Intro to Uncertainty Quantification

Backward UQ, Markov chain Monte Carlo, High Performance Computing

Linus Seelinger

Scientific Computing Center, KIT

Backward UQ

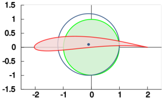
Given uncertain observation, how likely is a parameter / quantity of interest?

$$\underbrace{y}_{\text{Observation}}$$

Backward UQ

Given uncertain observation, how likely is a parameter / quantity of interest?

$$\underbrace{F^{-1}(y)}_{\text{Parameter}} \leftarrow \underbrace{y}_{\text{Observation}}$$

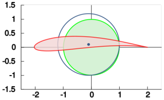


$\leftarrow$

Given uncertain observation, how likely is a parameter / quantity of interest?

$$\underbrace{Q(F^{-1}(y))}_{\text{Quantity of interest}} \leftarrow \underbrace{F^{-1}(y)}_{\text{Parameter}} \leftarrow \underbrace{y}_{\text{Observation}}$$

? $\leftarrow$



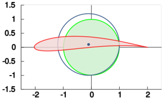
$\leftarrow$

Backward UQ

Given uncertain observation, how likely is a parameter / quantity of interest?

$$\underbrace{Q(F^{-1}(y))}_{\text{Quantity of interest}} \leftarrow \underbrace{F^{-1}(y)}_{\text{Parameter}} \leftarrow \underbrace{y}_{\text{Observation}}$$

? ←



←

Cannot invert F, therefore Bayesian problem!

$$\pi(\theta|y) := \frac{\pi(y|\theta)\pi(\theta)}{\pi(y)} \propto \pi(y|\theta)\pi(\theta).$$

Bayesian posterior

Goal: Given uncertain observation, infer distribution of model parameters.

Use Bayes' theorem to make posterior tractable:

$$\underbrace{\pi(\theta|y)}_{\text{posterior}} \propto \underbrace{\pi(\theta)}_{\text{prior}} \cdot \underbrace{\pi(y|\theta)}_{\text{likelihood}}$$

Prior: Encode prior knowledge about parameter θ

Likelihood: Mismatch between model prediction and observation (often: observation error)

Bayesian posterior

Goal: Given uncertain observation, infer distribution of model parameters.

Use Bayes' theorem to make posterior tractable:

$$\underbrace{\pi(\theta|y)}_{\text{posterior}} \propto \underbrace{\pi(\theta)}_{\text{prior}} \cdot \underbrace{\pi(y|\theta)}_{\text{likelihood}}$$

Prior: Encode prior knowledge about parameter θ

Likelihood: Mismatch between model prediction and observation (often: observation error)

Bayesian posterior with specific distributions

$$\underbrace{\pi(\theta|y)}_{\text{posterior}} \propto \underbrace{\pi(\theta)}_{\text{prior}} \cdot \underbrace{\pi(y|\theta)}_{\text{likelihood}}$$

Let's model prior and likelihood!

Prior knowledge: $\theta \sim \mathcal{N}(\mu_{\text{prior}}, \Sigma_{\text{prior}})$

Observation error: $y = F(\theta) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \Sigma_{\text{obs}})$

This implies Bayesian posterior

$$\begin{aligned}\pi(\theta|y) &\propto \mathcal{N}(\theta; \mu_{\text{prior}}, \Sigma_{\text{prior}}) \cdot \mathcal{N}(y; F(\theta), \Sigma_{\text{obs}}) \\ &\propto \exp\left(-\frac{1}{2}(\theta - \mu_{\text{prior}})^T \Sigma_{\text{prior}}^{-1}(\theta - \mu_{\text{prior}}) - \frac{1}{2}(y - F(\theta))^T \Sigma_{\text{obs}}^{-1}(y - F(\theta))\right)\end{aligned}$$

Bayesian posterior with specific distributions

$$\underbrace{\pi(\theta|y)}_{\text{posterior}} \propto \underbrace{\pi(\theta)}_{\text{prior}} \cdot \underbrace{\pi(y|\theta)}_{\text{likelihood}}$$

Let's model prior and likelihood!

Prior knowledge: $\theta \sim \mathcal{N}(\mu_{\text{prior}}, \Sigma_{\text{prior}})$

Observation error: $y = F(\theta) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \Sigma_{\text{obs}})$

This implies Bayesian posterior

$$\begin{aligned}\pi(\theta|y) &\propto \mathcal{N}(\theta; \mu_{\text{prior}}, \Sigma_{\text{prior}}) \cdot \mathcal{N}(y; F(\theta), \Sigma_{\text{obs}}) \\ &\propto \exp\left(-\frac{1}{2}(\theta - \mu_{\text{prior}})^T \Sigma_{\text{prior}}^{-1} (\theta - \mu_{\text{prior}}) - \frac{1}{2}(y - F(\theta))^T \Sigma_{\text{obs}}^{-1} (y - F(\theta))\right)\end{aligned}$$

Basic MCMC

Initialize the Markov chain with a starting point $\theta^0 \in \mathbb{R}^m$.
for *Sample n = 0, ..., N-1* **do**
  Draw a correlated proposal θ' from proposal distribution $q(\theta'|\theta^n)$.

$$\alpha(\theta'|\theta^n) := \min\left\{1, \frac{\pi(\theta')q(\theta^n|\theta')}{\pi(\theta^n)q(\theta'|\theta^n)}\right\}.$$

  Draw a random number $r \in [0, 1]$.
  if $r < \alpha(\theta'_k, \theta^j_k)$ **then**
    Accept proposal: $\theta^{n+1} := \theta'$
  else
    Reject proposal: $\theta^{n+1} := \theta^n$
  end
end

Algorithm 1: Metropolis-Hastings MCMC

- Evaluate target density at finite num. of points
- Result: Markov chain with stat. distribution π

MCMC Demo

`https://chi-feng.github.io/mcmc-demo/app.html`

Exercise

Task

- Run `backward_uq.ipynb`. What does the output tell you?
- The model server also provides a posterior `posterior2`. What's different?

Advanced UQ

Basic Idea of Multilevel Methods

A380: Roger Green, A319: A.Savin, WikiCommons

Basic Idea of Multilevel Methods

$>$

$>$


$>$


Do most of the work on cheap "coarse" models!

Links to multilevel PDE solvers, multiscale methods...

A380: Roger Green, A319: A.Savin, WikiCommons

Multilevel MCMC Overview

Telescoping sum of QOI like MLMC:

$$\mathbb{E}_{\nu^L}[Q_L] = \underbrace{\mathbb{E}_{\nu^0}[Q_0]}_{\text{Coarse approx.}} + \sum_{l=1}^{L} \underbrace{\left(\mathbb{E}_{\nu^l}[Q_l] - \mathbb{E}_{\nu^{l-1}}[Q_{l-1}]\right)}_{\text{Corrections}}.$$

How to sample?

High acceptance rates due to samples from coarser levels

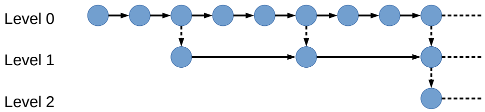
Coarser models cheap, low variance in finer corrections!

Multilevel MCMC Overview

Telescoping sum of QOI like MLMC:

$$\mathbb{E}_{\nu^L}[Q_L] = \underbrace{\mathbb{E}_{\nu^0}[Q_0]}_{\text{Coarse approx.}} + \sum_{l=1}^{L} \underbrace{\left(\mathbb{E}_{\nu^l}[Q_l] - \mathbb{E}_{\nu^{l-1}}[Q_{l-1}]\right)}_{\text{Corrections}}.$$

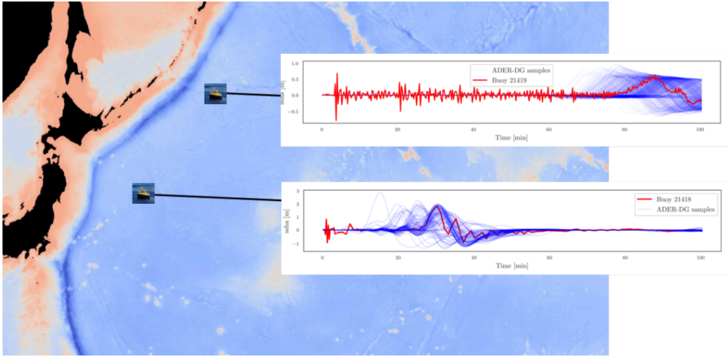
How to sample?



High acceptance rates due to samples from coarser levels

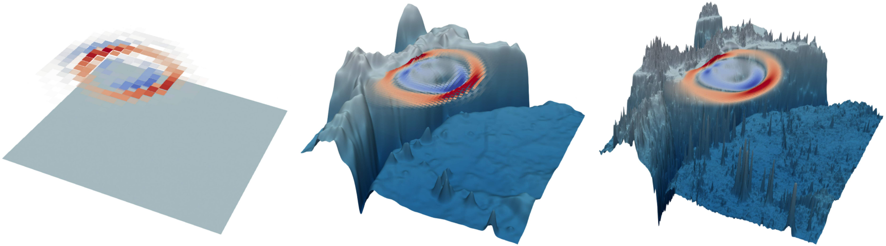
Coarser models cheap, low variance in finer corrections!

Tsunami Problem



- Modelling Tohoku event (2011) using Shallow Water eq. and real bathymetry
- Forward model built on ExaHyPE PDE engine
- Data: Buoy measurements. Parameter: Tsunami source

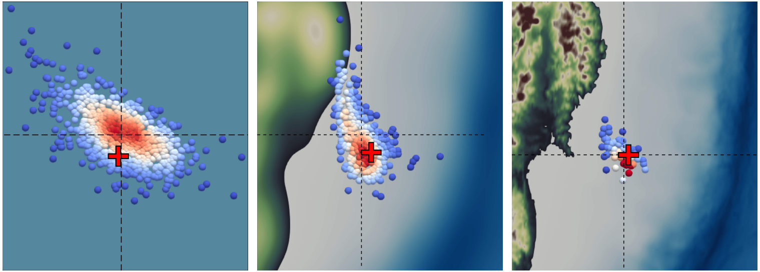
Model Hierarchy



Across levels, we adapt

- mesh size
- bathymetry smoothness (specific to hyperbolic solvers!)

Results

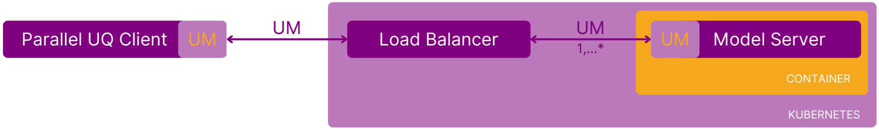


lvl l	t_l[s]	ρ_l	$\mathbb{V}[Q_0]$ or $\mathbb{V}[Q_l - Q_{l-1}]$		$\mathbb{E}[Q_0]+$ $\sum_{k=1}^{l} \mathbb{E}[Q_k - Q_{k-1}]$	
0	7.38	25	1984.09	1337.42	3.61	27.96
1	97.3	5	1592.17	1523.18	−12.29	23.39
2	438.1	0	340.56	938.53	−5.46	0.12

Run on 3456 cores (72 nodes of 48 cores)

Scaling up via UM-Bridge

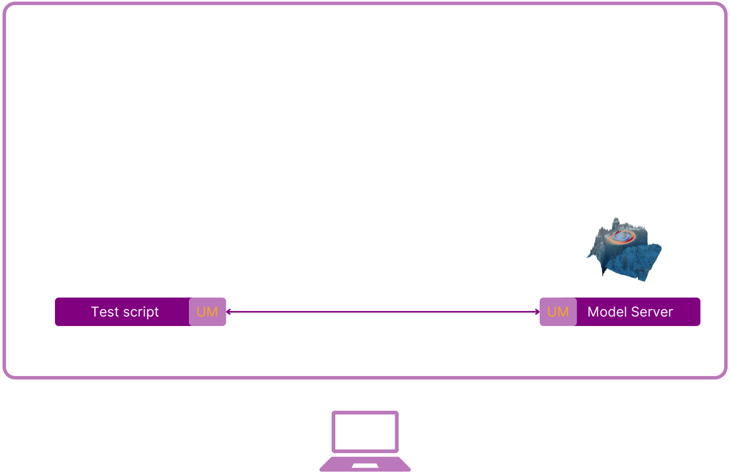
Kubernetes - Sequential Model



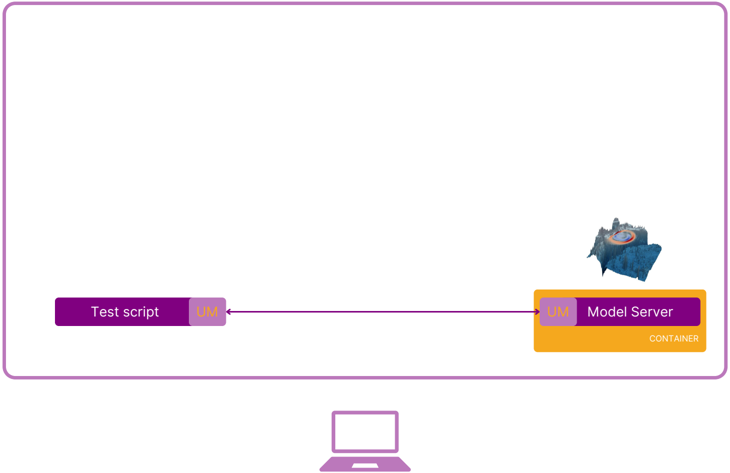
Pre-built configuration, simply plug in your own model container

UQ client only sees an UM-Bridge server. But may make parallel requests!

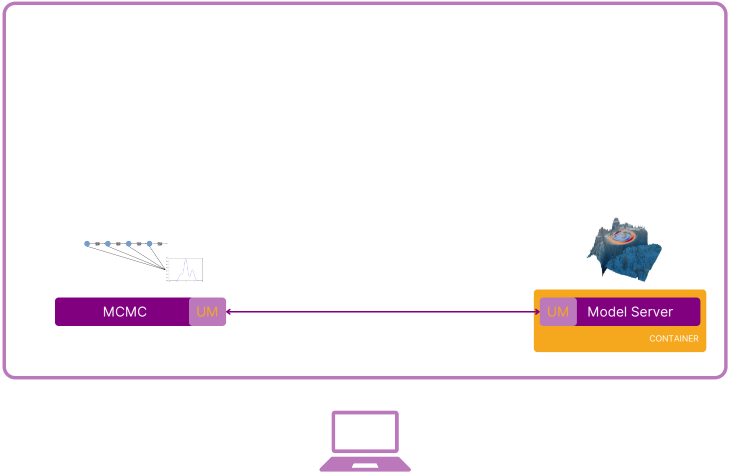
Workflow



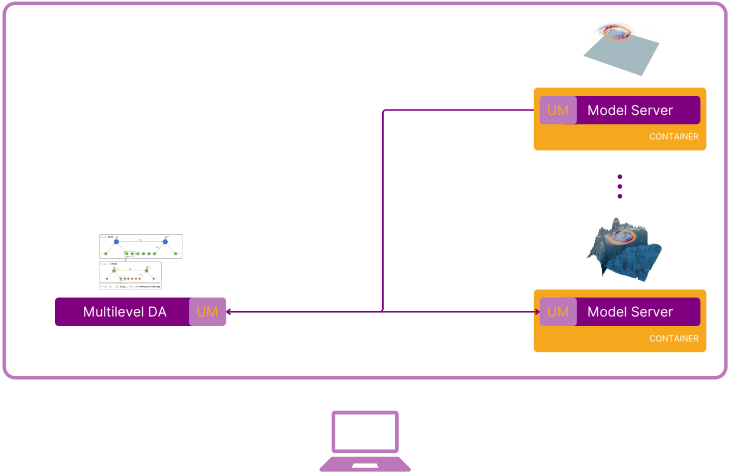
Workflow



Workflow

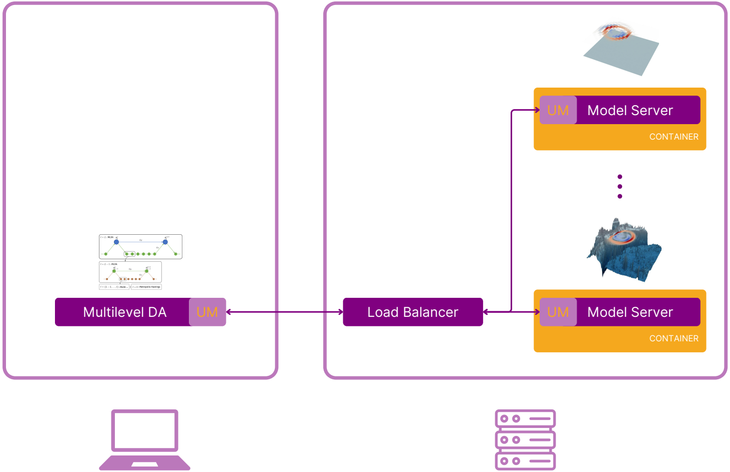


Workflow

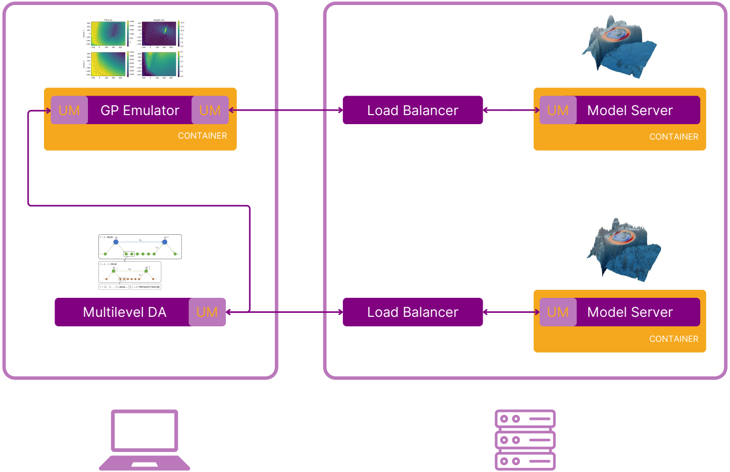


Model Server
UM
CONTAINER
Model Server
UM
CONTAINER
Multilevel DA
UM

Workflow


Multilevel DA
UM
Load Balancer
UM
Model Server
CONTAINER
UM
Model Server
CONTAINER

Workflow



Successfully run on 2800 core Google Kubernetes Engine cluster

Open end

Interesting points to discuss?

Task ideas

- Write your own model following UM-Bridge documentation examples
- Try a different observation likelihood in backward UQ (edit `model.py`)
- Try and solve the L2-Sea forward UQ benchmark (see UM-Bridge docs) with QMCpy

Upcoming UM-Bridge workshop:
`https://um-bridge.github.io/workshop/`