Report number:

2047352_1-TN-02

Title:

Recommendations for enabling the uncertainty quantification of future multiscale fusion modelling code

Authors:

Maxime Vassaux, Wouter Edeling, Peter V. Coveney

Executive summary:

Following our initial report on existing uncertainty quantification methods suited for single-scale fusion models and generic multiscale workflows, the present document reports on the process of their implementation and application to the building blocks of the future NEPTUNE fusion multiscale model. The extensive collaboration with the EU-funded VECMA project enabled rapid prototyping of non-intrusive uncertainty quantification all individual single-scale model applications involved in the NEPTUNE project (BOUT++, Nektar++, Particle-in-Cell method). The report details the collaboration process via multiple hackathons and the benefits drawn from the mutual feedback between VECMA and NEPTUNE developers. Drawing from collaboration with model-order reduction experts from the NEPTUNE consortium, further recommendations and associated prospective developments toward accelerated uncertainty quantification based on surrogate models are finally provided. These advanced methods will be key to quantifying uncertainties in the final multiscale coupled-codes model.

Table of contents

1 Introduction

It is widely recognised that uncertainty quantification (UQ) is essential for any model simulation to be actionable, i.e., to have genuine predictive value for decision-making purposes. Ensemble-based methods lie at the heart of techniques enabling the evaluation of the uncertainty associated with imperfectly known model parameters. For low dimensional toy models, many of these methods work very well, but when it comes to realistic, larger-scale problems, they are subject to the curse of dimensionality, which implies that the required number of model evaluations scale exponentially with the number of uncertain parameters. As a result, UQ can be intrinsically computationally very expensive in the case of high-dimensional input spaces. Nevertheless, it is often possible to accelerate UQ and reduce the computational cost in comparison to brute force (Quasi) Monte Carlo methods which exhibit slow convergence. Their efficiency depends on the model equations, the simulation workflow, and the number of parameters in play.

We investigate the structure of the several single-scale codes which will constitute the future NEPTUNE multiscale code and provide recommendations for integrating VVUQ. In the NEPTUNE project, the models and equations are already specified. We perform a shortlisting of suitable UQ methods for the NEPTUNE code in the present project. We are familiar with such codes both from personal experience with chaotic systems [1] and our longstanding collaboration with MPG-IPP. During the first attempt to perform UQ on NEPTUNE single-scale models (BOUT++, Nektar++ and EPOCH), we gained further expertise with these models. The major development needed was to produce efficient surrogate modelling methods. In particular, the BOUT++ application team faced limitations due to sampling point selection with already available stochastic collocation and polynomial chaos expansion surrogate models. Similar limitations were faced by the Nektar++ application team, in the case of chaotic time series surrogate modelling.

The project drew substantially on the VECMA project, led by the PI (PVC), which has been developing tools for the VECMA toolkit and provides training to enable new users to perform the rapid implementation of VVUQ prototypes as well as robust solutions. Details about resources offered by and interactions with the VECMA project are introduced in the second section. Expertise, software and support from VECMA have been key for testing and implementation of UQ throughout the course of this seven-month project. With such time constraints, we have chosen to train NEPTUNE application teams to implement and prototype their own code UQ. We summarise findings from the NEPTUNE application teams (BOUT++, Nektar++ and EPOCH) in the third section. The teams constitute a testing panel of the methods available in the VECMA toolkit. Based on their feedback from applying the VECMA toolkit to their single-scale codes, we provide final recommendations for pursuing the development of UQ software for the future multiscale fusion code.

Meanwhile, in conjunction with the model-order reduction (MOR) team, prospective theory and algorithms from parametric UQ and surrogate modelling are investigated, among which are active subspaces and advanced Gaussian process (GP) methods. We also discuss prospective methods to enhance coupling between single-scale models. These methods are introduced in the fourth and last section of this report.

2 Interacting with VECMA: hackathons and initial UQ use case applications

2.1 The project and the toolkit

The primary aim of the VECMA project is to enable a diverse set of multiscale, multiphysics applications to run on current multi-petascale computers and emerging exascale environments with high fidelity such that their output is actionable. The central deliverable is an open-source toolkit for multiscale VVUQ based on generic multiscale VV and UQ primitives, fully tested and evaluated in emerging exascale environments, actively promoted over the lifetime of this project, and made available in European HPC centres.

The full VECMA toolkit (VECMAtk) therefore establishes a platform for VVUQ featuring various in-built interfaces with HPC machines. One already available tool for forward uncertainty propagation is EasyVVUQ, which currently contains the following sampling methods:

- (Quasi) Monte-Carlo (QMC) methods,
- Polynomial Chaos Expansion (PCE) and Stochastic Collocation (SC) methods, both with full-tensor grid and isotropic sparse grid constructions,
- Dimension-adaptive Stochastic Collocation,
- GP surrogates.

QMC methods do not suffer from the curse of dimensionality but have a slow (at most linear) convergence rate. On the other hand, PC and SC exhibit exponential convergence, but only for a small number of inputs and smooth functions. Their cost typically scales as $\mathcal{O}(N^d)$ with *N* being the number of code evaluations per input parameter, and *d* the number of parameters. In the case of isotropic sparse grids, the exponent *d* is applied to $log(N)$ [2] (in the error estimate), thereby postponing, yet not eliminating, the curse of dimensionality. Via dimension-adaptive SC, the cost can be reduced further by finding the `effective dimension' in an iterative fashion. This is the subset of the stochastic input space responsible for most of the output variance. We have applied this to a well-known epidemiological model featuring more than 900 parameters [3]. By means of sensitivity analysis (SA) and adaptive sampling we have been able to find the parameters which dominate the variance in the code's output data; in fact, over 60% of the variance in the model's behaviour is attributable to a mere three parameters. A similar, but much more computationally demanding study on the impact of random number generators in molecular-dynamics simulations was also recently reported by us [4].

All methods can estimate the output uncertainty, as well as provide (global) sensitivity estimates on the input parameters. These methods were made fully available alongside training support to all NEPTUNE application teams. In turn, the toolkit has also been applied to simulations of BOUT++, Nektar++ and EPOCH during hackathons as we discuss in the next section.

2.2 Expertise in fusion modelling

Our expertise in generic UQ methods and software was supported by the expertise in plasma fusion modelling from colleagues at the Max Planck Institute for Plasma Physics (IPP) who have been working with us during the VECMA project. IPP is one of the largest fusion research centres in Europe, where the main goal is to investigate the physical basis of fusion reactions used as a new energy production source. The Institute coordinates leading expertise on both experimental and theoretical plasma physics and drives the development of some of the most advanced simulation codes in this field. IPP is also involved in the Integrated Modelling activities of EUROfusion, which develops a simulation platform composed of a generic set of tools for modelling an entire tokamak experiment. IPP has implemented a multiscale fusion model of interest to the current goals of NEPTUNE. In order to bridge the space and time scales, multi-scale simulations have been performed coupling a gyrofluid turbulence code, GEM [5],

to a transport code that describes the evolution of the profiles [6], using the MUSCLE2 toolkit (see figure 1). In the ComPat project (preceding VECMA), the gyrofluid description of the turbulence was replaced with a gyrokinetic description (5D, 3 of physical space and 2 of velocity space). Two different gyrokinetic codes are currently used: NEMORB and delta-FEFI. The particle-in-cell code ORB5 [7] is a global code for tokamak turbulence and is able to provide fully resolved electrostatic ITER simulations on present-day supercomputers, using 8k cores at near-optimal efficiency. ORB5 has recently been upgraded to include self-consistent perpendicular magnetic field perturbations. This code, named NEMORB [8], supports multiple ion species and electromagnetic turbulence studies. When using kinetic electrons for electromagnetic effects, the simulations become much more challenging, as the particle count is a factor 50 larger (400k cores, ~8TB), the time step size at least a factor 20 smaller, and the solver more expensive (two matrices to solve). The delta-FEFI code [9] is a turbulence simulation code solving the gyrokinetic equations on a grid of phase space. It is based on a fluxtube model that treats the intrinsic turbulence at one point on a larger profile. Obtaining the turbulence saturation in statistical saturation with delta-FEFI requires a 100-hour run, simulating 1 million steps on 10k cores, for one fluxtube instance.

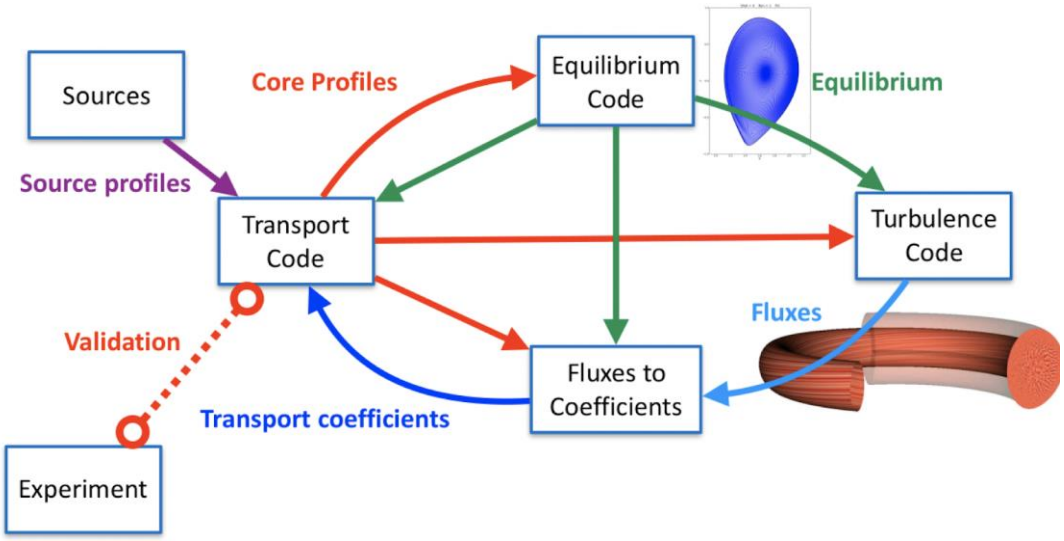

Figure 1: IPP multiscale fusion model. Schematic of the coupled models within the plasma fusion simulation workflow.

In the VECMA project, the propagation of uncertainties has been included in this fusion simulation workflow (see figure 2) [10]. EasyVVUQ, QCG-PJ, and MUSCLE3 were used in conjunction to perform this task. The turbulence code was used to provide the transport coefficients, but the output from the turbulence code was inherently “noisy” and more effort is required to extract as much information as possible from this noisy signal, going beyond averaging to try to quantify the uncertainty interval. These uncertainty intervals in the transport coefficients were then propagated through the transport code to produce an uncertainty interval in the calculated profiles, which would then be used in the turbulence code to calculate new uncertainty intervals. We performed these calculations non-intrusively using the gyro-fluid code, but we also sought more intelligent approaches than launching thousands of gyrokinetic simulations. Other sources of uncertainties were also quantified and propagated: the transport code uses

externally provided sources whose uncertainties also needed to be estimated and their effect taken into account.

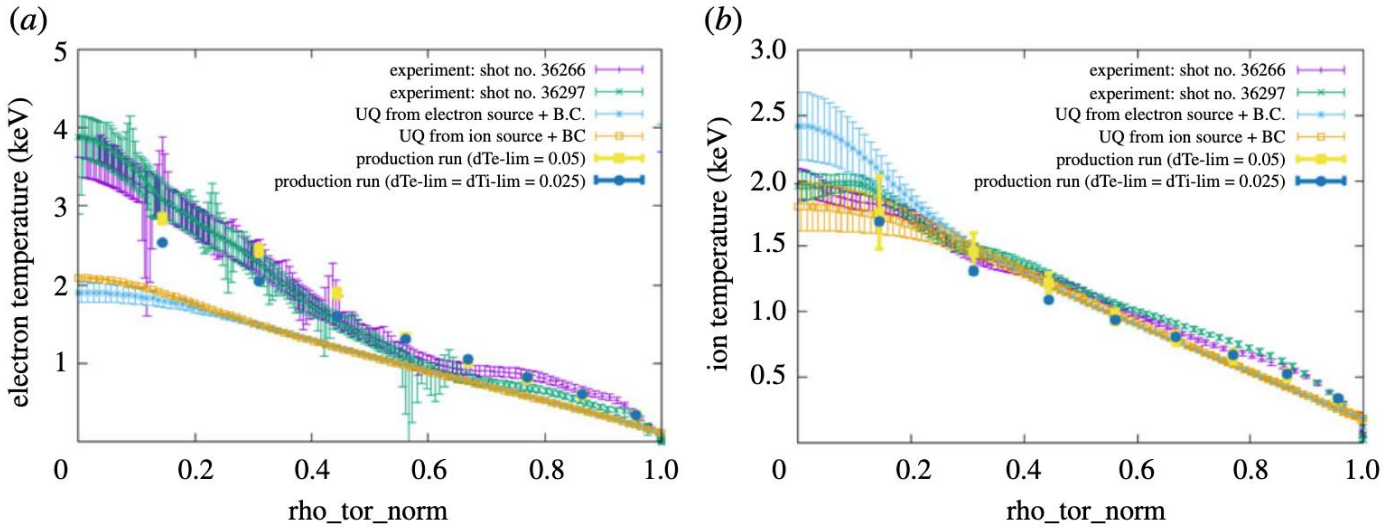

Figure 2: Electron Te (a) and ion Ti (b) temperature profiles measured from two production simulations (with different adaptive time step parameters: in thick yellow lines with filled-square markers and in thick blue lines with filled-circle markers), 2 UQ simulations (one with uncertainties coming from the electron heat source and Te boundary condition plotted in blue with asterisk markers; and the other with uncertainties coming from the ion heat source and Ti boundary condition plotted in gold with outlined-square markers), and experiments (shot no. 36266 in purple lines with plus markers and no. 36297 in green lines with × markers). The temperatures are plotted against normalized toroidal flux coordinate 'rho_tor_norm' (figure retrieved from [10]).

2.3 Hackathons

Hackathons constitute one of the ways of disseminating knowledge about methods and technology developed within the VECMA project. Here, hackathons were set up to enable prototyping of UQ for NEPTUNE applications, by application teams directly, with support from the toolkit developers. Hackathons enabled hands-on use of the existing toolkit but also consideration of the actual design of UQ campaigns and therefore discovery of the limitations of existing methods for specific applications. The presence of the developers also enabled live implementation of novel functionalities in the toolkit to support application teams' requests. During the course of the hackathons, one request made by the Exeter team led to the addition of an R language API enabling DataFrame importation.

We organised a series of three hackathons in the presence of VECMAtk developers, each lasting 2 to 3 days. The NEPTUNE community represented an important part of the participants in the three hackathons. We saw 5 to 10 participants from the NEPTUNE project join the online hackathon meetings out of the 25 to 30 participants. We report the contributions of each application team (BOUT++, Nektar++, EPOCH) during the hackathons in the following section.

3 Feedback from applications users UQ implementations

Three application teams were able to perform non-intrusive UQ of their single-scale model simulations. All teams were able to apply SC and PCE to compute sensitivity and uncertainty associated with up to four parameters. There is a consensus on using non-intrusive surrogates in the long term which will need to be refined and improved to handle a larger number of input parameters.

Below are reported the UQ attempts entirely and directly performed by the three applications teams.

3.1 Nektar++

Investigations of Nektar++ consisted in implementing UQ of single-scale simulations of a heat transport model providing a relationship between the quantity of interest (QoI) and two dimensionless numbers. In this preliminary work PCE, SC, and GP were used to fit various QoIs using EasyVVUQ only (not EasySurrogate).

$$\begin{aligned} \frac{1}{Pr}\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) &= -\nabla p + Ra\, T\, \hat{\mathbf{y}} + \nabla^2\mathbf{u} \\ \left(\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T\right) &= \nabla^2 T \\ \nabla\cdot\mathbf{u} &= 0. \end{aligned}$$

The two input parameters, Pr and Ra, were respectively the Prandtl number that is the ratio of kinematic viscosity to thermal diffusivity (fluid properties) and the Rayleigh number, that is the dimensionless temperature difference. Ra was varied in the range [1.0x10^4, 3.2x10^4] with log-uniform distribution; Pr varied in the range [1,10] (typical values for experiments with air and water) with uniform distribution.

For the steady-state problem, the Nektar++ team led by Ed Threlfall was able to perform SA and constructed PCE and SC surrogates, both using fifth-order polynomials during the hackathons (see figure 3).

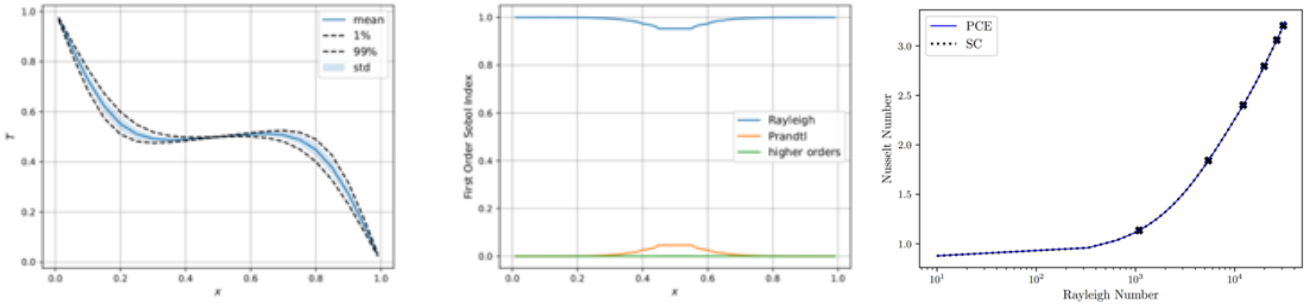

Figure 3: UQ of the horizontal temperature profile halfway up the cavity (left), SA (first-order Sobol indices) of the two parameters (centre), PCE and SC surrogate models of the Nusselt number (right).

Time-dependent simulations were also investigated using larger values of Ra. A GP surrogate model for time series was constructed using EasyVVUQ based on 12 samples varying only the parameter Ra. The surrogate model was in excellent agreement for quiescent cases (solution smooth) but issues were revealed in the case of higher Rayleigh numbers such as a persistent offset (see figure 4). More work is needed to design optimal surrogates for chaotic time series, in particular surrogates that predict the correct phase diagram of the many scaling regimes in convective turbulence. This may require better tuning of kernel parameters for the GP surrogate; such methods will be discussed in the next

section.

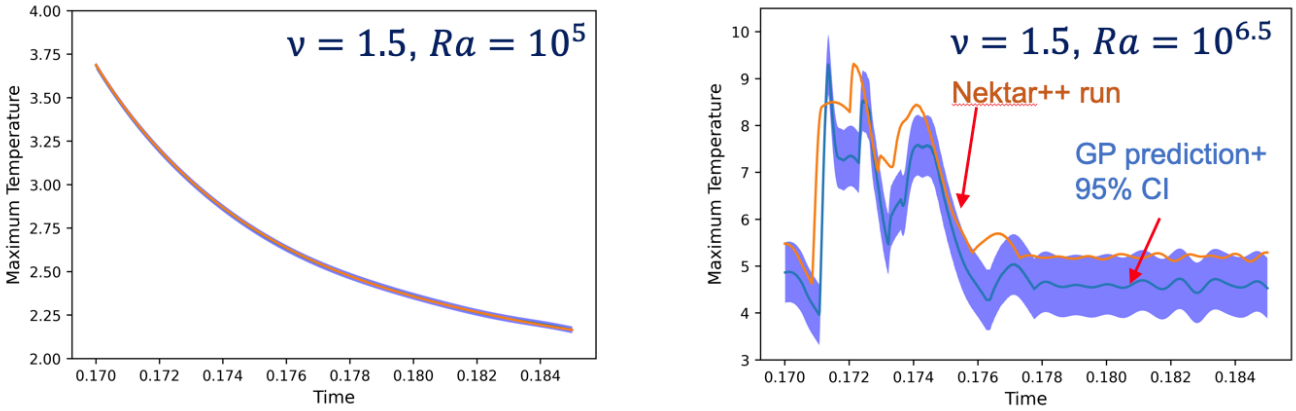

Figure 4: Fitting of the maximum wall temperature evolution for different regimes using a GP surrogate (Matérn v=1.5 in kernel, and Ra=10^5 or Ra=$10^{6.5}$).

3.2 BOUT++

Investigations of BOUT++ consisted in implementing UQ of single-scale simulations of a heat conduction 1D model and a plasma physics 2D model. In this preliminary work PCE, SC, and G were used to fit various QoIs using EasyVVUQ but also EasySurrogate in this case. The work was carried by a team constituted of Joseph Parker, Peter Hill, Ben Dudson and collaborators.

The 1D model focuses on the evolution of the temperature field expressed as different QoIs: $T(x, t_{end})$, $T(x_0, t)$, $\log[T(x, t_{end})]$, varying the initial temperature and a single diffusivity parameter χ (see figure 3):

$$\frac{\partial T}{\partial t} = \chi \frac{\partial^2 T}{\partial x^2}$$

Issues of negative values were reported but later were found to be an artefact of polynomial fitting to steep temperature gradients with respect to parameter variations near zero. The issue was fixed by using higher-order polynomial fitting in PCE and SC. Adaptive SC was also used to circumvent the curse of dimensionality anticipating performing UQ in high-dimension parametric spaces.

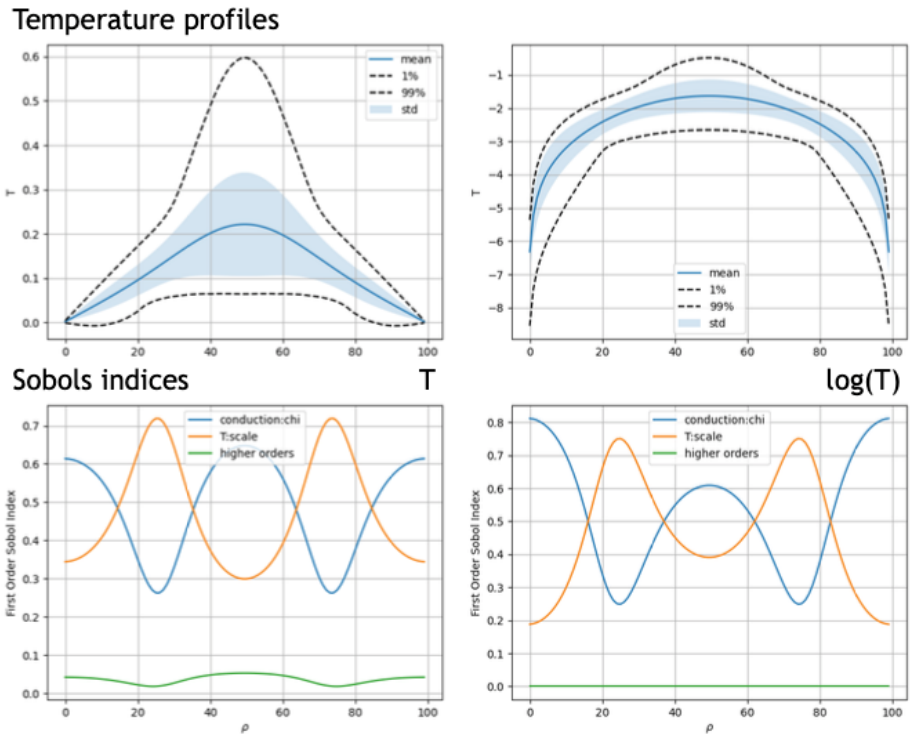

Figure 5: UQ (top) and SA (bottom) of the temperature profile expressed either as T (left) or log(T) (right).

A second campaign was then implemented to quantify uncertainty in the evolution of density perturbations in 2D plasma, varying four parameters: background density, temperature, and 2 dissipation parameters. Third-order PCE would have required 1296 cases, while SC only 256, the latter option was therefore retained.

In later hackathons, a surrogate model was used to determine numerical parameters, and in particular numerical (non-physical) parameters for the design of optimal simulations. The simulated model consisted of the time advance of hyperbolic PDE with elliptic PDE solved every time step:

$$\frac{\partial n}{\partial t} = -\{\phi, n\} + 2\frac{\partial n}{\partial z} + D_n \nabla^2 n$$

$$\frac{\partial \Omega}{\partial t} = -\{\phi, \Omega\} + \frac{2}{n}\frac{\partial n}{\partial z} + \frac{D_\Omega}{n}\nabla^2 \Omega$$

$$\nabla^2 \phi = \Omega$$

These equations were solved using nested solvers, namely using CVODE for time and multigrid for spatial integrations [11].

The surrogate model was built and trained for the aforementioned model which featured a 7-dimensional parameter space and a non-smooth dependence of behaviour on parameters. The QoI in the surrogate model is the error at a given timestep, that is max(E_a , αE_r) based on the absolute E_a, the relative E_r error tolerances (E_a, $E_r \in [10^{-15},1]$) and α a representative value of ||x||. The first step consisted of using PCE and adaptive SC to generate the surrogate models. Both methods achieved qualitatively similar results, but adaptive SC required many fewer code evaluations (130 vs 441). Both types of surrogates provided a qualitatively good model of errors at moderate order but over-fitted the noisier data at high order. One

main issue was that the accuracy of the surrogate models stopped increasing even though more data was provided.

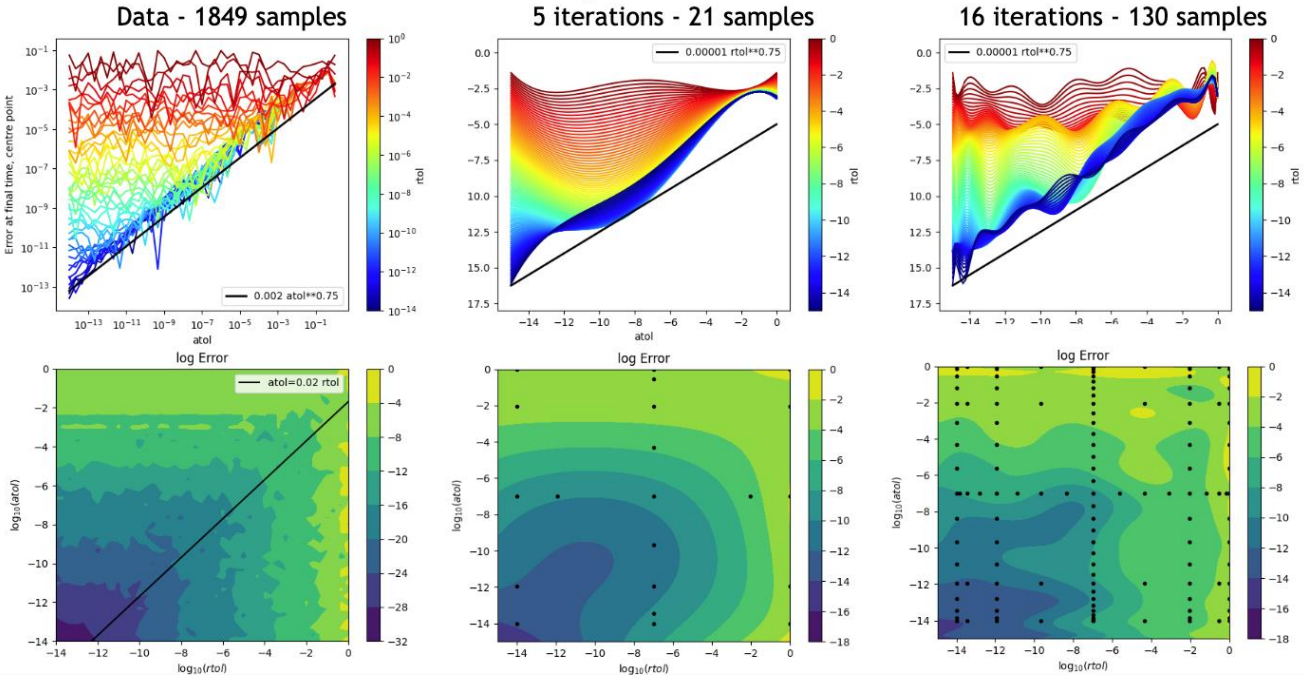

Figure 6: CVODE - 2D scan with adaptive SC. Training data obtained using a 2D manual parameter sweep (left), error model prediction at partial convergence (middle), at full convergence (right) of adaptive SC algorithm.

Two more advanced surrogate modelling methods were subsequently considered: Artificial Neural Network (ANN) with EasySurrogate, and GPs with the SKLearn library in EasyVVUQ; both capabilities were added to the VECMA Toolkit during the period of this project. ANNs were able to emulate the data correctly, at the cost of training on approximately 50% of the dataset. By contrast, GPs were able to emulate data using only 5% of the dataset, albeit the results were highly sensitive to the sampling points. Adaptive sampling of training data for GP surrogate models might reduce such sensitivity while preserving the need for the smaller parts of the dataset. This is an open question requiring further investigation.

3.3 EPOCH

Investigations led by Tom Goffrey and collaborators consisted in implementing a surrogate model for stimulated Raman scattering (SRS) in laser-plasma interactions. The surrogate was trained using data produced by the EPOCH code, a mini-app version of which is currently being used in the NEPTUNE project. The EPOCH model simulations are typical of particle-based simulations yielding chaotic systems. Simulations featured 5 to 10 parameters and PCE via EasyVVUQ was applied to build the surrogate models. Custom encoders and decoders were developed for the campaign and execution featuring the QCG-PJ was implemented. A proof of concept sensitivity analysis on the performance of simulated laser-driven implosions was completed. Further conclusions regarding UQ of particle-based methods using EPOCH were not obtained due to time constraints during the course of the hackathons and the project.

Nonetheless, in the meantime, we completed our investigations of standard classical molecular dynamics simulations, another exemplar particle-based simulation method [12]. Our work is the first and only fully detailed and thorough analysis of a particle-based code using modern UQ methods. Many-body particle-based simulations are chaotic systems. We showed that the uncertainty arises from a combination of (i)

the input parameters and (ii) the intrinsic stochasticity of the method controlled by the random seeds. We performed a sensitivity analysis, which revealed that, out of a total of 175 parameters, just six dominate the variance in the code output. The sensitivity analysis computed first and higher-order Sobol indices, which respectively highlight the individual and combined influence of the parameters. We showed that simulations of free energies dampen the input uncertainty, in the sense that the variation around the mean output free energy is less than the variation around the mean of the assumed input distributions if the output is ensemble-averaged over the random seeds. Without such ensemble averaging, the predicted free energy is five times more uncertain. The distribution of the predicted properties is thus strongly dependent upon the random seed. Owing to this substantial uncertainty, robust statistical measures of uncertainty in molecular dynamics simulation, and more widely we anticipate particle-based simulation methods, require the use of ensembles in all contexts.

3.4 Limitations of parametric UQ

The aforementioned variants of the SC/PCE methods all have underlying regularity assumptions and might not work well in cases of discontinuities or high gradients. The root cause of this is their use of global interpolation polynomials. Future work could focus on extending EasyVVUQ's capabilities by implementing a state-of-the-art method with polynomial basis functions of local support, capable of detecting irregularities and local refinement [13], extending the applicability of EasyVVUQ to handle outputs with (local) discontinuities or sharp gradients. GPs are also capable of handling irregular outputs and have lower sampling costs (typically linear in the number of parameters) but the cost of the fitting is cubic in the number of code evaluations. This need will push for the development of high-dimensional surrogates in EasySurrogate as we discuss in the next section, to facilitate coupling of surrogates (MUSCLE3) and easy execution on HPC infrastructures (QCG-PJ).

4 Methodological recommendations

In order to pursue this initial UQ effort in the NEPTUNE project, key challenges remain to be addressed:

- Feedback across models at different scales,
- High-dimensional parameter spaces,
- Possible abrupt changes in physical regimes leading to biased training or even unadapted surrogate model methods,
- Complexities of stochastic outputs in particle-based simulations,
- Real-time updating of the parameterisations when observations become available or when additional training is required.

These challenges could be tackled by investigating the methods described below during the subsequent phases of the NEPTUNE project. Non-intrusive UQ is not treated in this section as it has already been largely discussed in the first report 2047352 1-TN-01.

In a multiscale model, the equations describing the small scales tend to be the most uncertain and represent the largest drain on computational resources. The large-scale models, on the other hand, are often better known and are comparatively cheap. In this context, it might make sense to keep the large-scale models intact, and only replace the models that represent the small scales with cheap approximations. Gaussian processes and neural networks are key tools to build such models. In turn, such coupled physical–surrogate model systems can be simulated many times at far lower computational costs, making them suited for UQ. That said, depending on the structure of the multiscale model, such a semi-intrusive system can contain feedback between the surrogate and the (large-scale) physical model, which might push the surrogate away from the regime on which it was trained. Dealing with such kinds of feedback should be the target of future work. One option would be to train the surrogate to output physics consistent with the coupled environment, rather than training it `offline' to only represent (small-scale) data.

Active subspace methods attempt to circumvent the curse of dimensionality by dimension reduction of the input space. Specifically, the input vector $x \in \mathbb{R}^D$ is projected to a lower-dimensional "active" subspace $y \in \mathbb{R}^d$, via a tall-and-skinny matrix $W_1 \in \mathbb{R}^{D \times d}$ of orthogonal basis vectors. The active subspace is thus given by $y = W_1^T x \in \mathbb{R}^d$, and the idea is to create a surrogate model in the lower-dimensional space, rather than in the original (high-dimensional) input space. The key to this method is finding the matrix W_1, which identifies the directions in the input space along which the model varies the most. The original active-subspace method [14] builds this matrix from the orthogonal eigenvectors of the symmetric gradient matrix. This requires the gradient of the code output with respect to the inputs to be available, which might be difficult in practice. For this reason, a very promising alternative is the use of active-subspace variants which do not require gradient information, in particular those based on GPs [15] and neural networks [16].

One key ingredient is to use Gaussian Processes (GPs) as surrogates for (alternatively "emulators") of the computer models. These are comparable to Polynomial Chaos (PC) approaches in terms of accuracy [4]. GPs can be readily parallelised for large input/output spaces, as we are doing in the recently released Alan Turing Institute Package for fitting Gaussian Process Emulators to multiple output computer simulation results. It is named the Multi-Output Gaussian Process Emulator (MOGP) and relies on Research Software Engineering to manage, organise, and accelerate/parallelise some computations. The Turing platform MOGP and its (relatively) few existing tools for ROM can interoperate with other UQ platforms such as the now mainstream EasyVVUQ: both are Python libraries with similar philosophy but complement each other in terms of workflows (which are actually much more elaborate in EasyVVUQ), sampling (more elaborate in MOGP), surrogates (PC and GP for EasyVVUQ, advanced GP for MOGP) which are now well understood and interchangeable in workflows.

In parameter estimation problems for chaotic dynamical systems, data may only be available in time-averaged form; or it may be desirable to study time-averaged quantities to reduce difficulties arising from

the complex objective functions, with multiple local minima, which arise from trying to match trajectories. Indeed, the idea fits the more general framework of “feature-based data assimilation” which, in turn, is closely related to the idea of extracting sufficient statistics from the raw data. Many parameter estimation problems arising in applications can be cast in the framework of Bayesian inversion. This allows not only for an estimate of the parameters, but also for the quantification of uncertainties in the estimates. Data Assimilation (DA) approaches have the potential to greatly enhance UQ methods and algorithms when data will feed the models in real-time, as done in so-called digital twins’ technology. This should help in the context of active or online training of surrogates, incorporating data when necessary. When observations or additional data become available, the UQ approach of Bayesian calibration against observations can be employed. This approach needs work in the context of nuclear fusion due to both the high dimensionality of the problem and the requirement for speed. When observations become available for updates in real-time (even with proxies for simple experiments) it may be possible to carry out DA.

Real-time requires extremely highly optimised and robust techniques and software, which will be a serious undertaking within future activities in the NEPTUNE project. Particle filters (PFs) [17,18] offer an ensemble-based approach to sequential state inference that provides consistent estimates for non-Gaussian distributions. The simplest variant, the bootstrap PF, alternates propagating the ensemble members forward in time under the model dynamics with resampling according to weights calculated from the likelihood of the particles given the observed data. While PFs offer asymptotically consistent inference for general state space models, in practice they typically suffer from weight-degeneracy in high-dimensional systems: after propagation only a single particle has non-negligible weight. For even simple linear-Gaussian models, PFs have been shown to require an ensemble size which scales exponentially with the number of observations to avoid degeneracy [19,20,21]. For large state dimensions, even inference in linear-Gaussian models using the Kalman filter (KF) is computationally infeasible due to the high processing and memory costs of operations involving the full covariance matrix of the state distribution. This motivated the development of ensemble Kalman filter (EnKF) methods [22] which use an ensemble of particles to represent the state distribution rather than the full mean and covariance statistics. As the ensemble sizes used are typically much smaller than the state dimension the computational savings can be considerable. Although EnKF methods are only rigorously valid in an infinite ensemble limit for linear-Gaussian models [23], they have been empirically found to perform well in models with weakly non-linear state update and observation operators, even when using relatively small ensembles of size much less than the state dimension [24]; the performance of the EnKF in non-asymptotic regimes has been theoretically investigated in several recent works [25,26].

The RADDISH project (Real-time Advanced Data Assimilation for Digital Simulation of Numerical Twins on HPC) led by Serge Guillas offers new techniques and tools (https://github.com/Team-RADDISH) to overcome the serious issues which will arise, confronting EnKF due to nonlinearities and non-normal statistics through:

- a particle filter method, powerful yet general;
- a geometrically-driven and enhanced EnKF when parameters need to be estimated as well as state space updated.

5 Conclusions

Our original shortlist of suitable UQ methods has now been made available within the VECMA toolkit for implementation and testing by the NEPTUNE application teams. Prototype UQ campaigns on NEPTUNE single-scale models (BOUT++, Nektar++ and EPOCH) were performed during three hackathons which we organised during the course of the project. Such extensive testing helped to generate specific knowledge on the single-scale models as well as the efficiency of the UQ methods. We noted the limitations of current non-intrusive methods (PCE, SC) to handle high-dimension parametric spaces. We also reached the limits of PCE and SC to build surrogate models, in terms of accuracy and training data required.

Recommendations for major developments needed in the future include:

- dimension reduction methods based on active subspaces,
- production of efficient surrogate modelling methods
- online learning methodology such as DA.

Indeed, the BOUT++ application team faced limitations due to sampling point selection within the stochastic collocation and polynomial chaos expansion surrogate models that were available. Similar limitations were faced by the Nektar++ application team in the case of chaotic time series surrogate modelling.

Advanced surrogate modelling techniques will be key as they lessen the computational and mathematical complexity of challenging codes and their UQ. For instance, inputs can be reduced to a smaller set, either by ruling out parameters or in terms of dimensionality [27,28]. Similarly, outputs can be shrunk to a more manageable representation in terms of basis expansions of either time series of outputs or spatial representations.

6 References

[1] Boghosian, B. M., Coveney, P. V., & Wang, H. (2019). A new pathology in the simulation of chaotic dynamical systems on digital computers. Advanced Theory and Simulations, 2(12), 1900125.

[2] Smolyak, S., Quadrature and Interpolation Formulas for Tensor Products of Certain Classes of Functions, Doklady Akademii Nauk SSSR, Volume 4, 1963, pp. 240-243.

[3] Edeling, W., Coveney, P.V., et al. "The impact of uncertainty on predictions of the CovidSim epidemiological code", Nature Computational Science 1.2 (2021): 128-135. Doi: https://doi.org/10.1038/s43588-021-00028-9

[4] Vassaux, M., Edeling, W., Coveney, P.V., et al. "On the need for ensembles to handle aleatoric and parametric uncertainty in molecular dynamics simulation." (2021), Journal of Chemical Theory and Computation. Doi: https://doi.org/10.1021/acs.jctc.1c00526 [**5**] Scott BD. Free-energy conservation in local gyrofluid models. Phys Plasmas [Internet]. 2005;12(10):1–18. Available from: http://dx.doi.org/10.1063/1.2064968

[6] Coster DP, Basiuk V, Pereverzev G, Kalupin D, Zagórksi R, Stankiewicz R, et al. The European Transport Solver. IEEE Trans Plasma Sci. 2010;38(9 PART 1):2085–92.

[7] Jolliet S, Bottino A, Angelino P, Hatzky R, Tran TM, Mcmillan BF, et al. A global collisionless PIC code in magnetic coordinates. Comput Phys Commun. 2007;177(5):409–25.

[8] Bottino a, Vernay T, Scott B, Brunner S, Hatzky R, Jolliet S, et al. Global simulations of tokamak microturbulence: finite-β effects and collisions. Plasma Phys Control Fusion. 2011 Dec;53(12):124027.

[9] Scott B. Gyrokinetic Edge Turbulence and the Edge / Core Transition. In: Second IAEA Technical Meeting on the Theory of Plasma Instabilities: Transport, Stability and their Interaction. 2005

[10] Luk, O. O., Lakhlili, J., Hoenen, O., von Toussaint, U., Scott, B. D., & Coster, D. P. (2021). Towards validated multiscale simulations for fusion. Philosophical Transactions of the Royal Society A, 379(2197), 20200074.

[11] Hindmarsh, A. C., Brown, P. N., Grant, K. E., Lee, S. L., Serban, R., Shumaker, D. E., and Woodward, C. S. (2005). SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. ACM Transactions on Mathematical Software, 31, 363–396.

[12] Vassaux, M., Wan, S., Edeling, W. N., & Coveney, P. V. (2021). Ensembles Are Required to Handle Aleatoric and Parametric Uncertainty in Molecular Dynamics Simulation. Journal of Chemistry Theory and Computation, in press, doi: 10.1021/acs.jctc.1c00526

[13] Edeling, W. N., Dwight, R. P., & Cinnella, P. (2016). Simplex-stochastic collocation method with improved scalability. Journal of Computational Physics, 310, 301-328.

[14] Constantine, P. G., Dow, E., & Wang, Q. (2014). Active subspace methods in theory and practice: applications to kriging surfaces. SIAM Journal on Scientific Computing, 36(4), A1500-A1524.

[15] Liu, X., & Guillas, S. (2017). Dimension reduction for Gaussian process emulation: An application to the influence of bathymetry on tsunami heights. SIAM/ASA Journal on Uncertainty Quantification, 5(1), 787-812.

[16] Tripathy, R., & Bilionis, I. (2019, August). Deep active subspaces: A scalable method for high-dimensional uncertainty propagation. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (Vol. 59179, p. V001T02A074). American Society of Mechanical Engineers

[17] Gordon, N. J., David, J. S., and Smith, A. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation." IEE Proceedings F-radar and signal processing. Vol. 140. No. 2. IET, 1993.

[18] Del Moral, P. "Nonlinear filtering: Interacting particle resolution." Comptes Rendus de l'Académie des Sciences-Series I-Mathematics 325.6 (1997): 653-658.

[19] Snyder, C., Bengtsson, T., Bickel, P., and Anderson, J. "Obstacles to high-dimensional particle filtering." Monthly Weather Review 136.12 (2008): 4629-4640.
[20] Bengtsson, T., Bickel, P., and Li, B. "Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems." Probability and statistics: Essays in honor of David A. Freedman. Institute of Mathematical Statistics, 2008. 316-334.
[21] Snyder, C. "Particle filters, the “optimal” proposal and high-dimensional systems." Proceedings of the ECMWF Seminar on Data Assimilation for atmosphere and ocean. 2011.
[22] Burgers, G., van Leeuwen, P. J., and Evensen, G. "Analysis scheme in the ensemble Kalman filter." Monthly weather review 126.6 (1998): 1719-1724.
[23] Le Gland, F., Monbet, V. and Tran, V.-D. “Large sample asymptotics for the ensemble Kalman filter”. In The Oxford Handbook of Nonlinear Filtering (D. Crisan and B. Rozovskii, eds.) 598–631 (2011). Oxford University Press.
[24] Evensen, G. Data Assimilation: The Ensemble Kalman Filter, 2nd ed (2009). Springer.
[25] Del Moral, P., and Tugaut, J.. "On the stability and the uniform propagation of chaos properties of ensemble Kalman–Bucy filters." The Annals of Applied Probability 28.2 (2018): 790-850.
[26] Bishop, A. N., and Del Moral, P. "On the Stability of Matrix-Valued Riccati Diffusions”. arXiv preprint arXiv:1808.00235 (2018).
[27] Liu, X., & Guillas, S. (2017). Dimension reduction for Gaussian process emulation: An application to the influence of bathymetry on tsunami heights. SIAM/ASA Journal on Uncertainty Quantification, 5(1), 787-812.
[28] Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. Technometrics, 33(2), 161-174.