The Journal of Open Source Software

UQit: A Python package for uncertainty quantification (UQ) in computational fluid dynamics (CFD)

Saleh Rezaeiravesh[*1,2], **Ricardo Vinuesa**[1,2], **and Philipp Schlatter**[1,2]

1 SimEx/FLOW, Engineering Mechanics, KTH Royal Institute of Technology, **2** Swedish e-Science Research Centre (SeRC), Stockholm, Sweden

DOI: 10.21105/joss.02871

Software

- Review
- Repository
- Archive

Editor: Vincent Knight

Reviewers:

- @jayten
- @ctdegroot

Submitted: 09 November 2020
Published: 20 April 2021

Introduction

In computational physics, mathematical models are numerically solved and as a result, realizations for the quantities of interest (QoIs) are obtained. Even when adopting the most accurate numerical methods for deterministic mathematical models, the QoIs can still be up to some extent uncertain. Uncertainty is defined as the lack of certainty and it originates from the lack, impropriety or insufficiency of knowledge and information (Ghanem et al., 2017; Smith, 2013). It is important to note that for a QoI, uncertainty is different from error which is defined as the deviation of a realization from a reference (true) value. In computational models, various sources of uncertainties may exist. These include, but not limited to, the fidelity of the mathematical model (i.e., the extent by which the model can reflect the truth), the parameters in the models, initial data and boundary conditions, finite sampling time when computing the time-averaged QoIs, the way numerical errors interact and evolve, computer arithmetic, coding bugs, geometrical uncertainties, etc. Various mathematical and statistical techniques gathered under the umbrella of uncertainty quantification (UQ) can be exploited to assess the uncertainty in different models and their QoIs (Ghanem et al., 2017; Smith, 2013). The UQ techniques not only facilitate systematic evaluation of validation and verification metrics, but also play a vital role in evaluation of the confidence and reliability of the data acquired in computations and experiments. Note that accurate accounting for such confidence intervals is crucial in data-driven engineering designs.

In general, uncertainties can be divided into two main categories: aleatoric and epistemic (Smith, 2013). The aleatoric uncertainties are random, inherent in the models, and hence, cannot be removed. In contrast, epistemic uncertainties originate from using simplified models, insufficient data, etc. Therefore, they can be reduced, for instance, through improving the models. As a general strategy in UQ, we try to reformulate the epistemic uncertainties in terms of aleatoric uncertainties so that probabilistic approaches can be applied. To implement the resulting framework, we can adopt a non-intrusive point of view which does not require the computational codes, hereafter simulators, to be modified. As a result, the UQ techniques can be combined with different features of computer experiments (Santner et al., 2003).

These strategies constitute the foundations of developing `UQit`, a Python package for uncertainty quantification in computational physics, in general, and computational fluid dynamics (CFD), in particular. In CFD, the Navier-Stokes equations are numerically integrated as a model of fluid flows. The flows are, in general, three-dimensional and time-dependent (unsteady) and at most of the Reynolds numbers relevant to practical applications, turbulent. A wide range of approaches has been used for numerical modeling of turbulence (Sagaut et al., 2013). Moving from low- toward high-fidelity approaches, the nature of the uncertainties inherent in the simulations change from model-based to numerical-driven. Regardless of the approach, we may need to study the influence of different factors on the simulations QoIs, where UQ techniques are beneficial.

[*]Corresponding author

Statement of need & Design

Performing different types of UQ analyses in CFD is so important that it has been considered as one of the required technologies in the NASA CFD vision 2030 (Slotnick et al., 2014). In this regard, `UQit` can be seen as a good match noting that it can be (one of) the first Python-based open-source packages for UQ in CFD. In fact, there are many similarities as well as connections between UQ and the techniques in the fields of machine learning and data sciences in which Python libraries are rich. These besides the flexible design of `UQit` provide a good potential for further development of `UQit` in response to different needs coming up in particular applications. Due to the non-intrusive nature of the implemented UQ techniques, `UQit` treats the CFD simulator as a blackbox, therefore it can be linked to any CFD simulator conditioned on having an appropriate interface. As a possible future development, a Python VTK interface can be considered for the purpose of in-situ UQ analyses which will be suitable for large-scale simulations of fluid flows on supercomputers without the need of storing large data sets.

The documentation for each UQ technique in `UQit` starts from providing an overview of the theoretical background and introducing the main relevant references. These are followed by the details of implementation, instructions on how to use the method, and notebooks. The examples in each notebook are exploited not only as a user guide, but also as a way to verify and validate the implementations in `UQit` through comparison of the results with reference data. Considering these aspects, `UQit` provides an appropriate environment for pedagogical purposes when it comes to practical guides to UQ approaches.

Features

Here, a short summary of the main UQ techniques implemented in `UQit` is given. In general, the methods are implemented at the highest required flexibility and they can be applied to any number of uncertain parameters. For the theoretical background, further details, and different applications in CFD, see our recent paper (Rezaeiravesh et al., 2020).

Surrogates play a key role in conducting non-intrusive UQ analyses and computer experiments. They establish a functional relation between the simulator outputs (or QoIs) and the model inputs and parameters. The surrogates are constructed based on a limited number of training data and once constructed, they are much less expensive to run than the simulators. `UQit` uses different approaches to construct surrogates, including Lagrange interpolation, polynomial chaos expansion (Xiu, 2010; Xiu & Karniadakis, 2002), and Gaussian process regression (Gramacy, 2020; Rasmussen & Williams, 2005). In developing `UQit`, a high level of flexibility in constructing GPR surrogates has been considered especially when it comes to incorporating the observational uncertainties.

The goal of **uncertainty propagation** or **UQ forward problem** is to estimate how the known uncertainties in the inputs and parameters propagate into the QoIs. In `UQit`, these problems are efficiently handled using non-intrusive generalized polynomial chaos expansion (PCE) (Xiu, 2010; Xiu & Karniadakis, 2002). For constructing a PCE, `UQit` offers a diverse set of options for the schemes of truncating the expansion, types of parameter samples, and methods to compute the coefficients in the expansion. For the latter, regression and projection methods can be adopted. As a useful feature for computationally expensive CFD simulations, the compressed sensing method can be utilized when the number of training samples is less than the number of terms in the expansion. By combining standard PCE and GPR, `UQit` provides the novel probabilistic PCE which is applicable to many CFD applications.

Global sensitivity analysis is performed to quantify the sensitivity of the QoIs with respect to the uncertain inputs and parameters. Contrary to the local sensitivity analysis, in GSA all parameters are allowed to vary simultaneously and no linearization is involved in computing sensitivities. In `UQit`, Sobol Sensitivity Indices (main, interaction, and total) (Sobol, 2001) are computed as indicators of GSA.

Driven by the needs, different features will be developed and added to `UQit` in future.

Acknowledgments

This work has been supported by the EXCELLERAT project which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823691. The financial support by the Linné FLOW Centre at KTH for SR is gratefully acknowledged. PS and SR acknowledge financial support by the Knut and Alice Wallenberg Foundation as part of the Wallenberg Academy Fellow programme.

References

Ghanem, R., Higdon, D., & Owhadi, H. (Eds.). (2017). *Handbook of uncertainty quantification*. Springer International Publishing. https://doi.org/10.1007/978-3-319-12385-1

Gramacy, R. B. (2020). *Surrogates: Gaussian process modeling, design and optimization for the applied sciences*. Chapman Hall/CRC. ISBN: 9781000766523

Rasmussen, C. E., & Williams, C. K. I. (2005). *Gaussian processes for machine learning (adaptive computation and machine learning)*. The MIT Press. ISBN: 026218253X

Rezaeiravesh, S., Vinuesa, R., & Schlatter, P. (2020). An uncertainty-quantification framework for assessing accuracy, sensitivity, and robustness in computational fluid dynamics. *arXiv:2007.07071*.

Sagaut, P., Deck, S., & Terracol, M. (2013). *Multiscale and multiresolution approaches in turbulence: LES, DES and hybrid RANS/LES methods: Applications and guidelines*. Imperial College Press. https://doi.org/10.1142/p878

Santner, T. J., Williams, B. J., & Notz, W. I. (2003). *The design and analysis of computer experiments*. Springer New York. https://doi.org/10.1007/978-1-4757-3799-8

Slotnick, J. P., Khodadoust, A., Alonso, J. J., Darmofal, D. L., Gropp, W. D., Lurie, E. A., & Mavriplis, D. J. (2014). CFD vision 2030 study: A path to revolutionary computational aerosciences. In *NASA Technical Report* (NASA/CR-2014-218178).

Smith, R. C. (2013). *Uncertainty quantification: Theory, implementation, and applications*. Society for Industrial; Applied Mathematics. ISBN: 9781611973211

Sobol, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, *55*(1), 271–280. https://doi.org/10.1016/S0378-4754(00)00270-6

Xiu, D. (2010). *Numerical methods for stochastic computations: A spectral method approach*. Princeton University Press. https://doi.org/10.2307/j.ctv7h0skv

Xiu, D., & Karniadakis, G. E. (2002). The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, *24*(2), 619–644. https://doi.org/10.1137/S1064827501387826