Intro to Uncertainty Quantification

Forward UQ, basic methods, UM-Bridge, containers

Linus Seelinger

Scientific Computing Center, KIT

Mathematical models

Will this CO2 storage leak in 500 years?
Is this patient's aneurysm critical or not?

Models represent physical processes $\implies$ simulation experiments, digital twins, ...

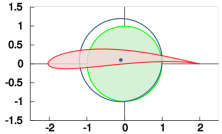
Often expressible as map taking parameter vector to model outcome:

$$F : \mathbb{R}^n \to \mathbb{R}^m$$

Task

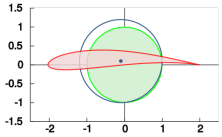
- Execute `model.py`
- While `model.py` is running, explore the model's behavior with `explore_model.ipynb`

Forward UQ



$\underbrace{\theta}_{\text{Parameter}}$

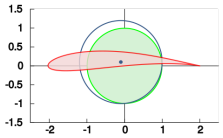
Forward UQ



$\rightarrow$

$$\underbrace{\theta}_{\text{Parameter}} \rightarrow \underbrace{F(\theta)}_{\text{Model outcome}}$$

Forward UQ



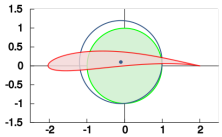
$\rightarrow$

$\rightarrow$

?

$$\underbrace{\theta}_{\text{Parameter}} \rightarrow \underbrace{F(\theta)}_{\text{Model outcome}} \rightarrow \underbrace{Q(F(\theta))}_{\text{Quantity of interest}}$$

Forward UQ



$\rightarrow$

$\rightarrow$

?

$$\underbrace{\theta}_{\text{Parameter}} \rightarrow \underbrace{F(\theta)}_{\text{Model outcome}} \rightarrow \underbrace{Q(F(\theta))}_{\text{Quantity of interest}}$$

Given uncertain parameter's distribution, what's the distribution of the outcome / quantity of interest?

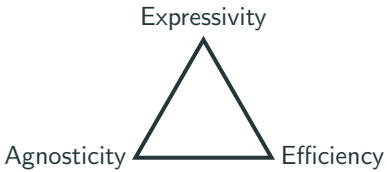
$$\theta \sim \pi. \quad F(\theta) \sim ? \quad Q(F(\theta)) \sim ? \quad \mathbb{E}[Q(F(\theta))] = ?$$

How to solve UQ problems?

Many methods:

- MC / MCMC
- Stochastic Galerkin
- Optimization-based MAP point search
- Multilevel / Multiindex MC / MCMC
- ...

Large-scale problems need HPC!



Monte Carlo Method

- Generate N samples $\theta_i \sim \pi$
- Compute $F(\theta_i)$ for each
- Estimate $\mathbb{E}_\pi[F] \approx \frac{1}{N} \sum_{i=1}^{N} F(\theta_i)$

Task

- Run `forward_uq.ipynb`
- What is the probability of hitting moon?

Efficiency of Monte Carlo

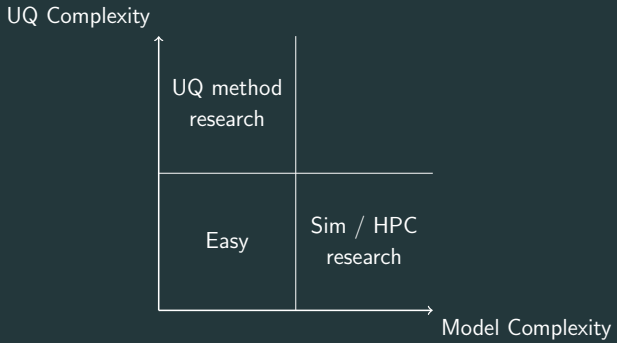
Monte Carlo error - variance of estimator:

$$\begin{aligned}
\mathrm{Var}[\mathbb{E}_\pi[F]] &= \mathrm{Var}\left[\frac{1}{N}\sum_{i=1}^{N} F(\theta_i)\right] \\
&= \frac{1}{N^2}\sum_{i=1}^{N} \mathrm{Var}[F(\theta_i)] \\
&= \frac{1}{N}\mathrm{Var}[F(\theta)]
\end{aligned}$$

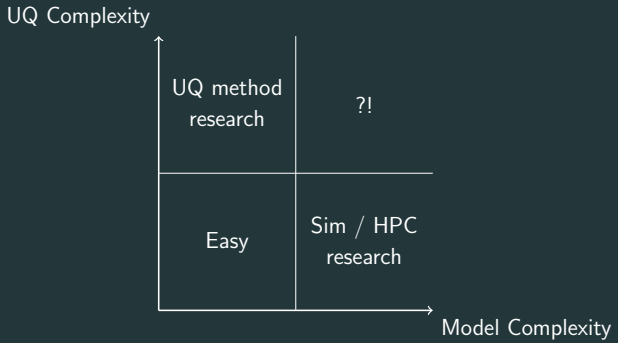
Note: In reality, we only have an approximate model $F_h \sim F$. This introduces a bias.

Linking Data and Simulation

Democratizing Uncertainty Quantification



Democratizing Uncertainty Quantification



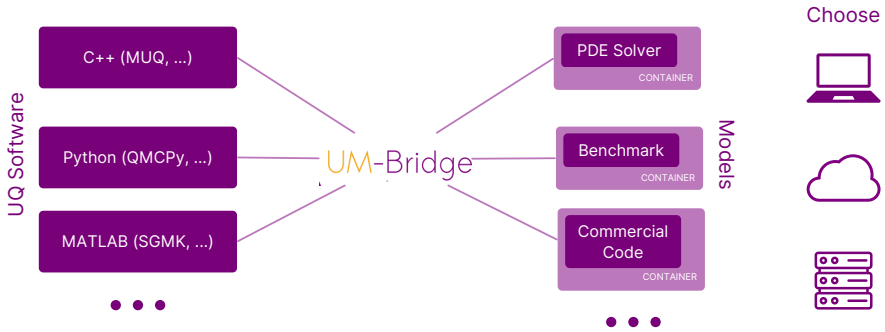
UM-Bridge: Model Abstraction in Software

Interface mimics math: Model provides pointwise

- Evaluation $F(\theta)$,
- Gradient $v^\top J(\theta)$ (optional),
- Jacobian action $J(\theta)v$ (optional),
- Hessian action $H(\theta)v$ (optional).

Inspired by microservices (established in software industry)

UM-Bridge



Break down complexity, enable collaboration

Link *any* UQ code with *any* model, scale to HPC & cloud

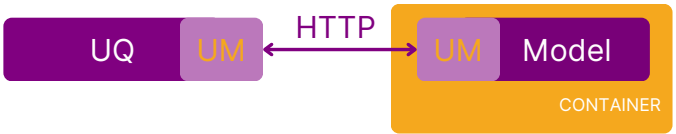
Community-driven benchmark library available now!

Task

- Run `umbridge.ipynb`
- Change input parameters
- Query the model's input dimensions

Containerized Models

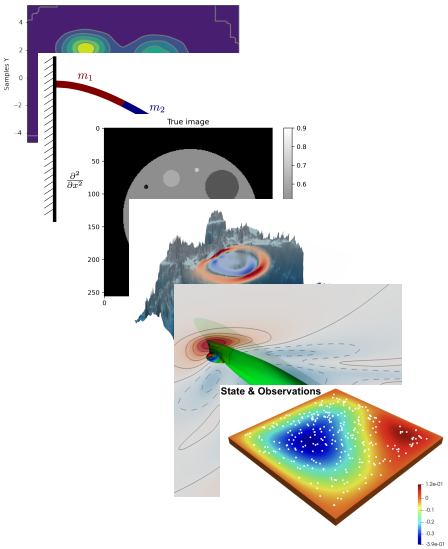
UM-Bridge: Containerization - Portable Models



- Run tsunami model as easy as
  `docker run -p 4242:4242 linusseelinger/model-exahype-tsunami`
- Evaluate model in Python:
  ```
  model = umbridge.HTTPModel('localhost:4242', 'forward')
  model([[0.1,0.4]])
  ```

→ Separation of concerns!

UQ Benchmark Library



- Community project:
  $>$ 20 models and UQ problems,
  $>$ 15 contributors from
  $>$ 10 institutions
- Ready-to-run via UM-Bridge
- Democratizing UQ (arXiv:2402.13768)
  L. Seelinger, A. Reinarz, M. B. Lykkegaard, A. M. A. Alghamdi, D. Aristoff, W. Bangerth, J. Bénézech, M. Diez, K. Frey, J. D. Jakeman, J. S. Jørgensen, K. Kim, M. Martinelli, M. Parno, R. Pellegrini, N. Petra, N. A. B. Riis, K. Rosenfeld, A. Serani, L. Tamellini, U. Villa, T. J. Dodwell, R. Scheichl

Task

- Run xfoil model on your system (or use `https://xfoil.linusseelinger.de`)
  `docker run -p 4242:4242 -it linusseelinger/xfoil`
- Use `umbridge.ipynb` to see how lift responds to flap deflection (set other values to something from the ranges below)
- Modify `forward_uq.ipynb` to solve expected lift for $\theta \sim \mathcal{U}([-0.3, 0.3] \times [492500, 507500] \times [0.225, 0.345] \times [0.637, 0.763] \times [-0.24, 0.24])$

Model inputs: Angle of attack, Reynolds number, upper surface trip location, lower surface trip location, flap deflection

Model outputs: CL (lift), CD (resistance), CDp (resistance due to pressure), CM (angular momentum)

→ Demo: Custom model

→ Demo: Building a container