainty
- Model form uncertainty



SimCenter's scientific workflow

SimCenter Applications



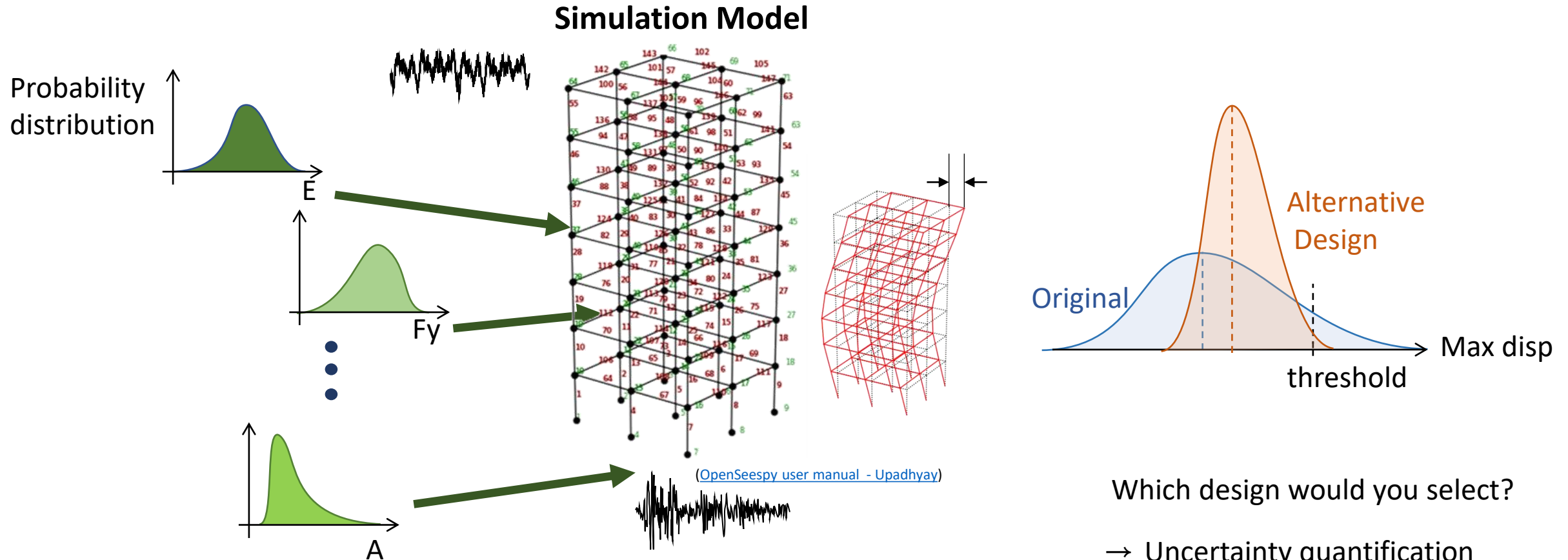
Types of UQ

Forward Uncertainty Quantification

Inverse Uncertainty Quantification

Forward UQ

- Propagation of uncertainty from inputs to outputs

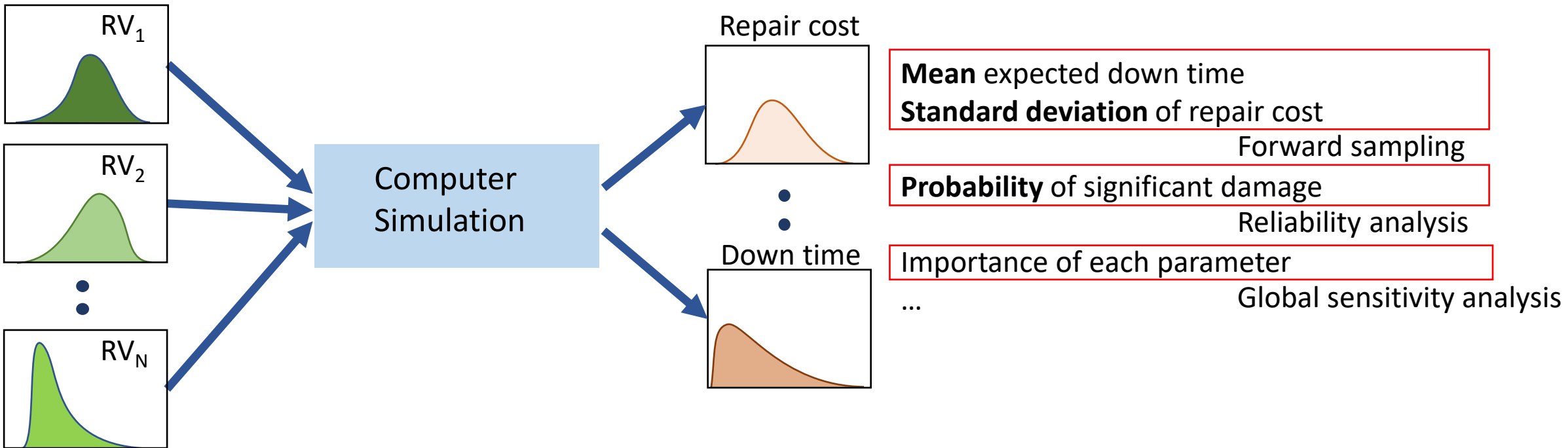


Which design would you select?

→ Uncertainty quantification is essential in making decisions

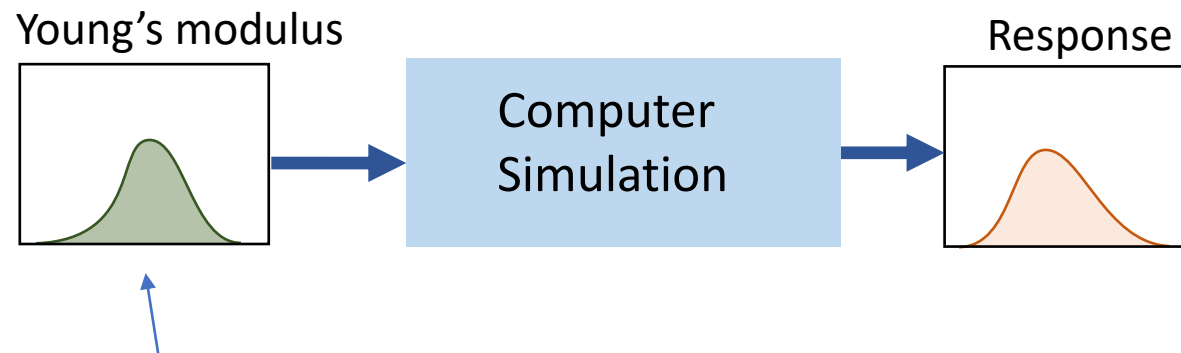
Forward UQ

- Based on assumptions on inputs, predict the uncertainty in outputs



Forward UQ

- Let us first consider single RV / Response



How do we get the input distribution?

Mean expected down time
Standard deviation of repair cost

Forward sampling

Probability of significant damage

Reliability analysis

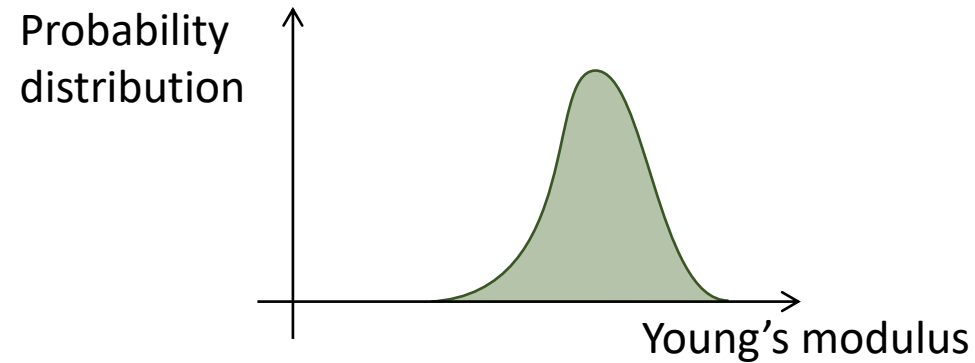
Importance of each parameter

Global sensitivity analysis

...

Probability distribution of RV

- Everything is **possible** but not everything is **probable**

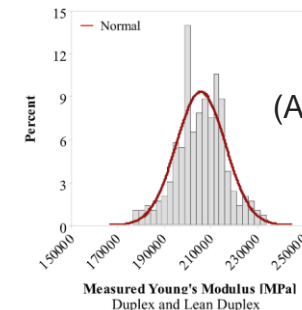
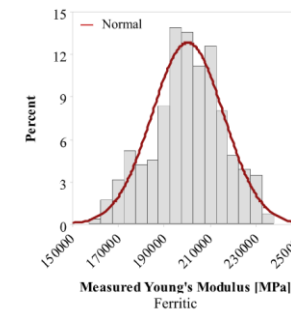
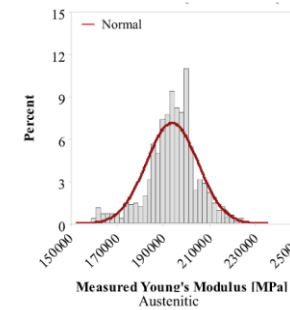


Engineering Judgement + Observation

- Based on underlying physics of quantity
- Select a simple distribution and apply reasonable limits
- Bayesian updating
→ inverse UQ

Examples

Known range of interest - Uniform distribution
Assumption - Gaussian with 10% variation
From reference



(Arrayago et al. 2020)

Forward Propagation

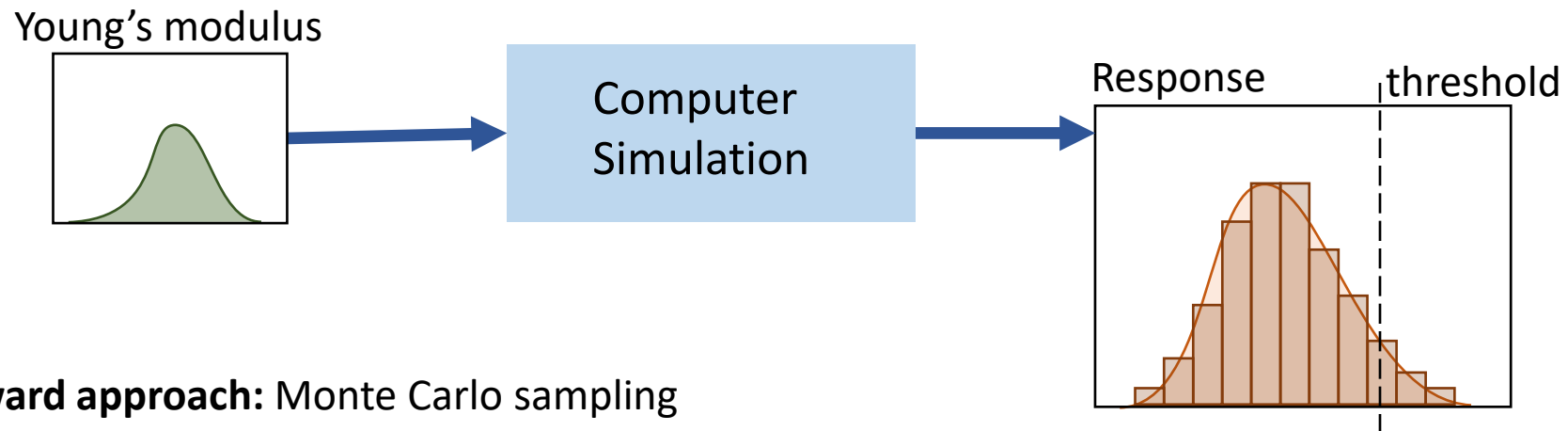
- Based on assumptions on inputs, predict the uncertainty in outputs



- A straightforward approach** Monte Carlo sampling
- Is the model numerically expensive?** Few simulations as much as possible
 - **Better UQ algorithms**
e.g. Latin hyper cube sampling
 - **Approximation methods**
e.g. Surrogate modeling
 - **Combination of both**
e.g. Multi-fidelity modeling

Reliability Analysis

- Probability of the response exceeding a threshold level
- Important for design decision



- **A straightforward approach:** Monte Carlo sampling
- **When the model is expensive & when failure probability is low**

It is desirable to reduce the number of simulations

$P_f = 10\%$ requires **1000** simulations

$P_f = 0.001\%$ requires **1000000** simulations

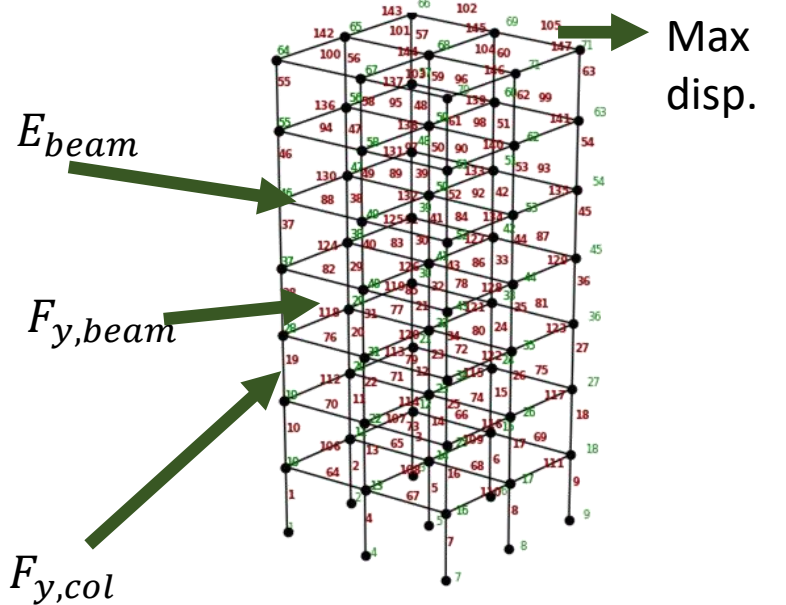
$$c.o.v = \sqrt{\frac{NP_f}{1 - P_f}} < 0.1$$

To reduce the number of simulations

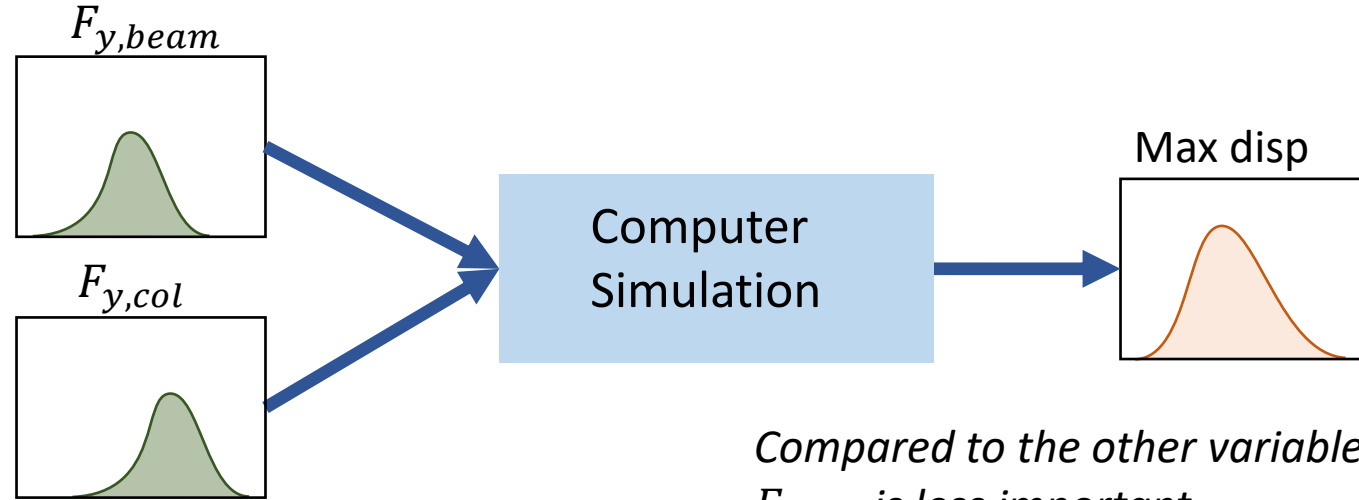
- **Better UQ algorithms**
Importance sampling, subset simulation
- **Approximation methods**
Surrogate modeling, First-order approximations
- **Combination of both**

Global Sensitivity Analysis

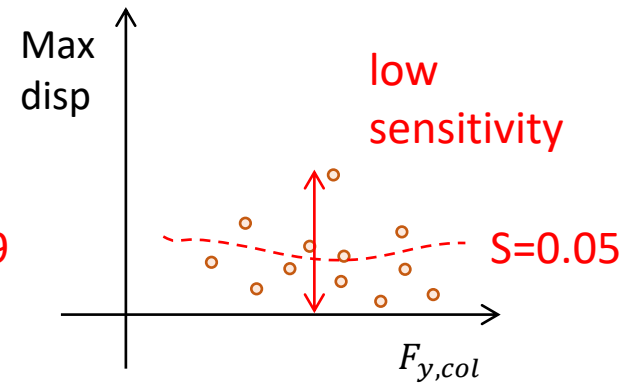
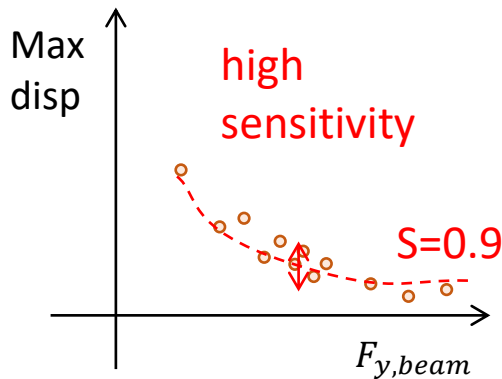
Are all the parameters equally impacting the response?



Probably not.
Then what are the variables that are **actually** influential?

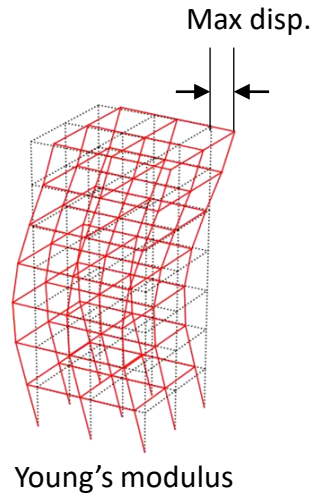


Compared to the other variables, $F_{y,col}$ is less important.
→ You might want to let $F_{y,col}$ be deterministic

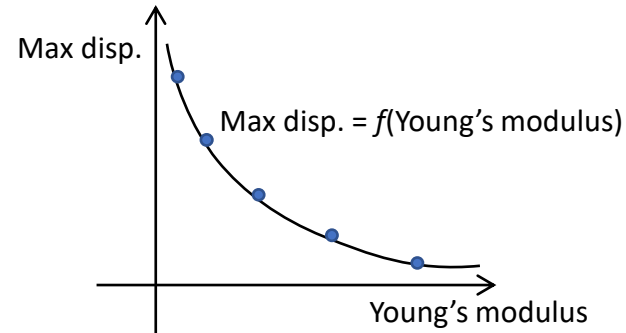


Surrogate modeling

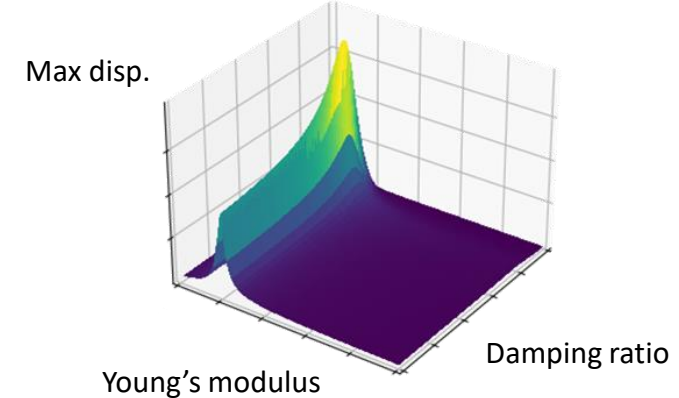
- Response surface representation



One parameter



Two parameters



- Usually the curve (surface) is very flexible & general
Neural networks, Gaussian process model, polynomial chaos...
- **Design of experiments** are used to reduce the number of simulations

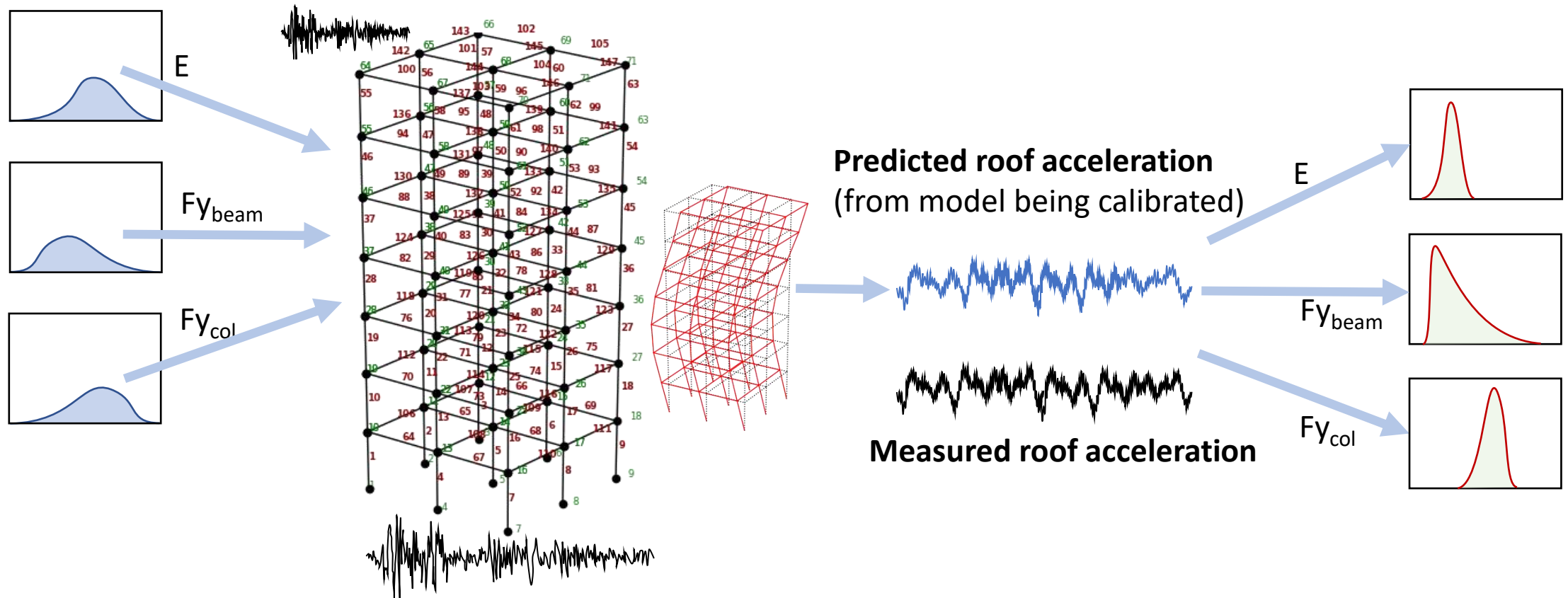
Types of UQ

Forward Uncertainty Quantification

Inverse Uncertainty Quantification

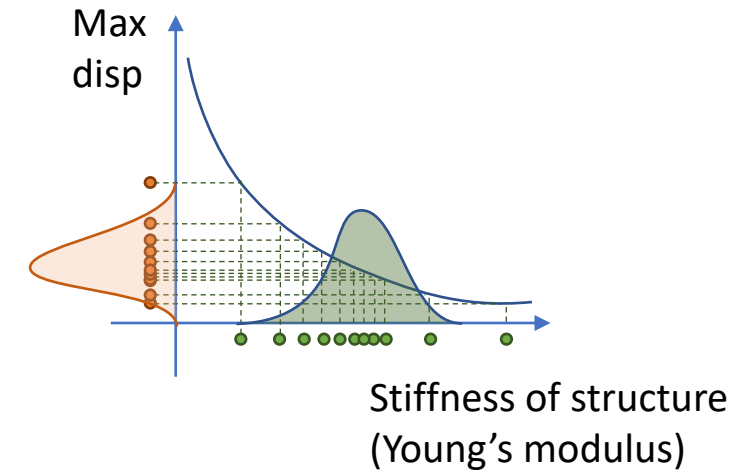
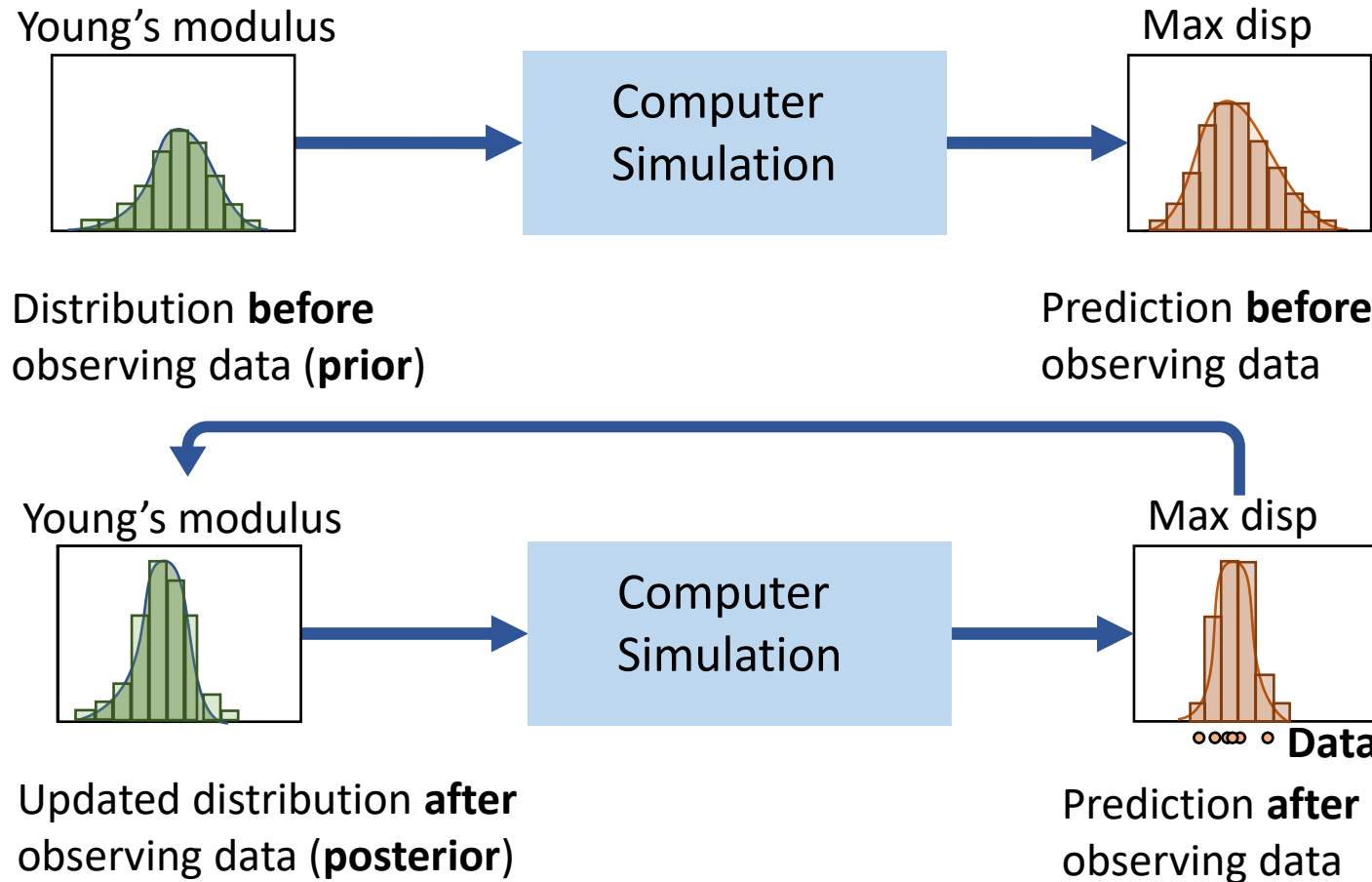
Inverse UQ

- Based on observed data, update the assumptions about the inputs and/or the model



Inverse UQ Methods – Bayesian calibration

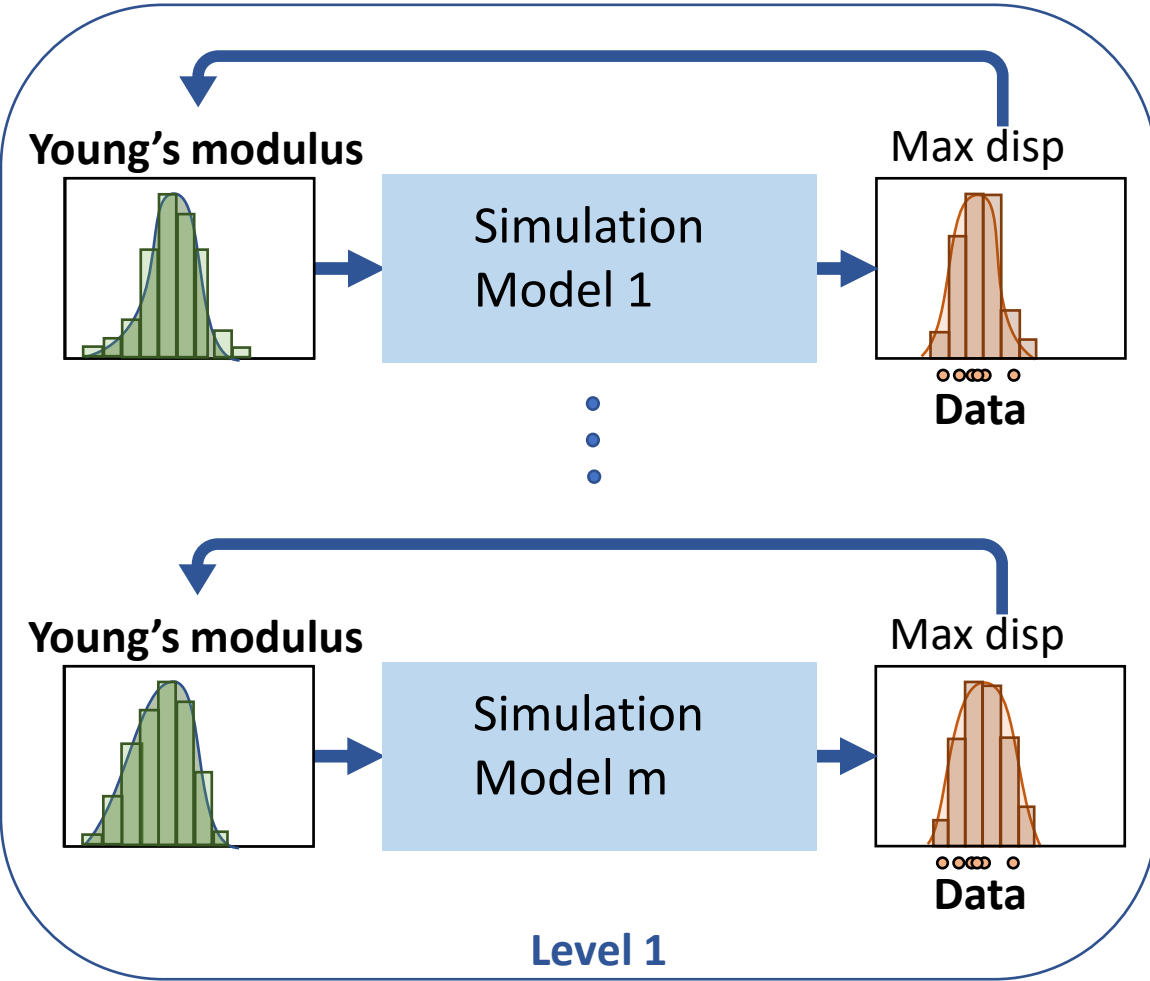
- Based on observed data, update the distribution of the inputs to be consistent with the observations



- Calibration of material model
- Calibration of reduced order model
- Post-event damage diagnosis and prognosis
- Digital twin of structure – real time updating
- Reliability updating

Inverse UQ Methods – Model Class Selection / Averaging

- Based on observed data, update the probability of a set of plausible models



“All models are wrong, but some are useful”

– George E. P. Box

Model 1 probability

⋮

Model m probability

Level 2

- Model parsimony:** if two models fit the data equally well, the simpler model is assigned higher probability
- Model class selection** – select the best model from the set and use for prediction
- Model class averaging** – select all or the best few models, take weighted average of predictions from these models

Running UQ

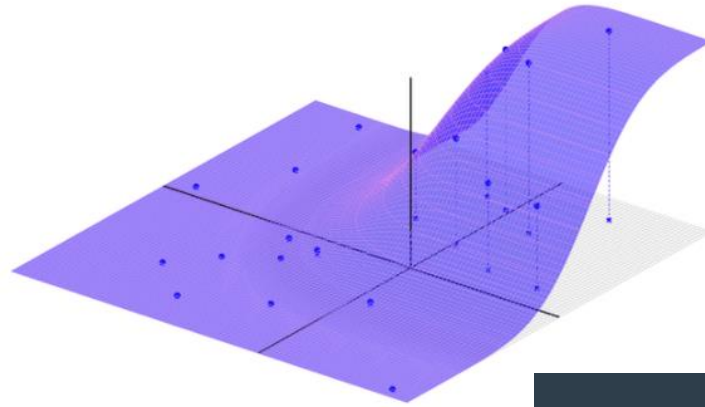
- Toolbox/software packages for UQ analysis

RSUQ PRESENTS
UQ[py]Lab

Uncertainty Quantification
with Python, powered by
UQLab

JOIN THE BETA

Learn More



UQ_{py}

QUO
FEM



MUQ

The MIT Uncertainty Quantification Library

quoFEM

- A software tool with a user interface developed in SimCenter



“You bring the FEM model, we do the rest”

- Need more than what we have?
 - **Build your own quoFEM**
Github page: <https://github.com/NHERI-SimCenter/quoFEM>
 - **Tell us what you need**
SimCenter Forum: <http://simcenter-messageboard.designsafe-ci.org/smf/index.php>

quoFEM (v.2.4)

Simulation (FEM)

Model

OpenSees

FEAPpv

OpenSeesPy

(or any program written in python)

Custom

Surrogate
Model

UQ Type

Sampling

Global
Sensitivity

Reliability

Parameter
Estimation

Bayesian
Calibration

Custom UQ

Surrogate
Modeling

Algorithm

- Latin Hypercube Sampling
- Monte Carlo
- Gaussian Process Regression
- Polynomial Chaos Expansion

- Quasi-Monte Carlo
- Probability-model-based approximation

- Local Reliability (FORM, SORM,..)
- Global Reliability
- Importance Sampling

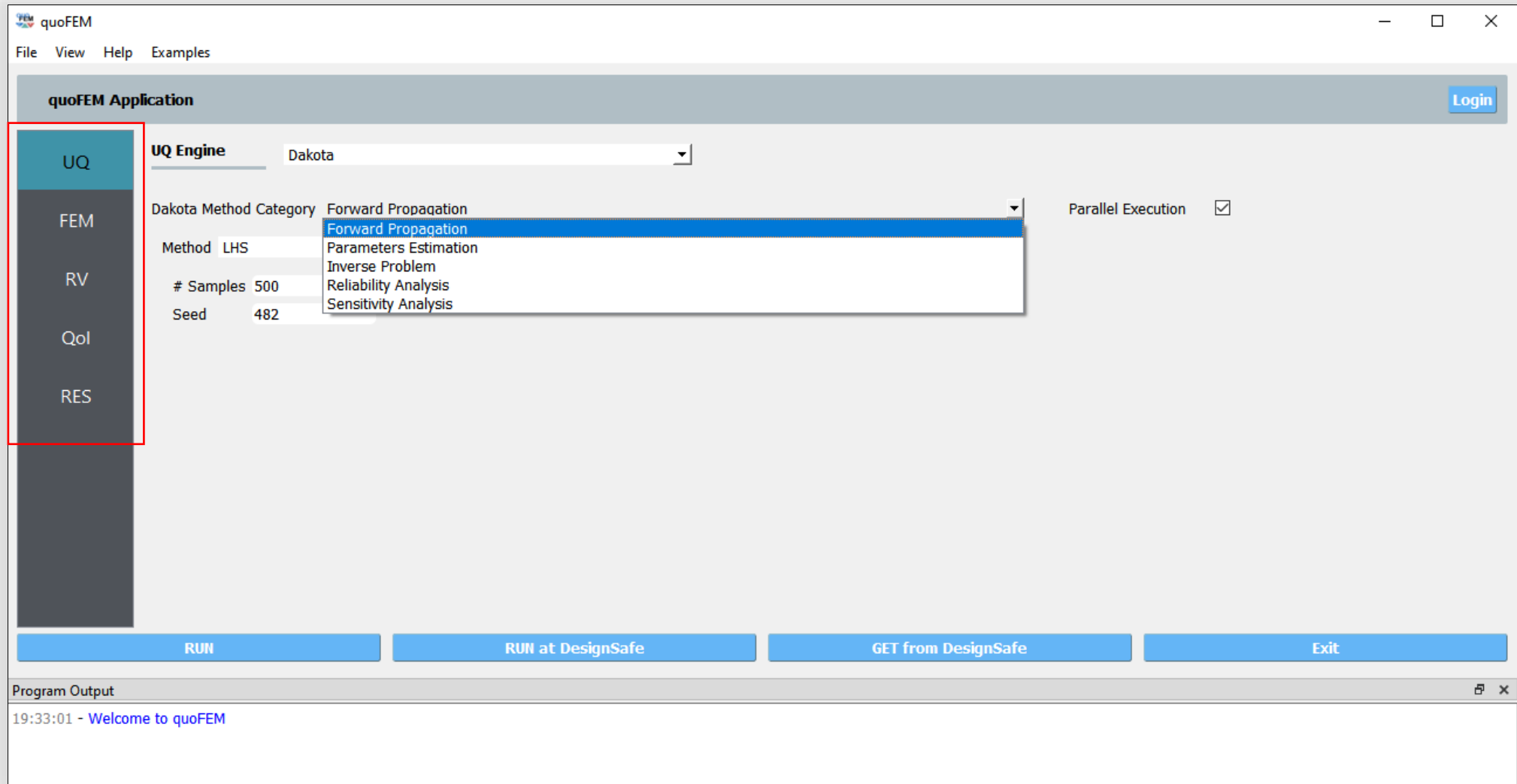
- OPT++GaussNewton
- NL2SOL

- DREAM
- TMCMC

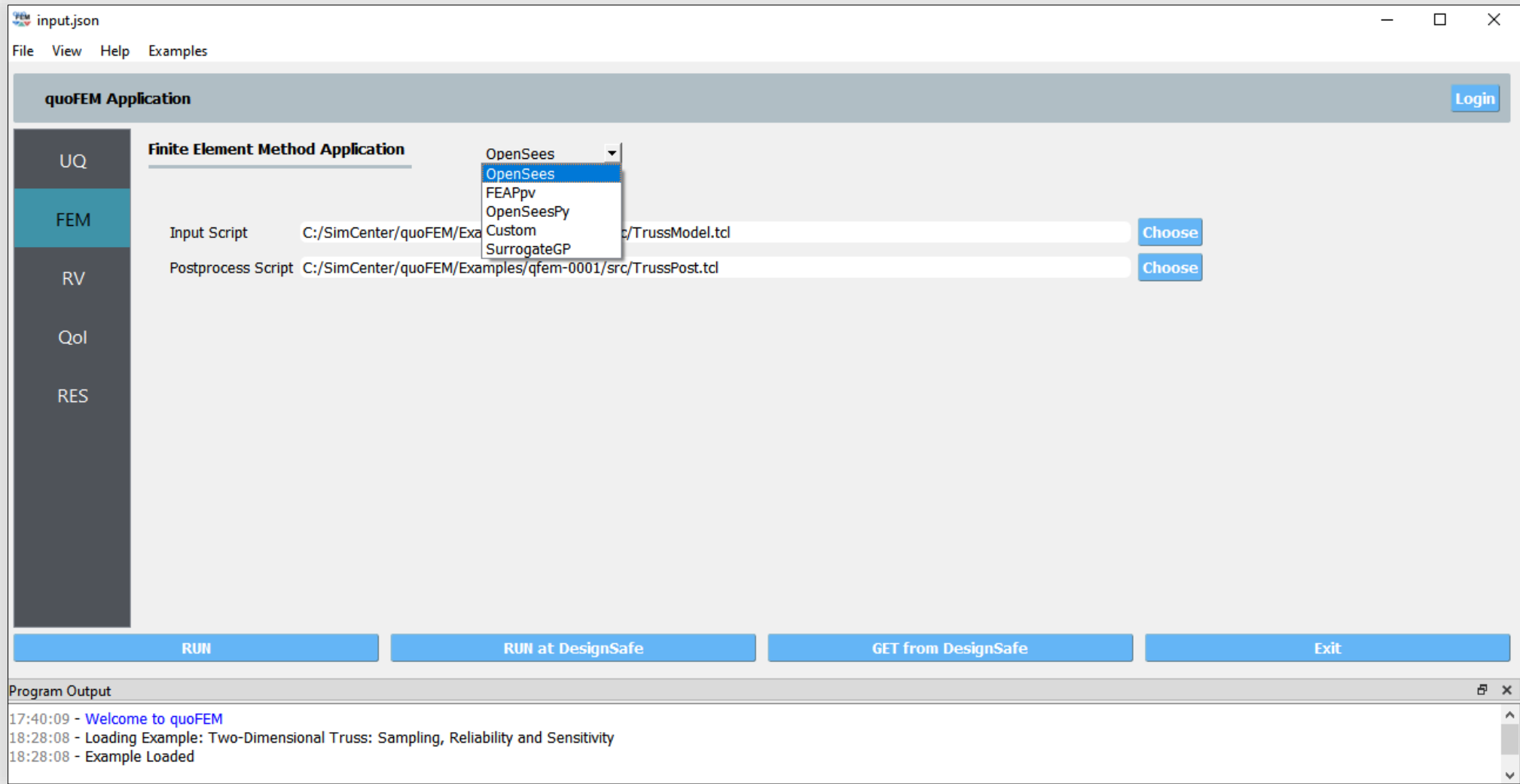
- Custom UQ algorithm

- Gaussian process surrogate modeling
- Gaussian process multi-fidelity modeling

quoFEM User Interface



quoFEM User Interface



quoFEM User Interface

The screenshot displays the quoFEM User Interface. The main window is titled "input.json" and has a menu bar with "File", "View", "Help", and "Examples". The "quoFEM Application" header includes a "Login" button. On the left, a vertical sidebar contains navigation options: "UQ", "FEM", "RV" (highlighted), "QoI", and "RES".

The "Input Random Variables" section features a table with four rows of variables. Each row includes a radio button, a text input for the variable name, a dropdown for the distribution, and input fields for the mean and standard deviation. A "Show PDF" button is associated with each row. Control buttons "Add", "Remove", "Correlation Matrix", "Export", and "Import" are located above the table.

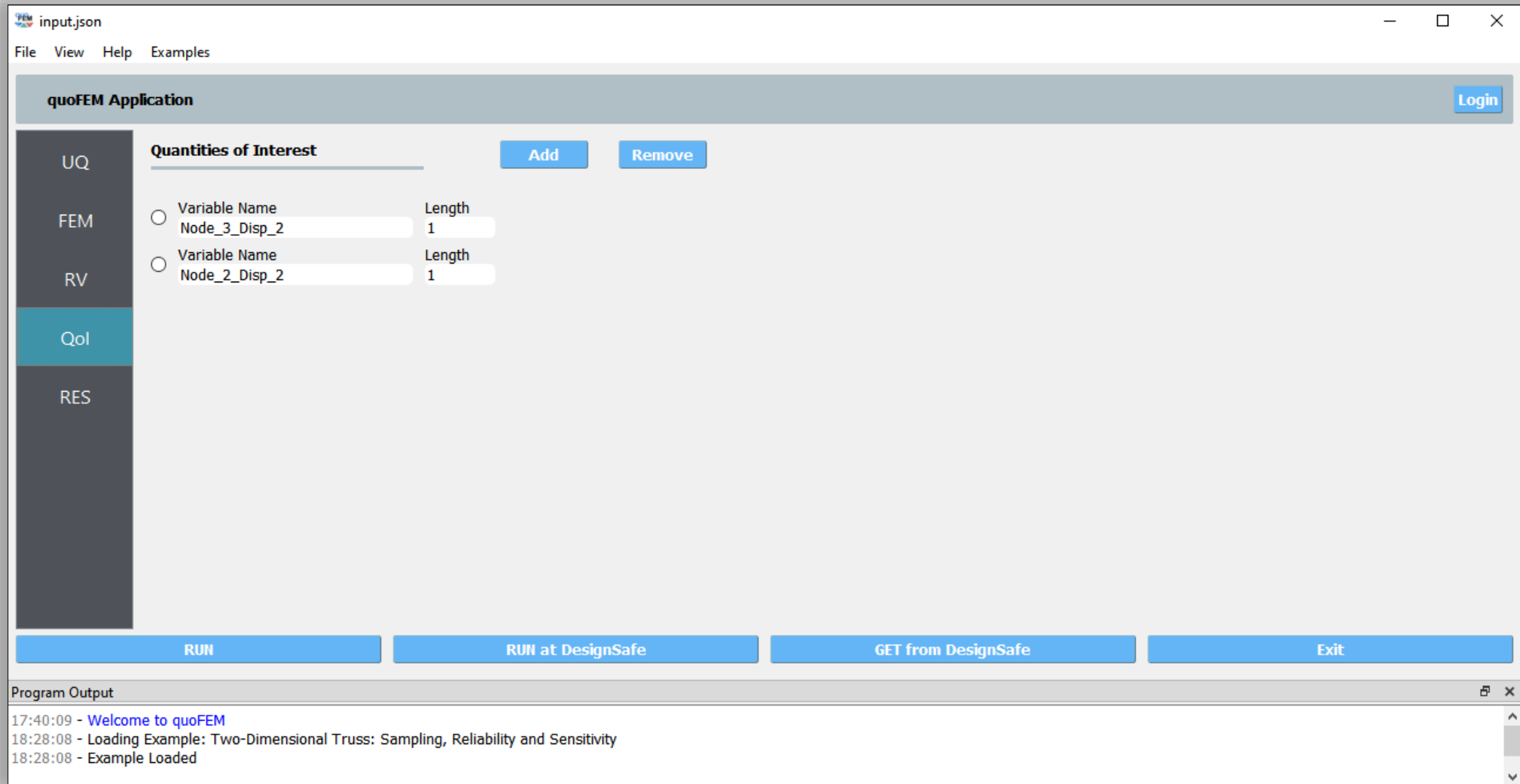
The "Correlation Matrix" dialog box is open, showing a 4x4 matrix for variables E, P, Ao, and Au. The matrix is lower triangular with 1.0 on the diagonal. The correlation between Ao and Au is 0.2.

	E	P	Ao	Au
E	1.0	0.0	0.0	0.0
P	0.0	1.0	0.0	0.0
Ao	0.0	0.0	1.0	0.2
Au	0.0	0.0	0.2	1.0

At the bottom of the main window, there are four buttons: "RUN", "RUN at DesignSafe", "GET from DesignSafe", and "Exit". A "Program Output" window at the bottom shows the following text:

```
17:40:09 - Welcome to quoFEM
18:28:08 - Loading Example: Two-Dimensional Truss: Sampling, Reliability and Sensitivity
18:28:08 - Example Loaded
```

quoFEM User Interface



quoFEM User Interface

The screenshot displays the quoFEM User Interface. The window title is "input.json". The menu bar includes "File", "View", "Help", and "Examples". The main area is titled "quoFEM Application" and has a "Login" button. A sidebar on the left contains navigation options: "UQ", "FEM", "RV", "QoI", and "RES" (which is highlighted). The main content area has two tabs: "Summary" and "Data Values".

Under the "Summary" tab, there are two sections for Sobol' indices:

- Node_3_Disp_2 Sobol' indices:**

Random Variable	Main	Total
P	0.777	0.791
Au	0.025	0.034
E	0.389	0.331
Ao	0.068	0.025
- Node_2_Disp_2 Sobol' indices:**

Random Variable	Main	Total
P	0.779	0.791
Au	0.012	0.018
E	0.394	0.330
Ao	0.090	0.038

Each table is accompanied by a bar chart showing the relative contribution of each variable to the total variance. The y-axis ranges from 0.00 to 1.00. The x-axis lists the variables: P, Au, E, and Ao. The legend indicates that blue bars represent the "Main" effect and green bars represent the "Total" effect.

At the bottom of the main area, there are four buttons: "RUN", "RUN at DesignSafe", "GET from DesignSafe", and "Exit". A "Save Results" button is also present in the bottom right corner of the main area.

The "Program Output" window at the bottom shows the following log entries:

```
11:31:48 - Running Analysis..  
11:32:21 - Processing Results  
11:32:21 - Processing Results ...
```

Conclusion

“An estimate without a standard error is practically meaningless.” (Jeffreys 1967)

- Deterministic result is just one of many **possible** outcomes
 - In order to make decisions, we need to also know how **probable** the outcome is
-
- There are tools available out there to help you apply UQ methods

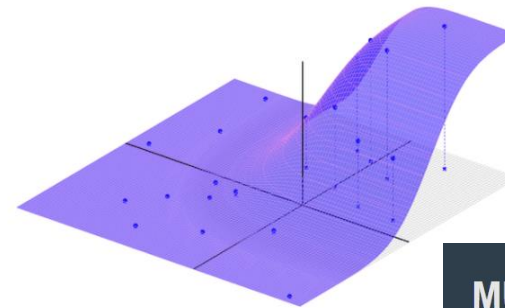
RSUQ PRESENTS

UQ[py]Lab

Uncertainty Quantification
with Python, powered by
UQLab

JOIN THE BETA

Learn More



UQ py

MUQ

The MIT Uncertainty Quantification Library

QUO
FEM



Thank you for your attention!