ExCALIBUR 10

NEPTUNE DOCUMENTS REPOSITORY

AND

UQ FOR NEPTUNE

Joseph Parker

28th January 2021

UK Atomic Energy Authority

Introduction

- Neptune Documents Repository
  - UKAEA ExCALIBUR task reports
- Uncertainty Quantification report
- VECMA Hackathon work with BOUT++ and EasyVVUQ

Neptune Repositories

github.com/ExCALIBUR-NEPTUNE/Neptune

github.com/ExCALIBUR-NEPTUNE/Documents

drive.google.com

> Shared Drives on sidebar > ExCALIBUR-Neptune

Shared Google Drive - quick, private

Two Github repositories:

- *Neptune* - main software repo
  - software, indirectly as git submodules
  - project-level software documentation
- *Documents* - collection of useful documents for collaborators
  - Project documentation (e.g. bid documents, Neptune Charter)
  - Neptune-generated documents (e.g. presentations, reports)

Publicly available

- *Documents* is still private, but all should have access

Owned by UKAEA, admins Alys Brett, James Cook, Peter Hill

Neptune Documents Repository

- Current contents
  - Bid documents (bid_documents/)
  - Call documents (tex/call_<n>)
  - Equations document (tex/equations_for_neptune_proxyapps)
  - Neptune Charter (tex/neptune_charter)
  - Science plan (tex/science_plan)
  - Assorted notes (tex/note_on_finite_elements, tex/software_development_guide)
  - UKAEA reports (tex/t<n>)

to build documents in tex/, cd to the relevant directory and do make

- Contribution via Pull Request
  - Curating repo contents, ensuring quality, preventing mistakes
  - Help is available! Best ask on Slack

UKAEA Task Reports

Reports for Task N.M are in the folder `tex/t<NM>`

- t12 - Year 1 summary
- t21 - Options for **geometry** representation
- t23 - Options for **particle algorithms**
- t31 - User **frameworks** for tokamak multiphysics
- t31 - User layer design for **uncertainty quantification**
- t33 - **Design patterns** specifications and prototypes
- t33 - **Design patterns** evaluation

UQ Report

Written before/during call specification process

UKAEA staff getting up to speed with the state of the field

Long (~30 pages) but mostly short digestible sections

1. Description of mathematical basis for UQ techniques

    - Global sensitivity analysis (e.g. polynomial chaos, sparse linear regression, multifidelity Monte-Carlo, multi-level Monte-Carlo)
    - Producing surrogates (e.g. sparse quadrature sampling, forward UQ, Sobol indices)
    - Use of surrogates (e.g. Gaussian process regression, artificial neural networks, reduced order models)
    - Optimisation Under Uncertainty

2. Outline of how we think UQ might be implemented in a NEPTUNE workflow

Possible Neptune Workflow

1. **Global Sensitivity Analysis** - reduce number of important uncertain parameters
2. **Optimisation Under Uncertainty** loop
    a. **Forward UQ** - information about Quantities of Interest
    b. **Inverse UQ** - information about design parameters

OOU loop is driven by a “simulator”, a surrogate model

e.g. 3D reduced fluid model instead of 5D or 6D kinetic model

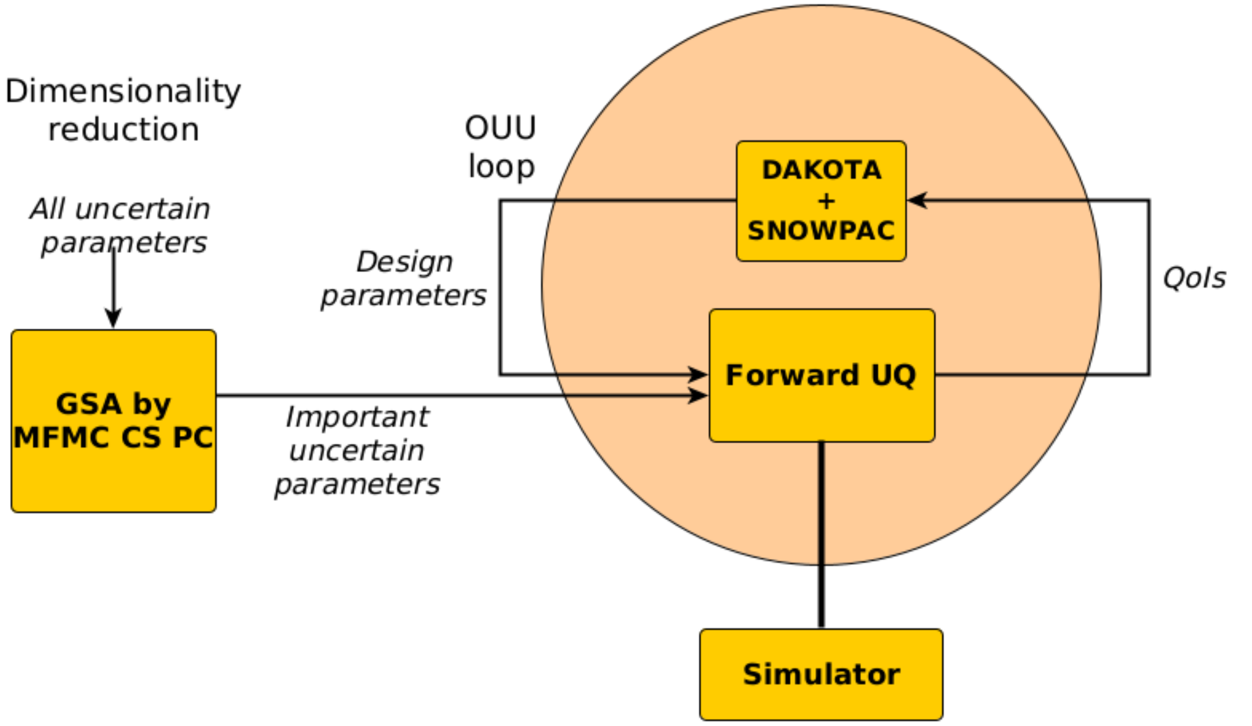


Figure from H. Najm. Uncertainty Quantification in Computational Models of Large Scale Physical Systems. https://www.osti.gov/servlets/purl/1593073, 2018. Seminar, NRC Institute of Marine Engineering, Rome, Italy.

BOUT++ with EasyVVUQ: Update from the VECMA Hackathon

- BOUT++ working with EasyVVUQ for a simple 1D test case
  - Written custom encoders/decoders (i.e. the interface between BOUT++ and EasyVVUQ)
  - Got simple statistics, error bars, Sobols indices, etc.
- 2D plasma filament propagation (10 mins on 16 cores for each parameter case)
  - Running in parallel on York's cluster
  - Different decoder / data processing work added to workflow
  - 3rd order polynomial chaos would have required 1296 case
  - Varying 4 parameters, using stochastic collocation needed 256 cases

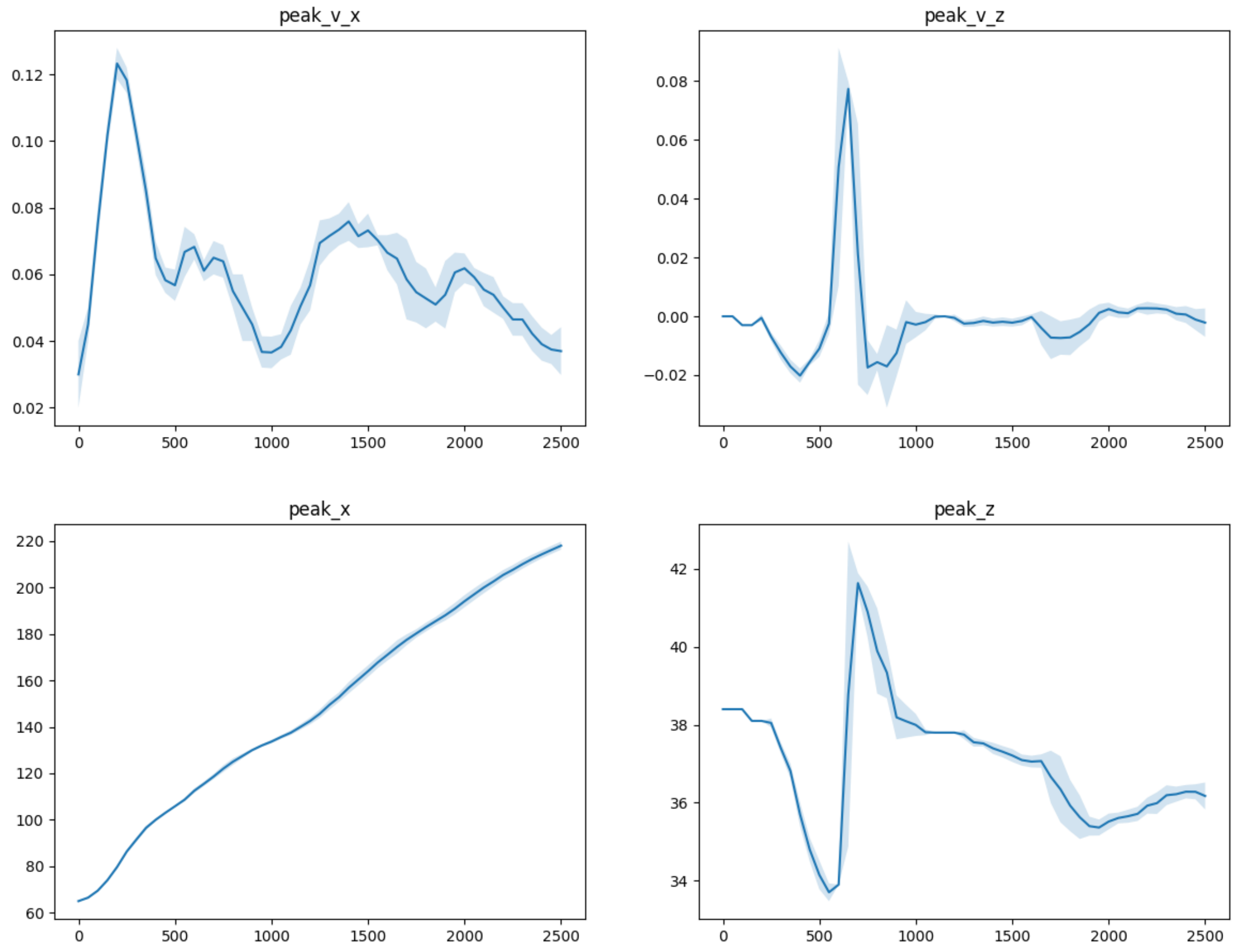
To do:

- Investigate more realistic 3D models
- Investigate EasySurrogate and FabSim3

BOUT++ Team:
Peter Hill
Ben Dudson
David Dickinson
Joseph Parker

2D filament propagation

- Model for time evolution of density perturbation in 2D plasma
- Varying 4 parameters: background density, temperature, and 2 dissipation parameters



horizontal axis: time
(top) velocity of perturbation peak
(bottom) position of perturbation peak
(left) x-coordinate, (right) z-coordinate

mean and 10th to 90th percentile

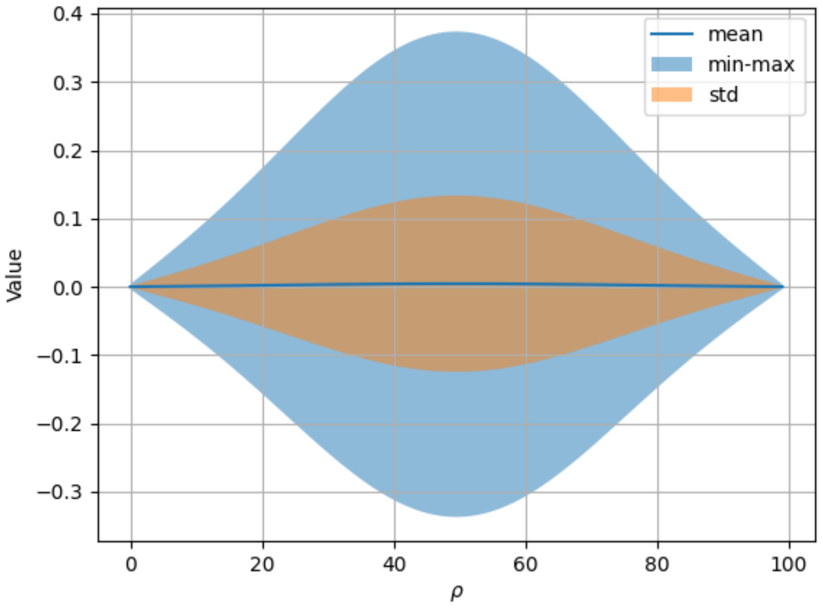
Problems

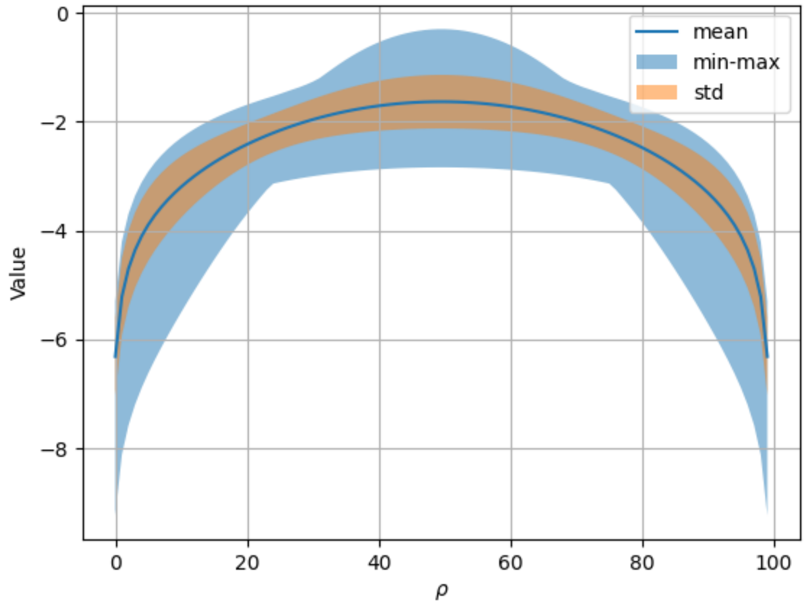
2D filament propagation

- Differences in outputs for SC and PCE
  - SC less developed than PCE, some discrepancies, e.g. SC outputs a single Sobol index not an array.

1D heat conduction

- Found negative temperatures from fitting polynomials to near-zero solutions
  - “Fixed” by solving for log(T), but can we implement constraints?





Thoughts and Questions

- This is going to be expensive! Want to vary many more than 4 parameters, and want to use much more detailed models.
- Is intrusive UQ any cheaper? Intuitively, it seems that intrusive UQ wraps the work for a whole campaign into a single run. Non-intrusive lets us break up a campaign into smaller pieces, explore parameter space in more adaptive ways.
- How do we go about constructing surrogates? Should we be sharing surrogates with the preconditioning project?
- How useful is UQ for models that we know are missing physical effects? Missing terms is a qualitative change.
- Of what should we be measuring the uncertainty? Physical quantities? Perhaps we better cross-code comparisons from measuring synthetic diagnostics?