



Enabling UQ for Complex Modelling Workflows

Małgorzata J. Zimon^{1(✉)}, Samuel Antão^{1(✉)}, Robert Sawko¹, Alex Skillen²,
and Vadim Elisseev¹

¹ IBM Research, Daresbury, UK

{malgorzata.zimon,rsawko}@uk.ibm.com,

{samuel.antaov,vadim.v.elisseev}@ibm.com

² Scientific Computing Department, STFC, Daresbury, UK

alex.skillen@stfc.ac.uk

<http://research.ibm.com/labs/uk>

<http://www.scd.stfc.ac.uk>

Abstract. The increase of computing capabilities promises to address many scientific and engineering problems by enabling simulations to reach new levels of accuracy and scale. The field of uncertainty quantification (UQ) has recently been receiving an increasing amount of attention as it enables reliability study of modelled systems. However, performance of UQ analysis for high-fidelity simulations remains challenging due to exceedingly high complexity of computational workflows. In this paper, we present a UQ study on a complex workflow targeting a thermally stratified flow. We discuss different models that can be used to enable it. We then propose an abstraction at the level of the workflow specification that enables the modeller to quickly switch between UQ models and manage underlying compute infrastructure in a completely transparent way. We show that we can keep the workflow description almost unchanged while benefitting of all the insight the UQ study provides.

Keywords: Uncertainty quantification · Workflows · Modelling · High-performance computing

1 Introduction

With the increase of computational power, scientific and engineering simulations are tackling increasingly higher complex systems. Modelling is being used as a strategic tool to provide key insights for predictions and decision making. Therefore, it is important to address the presence of uncertainty in the model data, such as inexact knowledge of initial conditions, and incorporate probabilistic behaviours into analysis. Uncertainty quantification (UQ) methods can be used in model certification, parameter estimation, and inverse problems [10].

This work was supported by the STFC Hartree Centre's Innovation Return on Research programme, funded by the Department for Business, Energy & Industrial Strategy.

© Springer Nature Switzerland AG 2019

J. M. F. Rodrigues et al. (Eds.): ICCS 2019, LNCS 11540, pp. 269–281, 2019.

https://doi.org/10.1007/978-3-030-22750-0_21

Although the need for confidence intervals in modelling predictions is apparent, the inclusion of UQ into complex engineering studies is non-trivial. Not only can the model have a high degree of sophistication but also the underlying high-performance computing (HPC) resources can be challenging to manage. HPC is usually associated with UQ as an enabler, in the sense that the modelling applications typically require the memory, parallelisation ability, and compute power only found in top-tier HPC resources. Also, the need to replay simulations for multiple sets of inputs increases the amount of computation needed. Other property of UQ that stresses the HPC infrastructure is how the runtime requirements can change between runs, particularly with multi-resolution methods [17].

As scientific simulations and modelling become more complex, they tend to be organised in workflows as opposed to single monolithic applications. During the process of designing an orchestrated set of applications, the requirements and the way data is used may change. Additionally, the number of applications and their dependences can vary. Therefore, the end-user would appreciate a flexible and interactive environment. The modeller should be able to focus exclusively on improving the computational experiment and avoid having to dive into challenging infrastructure details. In addition, the assimilation of the UQ study within the workflow in a integrated way would bring an increased benefit. This can be used to drive the workflow design itself and enable defining trade-offs at an early stage, e.g. the accuracy of the UQ model versus the computational cost. Workflows also add an auditability dimension to the modelling work which is relevant for many organisations. A platform that delivers on these requirements would democratise the use of UQ (and HPC for that matter) across less experienced users who might avoid performing stochastic modelling due to a steep learning curve.

In this work, we present an example of a complex UQ analysis needed to estimate the influence of parametric variability in the transient simulation of conjugate heat transfer. We demonstrate the use of a non-intrusive, adaptive method by assessing the propagation of thermal shock within a u-shaped bend. This case is relevant to nuclear applications, where hot-cold cycles can lead to thermal fatigue and material failure. The influence of shock-magnitude on wall temperatures is assessed.

Supported on this analysis, we then discuss an abstraction in which the end-user can express UQ studies cleanly as part of its own simulation/modelling workflow and with them have a common infrastructure that handles these two components jointly as they were a single one. Furthermore, we explore possibilities to enhance reusability of workflow components by multiple end-users.

This paper expands on [17]. Therein, an architecture to easily deploy UQ studies as workflows is proposed. That solution uses an as-a-service approach, integrating a mix of cloud and traditional HPC technology as well as the components to manage a dynamic and scalable infrastructure. That architecture not only provides the requirements of UQ studies but also reduces the load on the users, as it is able to *smart* schedule jobs and predetermine important runtime param-

ters (domain partitions, compute nodes, threading) with an artificial intelligence engine, providing the baseline infrastructure for the abstraction proposed herein.

The paper is organised as follows. Section 2 describes the general polynomial chaos method and its non-intrusive, multi-element variant; Sect. 3 discusses current infrastructures to enable UQ, while Sect. 4 describes our proposed abstraction; Sect. 5 demonstrates an application of the abstraction to perform a complex UQ study for heat transfer flow modelling; finally, Sect. 6 draws some conclusions.

2 Non-intrusive Methods for Uncertainty Quantification

In this work, the general polynomial chaos (gPC) approach has been applied to quantify uncertainty in the simulation of thermally stratified flow. The following section describes a variant of generalised polynomial chaos, non-intrusive spectral projection (NISP). We discuss an extension of the method which utilises discretisation of the stochastic space for low-regularity problems and long-time integration. This modification of gPC was first proposed by Wan and Karniadakis [12] as multi-element generalised polynomial chaos (ME-gPC), which yields approximate local statistics for uniformly distributed variables. The formulation was then extended to deal with stochastic inputs with arbitrary probability measures [13]. Non-intrusive multi-element surrogate modelling was later described by Foo et al. [5]. This approach performs the probabilistic stochastic collocation in a decomposed random space. In this article, we apply the same logic as in the multi-element probabilistic collocation method (ME-PCM), but instead of using a Lagrangian interpolant to obtain the surrogate, we perform a pseudo-spectral (discrete) projection [15].

Let $(\Omega, \mathcal{F}, \mu)$ be a probability space, where Ω is the sample space, \mathcal{F} is the σ -algebra of subsets of Ω , and μ is a probability measure. Given an uncertain \mathbb{R}^d -valued input, $\boldsymbol{\xi} = (\xi_1, \dots, \xi_d)$, gPC seeks a representation of a quantity of interest (QoI), $Y(\omega) \in L_2(\Omega, \mathcal{F}, \mu)$, as a sum of weighted polynomials:

$$Y(\omega) = \sum_{k=0}^{\infty} y_k \Psi_k(\boldsymbol{\xi}(\omega)), \quad (1)$$

where ω is a random event, the basis functions $\Psi_k(\boldsymbol{\xi}(\omega))$ are orthogonal with respect to μ and $\{y_k\}_{k=0}^{\infty}$ is a set of expansion coefficients. The list of polynomials whose weights are associated with a particular random variable can be found in [16]. The representation in Eq. 1, including the series truncated at $N_p + 1$ elements, enables the calculation of the stochastic moments, i.e. expected value, \mathbb{E} , and variance, \mathbb{V} , in terms of polynomial coefficients:

$$\mathbb{E}[Y] \langle \Psi_0 Y \rangle = y_0, \quad \mathbb{V}[Y] = \mathbb{E}[|Y - \mathbb{E}[Y]|^2] = \sum_{k=0}^{N_p} y_k^2 \langle \Psi_k^2 \rangle, \quad (2)$$

where $\langle \cdot \rangle$ denotes the inner product; number of terms $N_p + 1 = \frac{(p+d)!}{p!d!}$ depends on p and d representing polynomial order and stochastic dimension, respectively.

In the intrusive gPC approach, the polynomial expansion of QoI is introduced in the solver, resulting in a coupled system of equations which is often non-trivial to derive. On the other hand, stochastic collocation methodologies treat the simulator as a *black box* and only use the simulation outputs to estimate the polynomial coefficients. Therefore, the sampling methods are attractive if the UQ study has to be performed with large, complex codes and workflows which are not amenable to changes.

Non-intrusive algorithms comprise the following steps: (1) choose a set of inputs, (2) run the deterministic code for each value, and (3) using the set of solutions construct the expansion of summed polynomials, either through interpolation or discrete projection. The approach is independent of the solver, and executions of the simulator can be made concurrently. In the NISP approach, a surrogate model \tilde{Y} is constructed using the realisations of Y :

$$\tilde{Y} = \sum_{k=0}^{N_p} \tilde{y}_k \Psi_k \approx Y, \quad (3)$$

where $\tilde{y}_k = \frac{1}{\langle \Psi_k^2 \rangle} \sum_{q=1}^Q y(\xi^{(q)}) \Psi_k(\xi^{(q)}) w^{(q)}$ and $(\xi^{(q)}, w^{(q)})$ are the prescribed quadrature nodes and their corresponding weights.

In the multi-element variant of stochastic collocation, the parametric space is discretised into N_e non-overlapping subsets $\{B^j\}_{j=1}^{N_e}$. For each element, the set of collocation points (quadrature nodes) is prescribed and local surrogates are estimated. Over each element B^j , a change of variable is applied in order to have a mapping on $[-1, 1]$. Construction of orthogonal polynomials for arbitrary PDFs can be performed with a three-term recurrence relation. After the global approximant is reconstructed from multiple models, the approximate global stochastic moments can be assembled from the local statistics with the Bayes' formula [5]. In [5, 13], the level of stochastic discretisation is decided adaptively by assessing the contribution of the highest polynomial terms to total variance.

3 Existing Infrastructure for Workflow-Based UQ Studies

UQ models are typically designed disconnected from the management of the compute infrastructure (execution engine). On the UQ model side, DAKOTA is arguably one of the most popular frameworks, it comprises a vast range of UQ models that it exposes through textual specification with its own syntax or through a C++ application programming interface (API). OpenTURNS is another popular framework that exposes many UQ models through a Python API. None of these frameworks offer a way to easily specify infrastructure-related requirements unless the user explicitly integrates them in his/her workflow specification. There are some other projects, e.g. Sandia Analysis Workbench (SAW) [6] for DAKOTA and Salome [4] for OpenTURNS, that provide wrappers that enable some of this integration.

On the compute infrastructure side, there are many execution engines that support the execution of workflows, some of them comprise the architecture proposed in [17]. These execution engines can be more cloud-oriented (e.g. ICP [9]), others more traditional batch-job oriented (e.g. CWLEXEC [3] with IBM Spectrum LSF) [8], and others somewhere in between (e.g. Flux [1]).

Approaches to represent workflows based on programmatic languages have also been proposed in [14] (XML formats) and [2] (Python). Both focus on decorating the data types so that the execution engine can optimise data management.

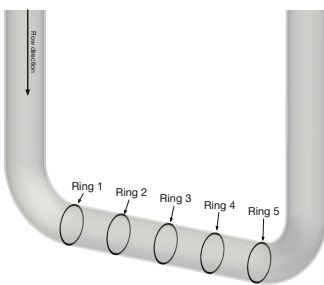
A characteristic of these works is that they all consider that the logic to perform UQ (or other kind of parametric studies) will have to be appended to the baseline workflow, making it more complex.

4 Abstraction for UQ Studies Specification

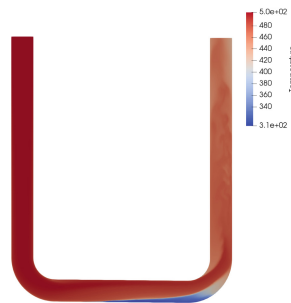
In this section, we propose an abstraction to allow users to easily specify a workflow and identify where quantities of interest for a UQ study are produced and execute all the required logic on a distributed HPC system seamlessly, with minimum user intervention. The scientific/modelling workflow is the central element, the UQ study is built/derived from it.

4.1 A UQ Case Study

We motivate the discussion of the proposed abstraction with a UQ case study around the thermal properties of a flow in a u-shaped bend. We aim to estimate the temperature distribution in a pipe as a function of the Froude number to gain better understanding of stratified flow development. We measure the wall temperature at rings shown in Fig. 1a.



(a) Sketch of ring locations. Rings are located at $x/D \approx 1.5, 3, 4.5, 6$ and 7.5 for rings 1 to 5 respectively, and have radius $0.525D$ (i.e. in the middle of the wall).



(b) Temperature distribution in a u-shaped bend at non-dimensionalised time $\tilde{t} = 40$ for $Fr \approx 0.621$.

Fig. 1. U-shaped bend geometry and simulation result sample.

The Reynolds-averaged Navier Stokes (RANS) in our target simulation is based upon that of Viollet [11]. The thermal and material properties of the system are determined from the following dimensionless groups:

- Reynolds number: $Re = 10000$, based on pipe diameter, D .
- Reduced Froude number: $Fr \equiv \frac{U}{\sqrt{g \frac{\delta\rho}{\rho} D}} = 0.67 \pm 40\%$.
- Peclet number: $Pe \equiv \frac{UD}{\alpha} = 6 \times 10^4$.

where U , g , ρ and α are the bulk velocity, gravitational acceleration, fluid density, and thermal diffusivity, respectively.

The flow is fully developed at the inlet, and is allowed to reach a (statistically) steady isothermal state before commencing the thermal transient. A hot shock is introduced at the inlet at time t_0 , and increases linearly until time t_1 where it remains at the maximum temperature. The duration of the ramp is $t_1 = t_0 + 7.5U/D$. The ratio of thermal diffusivities is based on water flowing within a steel pipe, i.e. $\alpha_{solid}/\alpha_{fluid} = 144.8$.

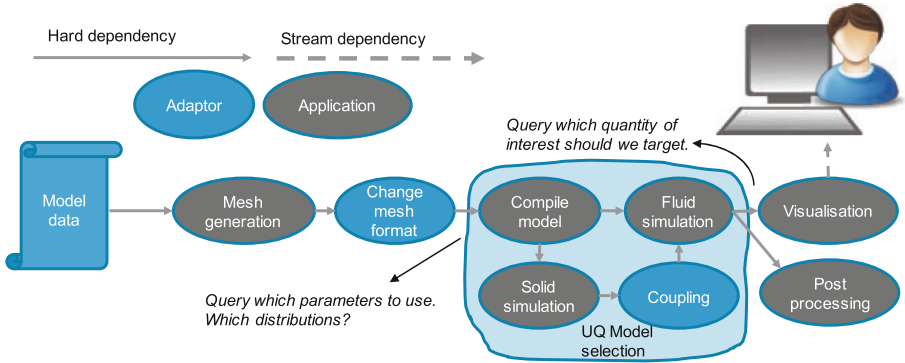
The surrogates are built at non-dimensionalised $\tilde{t} \equiv Ut/D = 40$; at this time a stable stratification occurs as shown in Fig. 1b. We aim to show the benefit of using a multi-element variant of NISP to build an accurate input-output representation relationship. At first, we approach the problem with a standard NISP method. We discuss the strengths of the method and identify its limitations which can be addressed with local polynomial bases. We estimate the stochastic response using global orthogonal bases. As the function of the simulator is unknown, the convergence is analysed based on results from consecutive polynomial resolutions. The approximation error of expected value E obtained with p -th polynomial order is defined here as $|E_p - E_{p-1}|/|E_{p-1}|$, where $|\cdot|$ denotes L_2 norm.

In this study, we treat the Froude number $Fr(\xi)$ to be a random variable, where ξ has a uniform distribution $Fr \sim U(0.402, 0.938)$. Both NISP approaches use Gaussian quadrature approximation as it is an optimal choice for problems with a small stochastic dimension. However, if a single simulation execution is expensive, nested quadratures may reduce the computational effort. In the multi-element variant of NISP, we perform the discretisation purely based on the value of the last term of the polynomial expansion with respect to the first coefficient. If the ratio is larger than a prescribed threshold, then the term has a significant influence and the random space should be divided.

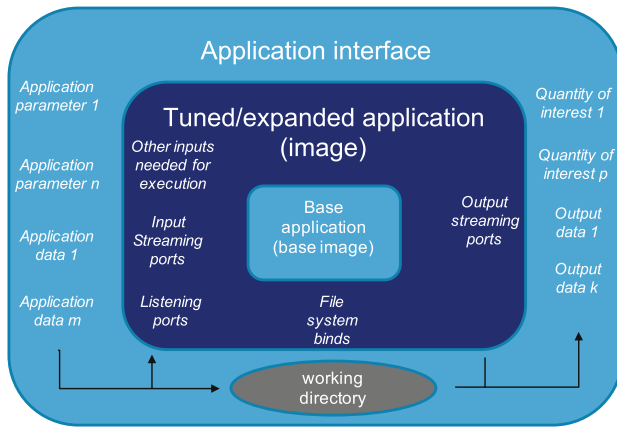
4.2 Workflow Abstraction for UQ

Figure 2a shows how a workflow underlying the UQ study specified in Sect. 4.1 would look like. In the following, we propose a set of directions to abstract the way users can specify a UQ study based on a given workflow.

The Data. Data can have a myriad of different formats depending on the tool/domain. If data needs to be filtered the user needs to specify an adaptor,



(a) Workflow specification. Hard and stream connector denote dependencies between applications and data.



(b) Application specification.

Fig. 2. Abstraction used to define the workflow.

see below. Connection to external data repositories can also be specified, either by valid URLs that can be used directly by applications, or by adaptors that access that data and mirror to a file system. Data persistence/replication can be controlled and minimised by using hashing techniques similar to what exists already today in modern data repositories, e.g. Git or DockerHub.

The Application. At the heart of every workflow is the application. Like the data, applications can be very diverse and require a specific environment to execute correctly. This environment can also depend on other applications. This kind of hierarchic dependence of applications is addressed by several cloud technologies through containerisation with Docker [7] being arguably the most popular. Therefore, here we assume that the application is defined through means of an image whose instances can be deployed as a container.

Similar to what happens today to containers, the description of the application will have to be decorated with information of exposed ports that can be used to stream in/out data without going through a file system - see Fig. 2b. Application parameters and generated quantities of interest can be exposed in the same way. This allows parametrisation to be easily hooked up with non-intrusive UQ studies.

Applications can also be visualisation tools. We consider the user owned workstation to be a possible resource to be used during deployment for which a matching client application exists. This could leverage local graphics for rendering a stream of data or deploy a client in the workstation connected with a remote server that can render visualisation frames locally (in-situ visualisation).

Adaptors. Adaptors are just like applications, except that they are meant to change data so that it can be adapted to work on an application downstream in the workflow. The assumption on the adaptors is that they are lightweight and can be replayed safely. They can be used to perform simple tasks like serialisation/deserialisation of data or to generate plots. Adaptors are designed to address a problem that limits the adoption of workflow design tools, which is the tweaking of an application for a slightly different format of data, or set of inputs. They make the components of the workflows more modular and therefore more reusable.

UQ Specification. UQ can be seen as a more sophisticated form of parameterisation study. As discussed above, applications and adaptors are configured to expose parts of the internal environment as parametrised entries. For a given workflow, one can potentially perform a UQ study on a connected subset of workflow components selected by the user. The infrastructure can detect the inputs and quantities of interests from the applications/adaptors in that subset. For that, it only has to evaluate the connections to other components that are not part of that subset (UQ model selection area in Fig. 2a). This can be done automatically based on the application/adaptor specification. The user only has to select the type of UQ model. Once the set of components and UQ model is known, the user is notified of which parameters and quantities were identified. Then, he/she can select which should be targeted, and which distributions should be used. This makes it almost trivial to incorporate a UQ study as part of the design of the workflow.

This approach has the property of being suitable for nesting, i.e. one can make UQ studies within UQ studies, by selecting subsets. The infrastructure would repeat the process above as needed.

5 Results

We executed the workflows based on the abstraction proposed in the Sect. 4 using the standard NISP approach. Initially the UQ study is performed on the temperature distribution in ring 2 as shown in Fig. 1a.

As shown in Fig. 3a and b, for given threshold values, we reach statistical convergence above $p = 10$. For the sake of the study, we validate the surrogate with $p = 13$ (which was obtained with $p + 1 = 14$ runs) against a large number of simulation results. Figure 4 shows that the polynomial expansion reproduces well the mean temperature in the second ring for varying Froude numbers.

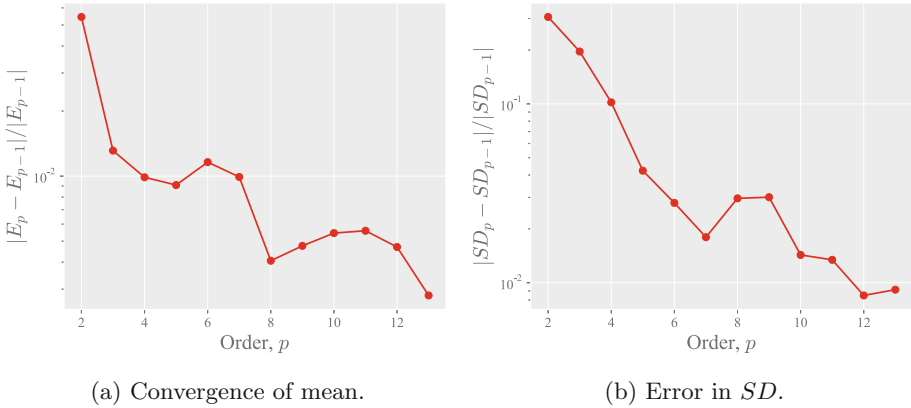


Fig. 3. Error in expected value and standard deviation of temperature. Statistics were obtained from surrogate models built with increasing polynomial order.

The resulting mean temperature distribution and standard deviation are plotted in Fig. 5a. Such representation of results suggests that each value of temperature within the uncertainty range is equally probable to occur. However, the PDFs of the simulation outputs in Fig. 5b give an interesting insight – the distribution of QoIs is bimodal in nature, i.e. extreme values of temperature are more likely to occur. This observation highlights misinterpretation of the system response in finite order statistics and stresses the need for performing the UQ analysis. If classical Monte Carlo approach was used, at least an order of magnitude more calculations would be required to obtain the same result as with polynomial expansion.

For the approximation, a fairly large polynomial order is used. Surrogates built with smaller number of quadrature nodes allow to accurately describe stochastic moments. However, if the approximant is to be treated as a replacement of high-fidelity simulation, then better representations might be needed. More polynomial terms are retained due to the complexity of the functions that describe input-output relationship. This is particularly visible when analysing the temperature sampled at the first ring. Although the convergence of the first moments is reached, even for $p = 15$ the model does not return all of the simulation points (see Fig. 6a). Using polynomials of higher degree could improve the result but at significant computational cost.

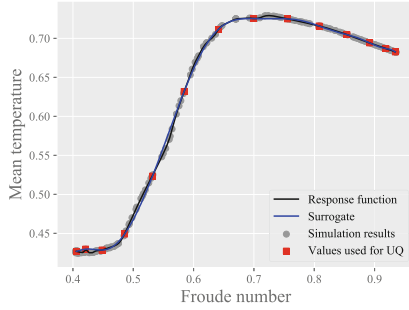
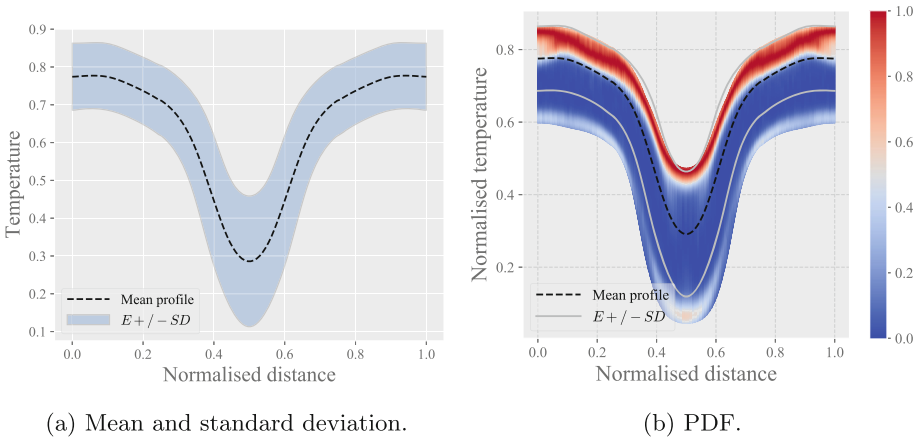


Fig. 4. Mean temperature as a function of Froude number obtained from a surrogate model with polynomial order $p = 13$. Black line with gray dots is a real response of the system that is being approximated. Each circle denotes simulation results. Red squares mark the outputs that were used to build the surrogate (blue line). (Color figure online)



(a) Mean and standard deviation.

(b) PDF.

Fig. 5. Statistics derived from the PC-based surrogate with $p = 13$.

To circumvent this limitation, we define an analysis which uses piecewise low-degree polynomial basis [5, 12]. The random space refinement (increasing number of elements N_e) is performed adaptively based on the behaviour of the last polynomial coefficient. If its value with respect to the zeroth polynomial term is larger than a prescribed threshold, then the random space is halved.

We should stress here that changing between methods does not require modifications to the defined workflow. Instead, we only adjust the selection of the UQ model to use. This highlights the benefits of using an abstraction for the parametric study and improved modelling.

Figures 6b and c showcase the improvement in surrogate modelling when using piece-wise smooth polynomials. By adaptively dividing the random space into sections and constructing partial model response functions, we can obtain

a better approximation as with global basis. The individual polynomial order of each surrogate was kept low, $p = 3$. Therefore, a better model approximation can be achieved with discretisation ($N_e > 1$) at comparable computational cost of a global surrogate. The mean of relative errors, $\delta = \frac{1}{N} \sum_{n=1}^N |Y(Fr_n) - \tilde{Y}(Fr_n)| / |Y(Fr_n)|$ for the partial surrogates was $\delta = 0.0018$ and $\delta = 0.0011$ for $N_e = 4$ and $N_e = 5$, respectively, while the error of the global approximant with 16 simulation runs was $\delta = 0.0053$.

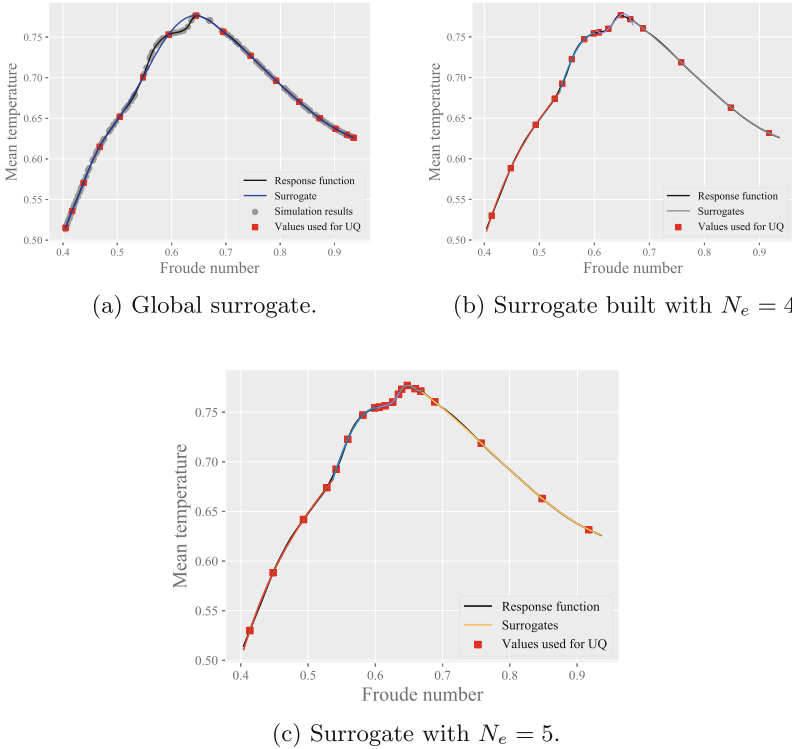


Fig. 6. Comparison between a surrogate for the 1st ring’s temperature constructed with a global basis and a set of locally supported polynomials. In (b) the same number of points was used as for the solution shown in (a). The global surrogate with the same number of points as $N_e = 5$ would have $3\times$ higher error.

6 Conclusions

We have shown that the knowledge extracted from the modelled system can be maximised by using surrogate-based UQ analysis, allowing the prediction of the influence of varying conditions on QoI. This can be done without changing existing simulation codes, and at much smaller cost than using a brute-force approach.

We have applied a non-intrusive polynomial-based approach to model parametric uncertainty associated with conjugate heat transfer modelling. The study of response functions shows that in this case piece-wise polynomials are better suited for constructing accurate surrogates.

The analysis is facilitated by the introduction of an abstraction that comprises a way to specify the scientific application along with high-throughput workflow. The solution derives the necessary details for the UQ study from the existing application specification. The paper has highlighted the need for flexibility allowing modellers to transparently deploy their workflows in a dynamic fashion.

References

1. Ahn, D.H., et al.: Flux: overcoming scheduling challenges for exascale workflows. In: WORKS 2018: 13th Workshop on Workflows in Support of Large-Scale, pp. 10–19, November 2018
2. Babuji, Y., et al.: Parsl: scalable parallel scripting in python. In: 10th International Workshop on Science Gateways (2018)
3. Computing, I.S.: CWLEXEC (2018). <https://github.com/IBMSpectrum-Computing/cwlexec>
4. EDF: SALOME (2018). <https://www.salome-platform.org/>
5. Foo, J., Wan, X., Karniadakis, G.E.: The multi-element probabilistic collocation method (ME-PCM): error analysis and applications. *J. Comput. Phys.* **227**(22), 9572–9595 (2008)
6. Friedman-Hill, E., Hoffman, E., Gibson, M., Clay, R.: Incorporating workflow for V&V/UQ in the sandia analysis workbench. In: NAFEMS World Congress, June 2015
7. GmbH, D.I.: Docker (2018). <https://www.docker.com/>
8. IBM: IBM Spectrum LSF (2017). <https://developer.ibm.com/storage/products/ibm-spectrum-lsf>
9. IBM: IBM Cloud Private (2018). <https://www.ibm.com/cloud/private>
10. Sullivan, T.J.: Introduction to Uncertainty Quantification. TAM, vol. 63. Springer, Cham (2015). <https://doi.org/10.1007/978-3-319-23395-6>
11. Viollet, P.: Observation and numerical modelling of density currents resulting from thermal transients in a non rectilinear pipe. *J. Hydraul. Res.* **25**(2), 235–261 (1987)
12. Wan, X., Karniadakis, G.E.: An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *J. Comput. Phys.* **209**(2), 617–642 (2005)
13. Wan, X., Karniadakis, G.E.: Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM J. Sci. Comput.* **28**(3), 901–928 (2006)
14. Wu, S., Mortveit, H.S.: A general framework for experimental design, uncertainty quantification and sensitivity analysis of computer simulation models. In: 2015 Winter Simulation Conference (WSC), pp. 1139–1150, December 2015
15. Xiu, D.: Numerical Methods for Stochastic Computations: A Spectral Method Approach. Princeton University Press, Princeton (2010)

16. Xiu, D., Karniadakis, G.E.: The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24**(2), 619–644 (2002)
17. Zimoń, M., Elisseev, V., Sawko, R., Antão, S., Jordan, K.: Uncertainty quantification-as-a-service. In: Proceedings of the 28th Annual International Conference on Computer Science and Software Engineering, CASCON 2018, pp. 331–337. IBM Corp., Riverton (2018)