Democratization of uncertainty quantification in industrial design

C. Ludwig, D. Loukrezis, S. Boschert
Technology, Siemens AG
Otto-Hahn-Ring 6, 81739 Munich, Germany

Abstract

This work discusses the issue of uncertainty in industrial design and presents an intuitive workflow which allows non-experts to set up uncertainty quantification (UQ) studies, deploy computational UQ methods, and assess their results. To make UQ accessible to non-experts, an intuitive workflow which simplifies the definition of uncertain parameters and probabilistic calculations has been implemented. The implementation supports the assignment of probability distributions to model parameters via intuitive software dialogues. The distributions are then taken into account by sampling based UQ algorithms yielding outputs that reflect the effect of model input uncertainties onto the simulation output. In post-processing, 3D visualization techniques, tailored to UQ, help engineers to rapidly assess their design in terms of performance and robustness. An exemplary test case demonstrates how this framework can help simulation engineers to intuitively assess the most critical characteristics of their product by taking into account uncertain model properties.

1 Introduction

Traditionally, simulation for industrial design is mainly focused on the computation of deterministic models. However, quantifying uncertainties is becoming increasingly important for the efficient and robust design of industrial products [1, 2, 3]. One reason behind the increasing need of uncertainty quantification (UQ) in industrial design is the fact that simulation models are idealized representations of reality which rely on some level of abstraction and simplification. Therefore, they can never be fully capable of accurately describing real world phenomena. Additionally, even if a model is considered sufficiently accurate for design purposes, there is always the possibility of random and difficult or impossible to control deviations between design parameters and their real world realizations, e.g., due to manufacturing tolerances. Such uncertainties can result in products that deviate from their set requirements, possibly leading to sub-optimal performance or even failure.

A major bottleneck that prevents employing UQ studies during the design phase and benefiting from their results is the lack of UQ tools, methods, and frameworks suitable for non-experts, for example, accessible by simulation engineers skilled in computer aided design (CAD) and computer aided engineering (CAE), but with limited knowledge in statistics and probability. This work contributes towards this direction by showcasing how an intuitive and simple to use framework can enable the seamless integration of UQ within industrial design. To that end, we present a tool-chain that enables the design engineer to set up a UQ study, run UQ algorithms, and evaluate their results, with minimum or no prior knowledge of these concepts. Our goal is to lower the current barrier to the use of UQ methodologies, essentially democratizing UQ in industrial design. In that way, we hope to change conventional design principles for industrial products, but also further advance research in UQ by creating a direct link between academic practice and industrial demand.

In particular, with respect to setting up UQ studies, the implemented framework offers intuitive software dialogues which guide the user towards assigning probability distributions to model parameters that are considered to be uncertain. Then, the prescribed distributions are employed within sampling based UQ algorithms which are based on model evaluations for a sample of parameter realizations drawn from the assigned probability distribution. In this work we use the non-intrusive variant of the polynomial chaos expansion (PCE) method [4, 5], which usually achieves significant computational gains compared to the more commonly employed Monte Carlo method [6, 7, 8]. This workflow proves to be significantly advantageous in terms of simplicity and usability compared to tediously setting up parameter studies. The resulting deterministic output fields can then be suitably post-processed as to reflect the effect of model input uncertainties onto the simulation output, where the latter is commonly called the quantity of interest (QoI).

During the post-processing step, we additionally present 3D visualization techniques suitable for representing output uncertainty, allowing engineers to rapidly assess their design in terms of performance and robustness. For scalar QoIs, many visualization techniques such as histograms and box plots are typically available [9]. However, the inspection of the full 3D field results directly overlaid onto the CAD model is often desired, for example, to analyze temperature distributions, animate node movement, or identify critical stress regions. Visualization methods that are suitable for such purposes have been identified in the literature on uncertainty visualization [10, 11]. This work demonstrates how such methods can be included in the proposed UQ framework and help simulation engineers to intuitively assess the most critical characteristics of their product by taking into account uncertain model properties.

The remainder of this work is organized as follows. In Section 2, we present an overview of typical uncertainty sources in industrial product design. In Section 3, we introduce existing challenges of including such uncertainties into industrial applications before proposing a basic workflow to approach this task in Section 4. Section 5 concludes the work giving an outlook on future integration of UQ into CAE workflows.

2 Uncertainties in technical systems - an overview

The main sources and types of uncertainty that arise in the modeling and simulation of engineering and other technical systems are presented in Section 2.1. Such uncertainties are commonly attributed to either incomplete knowledge about the modeled system, leading to modeling assumptions and simplifications, or to uncontrollable random variations in the model parameter values. Uncertainty sources falling under the former category are called *epistemic* or systematic, while the latter category contains so-called *aleatory* or statistical uncertainties [12], as discussed in Section 2.2. It must be noted that this distinction is not always possible. For better understanding, in Section 2.3 we use the well-known model problem of a loaded cantilever beam to describe and categorize the uncertainties considered in this particular test case.

2.1 Uncertainty sources and types

The following is a non-exhaustive list of the main sources and types of uncertainty that are typically taken into consideration when modeling and simulating technical systems. Note that the mentioned uncertainty sources and types are not always possible to be discerned from others, i.e., in many cases the encountered uncertainties are attributed to more than one source. For example, consider the scenario where an inherently uncertain model parameter is modeled as a random variable random variable (RV), the probability distribution of which is determined using a set of measurement data. In this case, parameter uncertainty, measurement uncertainty, stochastic modeling uncertainty, and interpolation uncertainty, all of which are discussed next, are simultaneously present, interconnected, and affecting both one-another as well as the model and its predictions.

Parameter uncertainty This is the most commonly encountered uncertainty in the context of modeling and simulation. It is attributed to the inherent variability of certain input model parameters, which cannot be reduced or controlled. Exemplary parameter uncertainties include random variations in the dimensions of a device component or in its material properties due to the unavoidable tolerances that exist within the manufacturing process. Uncertain parameters of this type are typically modeled as *random variables* or *random fields*.

Stochastic modeling uncertainty Stochastic modeling uncertainty arises when modeling uncertain parameters as above without accurate information about their statistical properties (e.g. mean, variance) or without information about the appropriate choice of the distribution itself. The lack of this knowledge introduces additional uncertainty into the model. As such, stochastic modeling uncertainty can be interpreted as a form of model-form uncertainty, discussed next, where now the term "model" does not refer to the idealized description of the system under investigation, but rather to the stochastic model which is used to describe the parameter uncertainty.

Model-form uncertainty Also referred to as model bias, discrepancy, or inadequacy, this form of uncertainty concerns the discrepancy between the real-world system under investigation and the corresponding model, due to assumptions, simplifications, and other types of approximation. For instance, the lack of knowledge regarding the true underlying physics governing the problem at hand necessarily leads to modeling assumptions, thus to model-form uncertainty as well.

Observation uncertainty Also known as measurement uncertainty, it refers to the errors in the measurements and observations that are used to estimate model parameters, e.g., due to measurement equipment tolerances, irreproducible measurement conditions, or approximation errors in indirect measurements. Note that observation uncertainty is often defined broadly enough as to include interpolation and numerical uncertainty, which are discussed next.

Interpolation uncertainty Interpolation uncertainty is connected to the lack of sufficient observations needed for making an informed decision. In this case, interpolation or even extrapolation must be performed to predict missing values, thus leading to this type of uncertainty.

Numerical uncertainty Also referred to as discrete or algorithmic uncertainty, it concerns the numerical approximation errors that arise from the transformation of the mathematical model into a computational one, e.g., accuracy limitations due to prescribed discrete meshes or grids.

2.2 Uncertainty categories: epistemic, aleatory, and mixed

Next we discuss the main categories of uncertainty, namely, epistemic, aleatory, and mixed-type, where the latter category is a combination of the first two.

2.2.1 Epistemic uncertainty

The sources and types of uncertainties characterized as epistemic or systematic are those that could in principle be reduced or even eliminated by additional knowledge or information that is currently lacking. The term derives from the Greek word *episteme* which is defined as verified and documented knowledge. For example, numerical and measurement uncertainties can be regarded as epistemic, since a more accurate computational model or measurement device would reduce them. However, this model or device might not be available at the time. Since any kind of uncertainty can in principle be attributed to the lack of knowledge and information, there exists the philosophical question as to whether all uncertainties are eventually epistemic [12]. Nevertheless, considering models and simulations of prescribed accuracy, the category of aleatory uncertainty, discussed next, becomes very much relevant.

2.2.2 Aleatory uncertainty

The category of aleatory or statistical uncertainty includes all unknown or inexactly known parameters, properties, and attributes, the value of which differs each time the same experiment is run. The term is derived after the Latin word *alea*, referring to chance, risk, or dice-games. The main characteristic of this

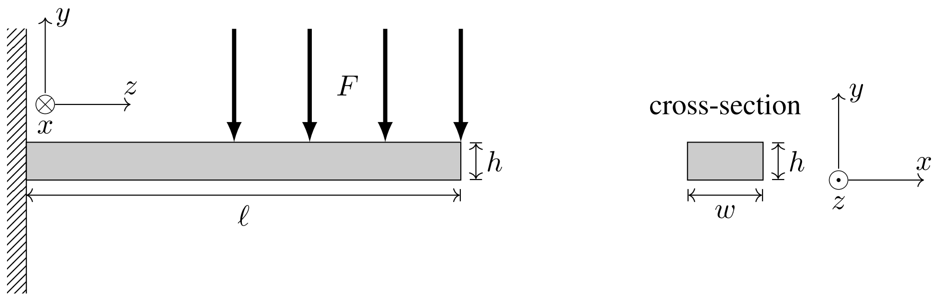

Figure 1: Sketch of a loaded cantilever beam with single fixed support.

category is that the uncertainty is considered to be inherent, unavoidable, and irreducible. For example, in an industrial context, manufacturing imperfections caused by hardware tolerances, e.g., due to complicated machine vibrations or uncontrollable ambient conditions, can be seen as aleatory uncertainties. As also noted above, the term “irreducible” is somewhat problematic, as it only refers to the current state of things. Nevertheless, from a modeling and simulation perspective, aleatory uncertainties are actually the ones most commonly taken into consideration in a UQ context.

2.2.3 Mixed-type uncertainty

In numerous situations, an uncertainty can be characterized as epistemic and aleatory at the same time. Consider for example the case of a model fitted to observations obtained with a fixed-accuracy measurement device. The measurements in this case entail aleatory uncertainty (cannot be reduced due to the unavailability of more accurate equipment). The fitted model then suffers from epistemic uncertainty with respect to its form, which is in fact dependent on the aleatory uncertainty in the observations.

2.3 Illustrative test case: cantilever beam

We consider a rectangular cantilever beam with single fixed support, upon which a force load F is applied, see Fig. 1. The geometry of the beam is described by its length ℓ, width w, and height h. The beam's modulus of elasticity, also known as the Young's modulus, is denoted by E and is a property that characterizes the material of the beam. Typical QoIs to be computed given specifications regarding the beam's geometry, material composition, and applied force load, are the beam deflection δ and the stress σ, which can then be evaluated and assessed in view of engineering targets and requirements, e.g., in the context of structural reliability and safety. Note that all dependencies on the spatial coordinates have been omitted for brevity of notation. The computation of the QoIs can be of varying complexity depending on the particularities of the problem at hand, the corresponding modeling assumptions, and accuracy requirements. That is, models ranging from simple analytical formulas to sophisticated finite element (FE) simulations can be employed for computing the QoIs. In this context, a number of uncertainties and their sources can be identified and categorized as epistemic, aleatory, or mixed.

The geometrical and material parameters of the beam, as well as the force applied to it, can take the form of aleatory uncertainties and be modeled as either random variables or random fields. The latter option applies to the Young's modulus E and the force load F, assuming that their variability is spatially dependent. It could also apply to the cross-section parameters w and h, considering a tapered beam. The exact area of load application can also be considered as an aleatory uncertainty in certain situations, as deviations are to be expected each time the force is applied to the beam. In all cases, these are parameter uncertainties.

On top of the aforementioned parameter uncertainties, stochastic modeling uncertainties regarding the random variables and random fields chosen to represent the uncertain parameters might apply. The latter uncertainties are of epistemic nature. The load distribution and the Young's modulus can also be considered to be epistemic or mixed. For example, the employed material model might be based on specific assumptions regarding the beam's material composition, thus leading to uncertainty in the parameters of the material model or even to the choice of the material model itself. Accordingly, the load might be modeled as uniformly distributed on the area of application, which is most probably an idealized situation. In both cases, the uncertainty is attributed to model bias, but also possibly to observation and interpolation uncertainty, depending on how the modeling decisions have been met. Epistemic uncertainties can also be encountered in the assumptions made for the employed mathematical and computational model, e.g., regarding the linear or nonlinear elastic deformation of the beam.

3 UQ for industrial applications

In the previous section it was shown that randomness and variability of parameters is present in virtually all applications of interest. As a consequence, numerical simulation of physical phenomena should take into account the uncertainty that is inextricably linked with the underlying model. This includes epistemic as well as aleatory uncertainties. Today, however, simulation is mostly performed deterministically. Fixed inputs are provided to the simulation tool that, in turn, yields a deterministic output. Therefore, a major goal consists in enabling simulation engineers to easily apply UQ methods in their simulation tasks. In doing so, the important distinction between intrusive and non-intrusive methods arise [13].

3.1 Intrusive vs. Non-intrusive methods

Intrusive methods make use of the underlying equations directly. One example are Galerkin projection methods where an approximation to the sought solution is inserted into the differential equations yielding (deterministic) equations for the approximation coefficients. In contrast, *non-intrusive* methods solely rely on evaluating the simulation tool for a number of deterministic parameter configurations. Coefficients of the approximation of the solution can then be found using e.g. interpolation or collocation approaches. An overview over intrusive and non-intrusive methods including an assessment of advantages and disadvantages of both is given in [5].

The objective of this work is lowering the barrier of using UQ methods in conventional CAE workflows. As we do not want to limit ourselves to particular applications and software, we focus on non-intrusive methods that use existing solvers as black boxes.

3.2 Coupling simulation software to UQ tools

It is noteworthy that the widespread disregard of sophisticated UQ methods in the context of industrial simulation is not caused by lack of appropriate tools that would allow bringing those methods to commercial solvers. In fact, several tools exist that offer the possibility to apply UQ methods to simulation models that are given as black-box software (and hence are non-intrusive). Notable examples include UQpy [14] and EasyVVUQ [15]. For a more extensive overview over available tools we refer to [14]. The application of an external UQ tool generally requires different steps. The users first need to define the parameters they deem to have a variability that would affect the simulation results. Indications on which parameters should be considered are given in Section 2. For each parameter, an appropriate distribution has to be defined, resulting in a joint probability distribution of the set of uncertain parameters. Sampling based, non-intrusive algorithms rely on computing solutions for a variety of parameter samples. This task needs to be converted into input decks the solver can understand. All simulation results need then to be collected and analyzed to provide useful information to the user.

It can be seen that the process of coupling an external UQ tool to a simulation software typically involves quite a lot of (possibly manual) work and rather targets users with expertise in programming. Promoting the use of UQ in industry will require a more holistic and integrated approach that is appealing to simulation engineers that might lack this experience.

3.3 Visualization of probabilistic results

Besides extraction of values of interest, visual assessment of simulation results is an integral part of CAE workflows. A typical example arises in mode shape analysis of mechanical components. A 3D animation of the natural vibration of a system allows a much more intuitive interpretation of the simulation results than numerical quantities of interest such as maximum displacement. For 1D and 2D results, standard representations of probabilistic data exist (e.g., box plots, histograms, probability distributions etc.) but the visualization of 3D fields containing random data is still an active field of research. Some approaches are presented in [16, 17].

It seems today that UQ has not found its way into industrial simulation despite its immense potential. To address this issue, in the next section we present an approach to the integration of UQ into conventional CAE workflow activities such as graphical modeling and visual inspection of results.

4 A workflow for intuitive application of UQ

In this section we present an approach for the integration of UQ tools into conventional CAD/CAE workflow activities providing graphical stochastic modeling and visual inspection of UQ results. We start by collecting basic requirements that such a workflow needs to fulfill and continue by showing an exemplary implementation. Note that we do not intend to develop new algorithms, but rather aim at lowering the barrier for simulation engineers to use established UQ methods as much as possible.

4.1 Basic requirements for an efficient and intuitive workflow

From a user perspective, it is highly desirable to have the possibility to define all uncertainty related parameters during model creation, similar to the way of prescribing deterministic parameter values. In that way, uncertainties can be considered with respect to the geometrical, material, loading, or other parameters of a simulation model. However, the authoring tools for parameter configuration during model design have been developed with regard to the deterministic case only. Hence, all kinds of uncertain model inputs must currently be defined in a separate step and usually using a different tool.

Furthermore, the available UQ tools or libraries typically offer several different algorithms and implementations, which are best suited (from the numerical computation point of view) to specific problem classes. To choose the best suited algorithm requires in-depth knowledge of the underlying mathematics, which cannot and should not be expected from a non-expert user. For that reason, it is necessary to offer the user at least some assistance, e.g., regarding the choice of UQ algorithm based on the type of engineering problem at hand. Even better would be if the most suitable algorithm were chosen automatically in the background, i.e., without any user-interaction. Nevertheless, this approach does not exclude that on an expert-level setting some modifications or manual overrides could be allowed.

The most important part of any type of uncertainty analysis is to deliver results in a way that is easily understandable by non-experts. This is especially true for three-dimensional fields that depend on spatial coordinates. Scalar fields such as temperatures or mean stresses are to be distinguished from vector fields such as deformations. For the latter, results might be deformed along the vectors to provide a visual impression of the statistical variation. An example for the former will be given in Section 4.3.3.

4.2 General workflow

To execute an analysis which includes UQ as part of it, additional information is needed during the simulation set up, such as the aforementioned probabilistic description of the model parameters regarded as uncertain. To facilitate a smooth integration of UQ, these additional data should be provided along the simulation workflow which is commonly followed by the practitioner with as few changes as possible. We propose the following general workflow that is derived from typical procedures in setting up simulation models.

A simulation process starts by creating a geometry model or importing it from a CAD file. For the latter, some simplification actions such as closing holes, removing unnecessary details, etc., must be additionally performed. This is also the first point where UQ-related information must be entered, in particular concerning geometric uncertainties. Modern simulation tools offer the possibility to display the geometric object and select parts of it via the GUI in order to highlight or modify them. This would also allow to mark faces or edges of the geometry model, which have a relevant (for the current investigation) uncertainty and specify the necessary stochastic modeling parameters, e.g., a probability distribution. When modeling the geometry, the user directly interacts with the graphical representation of the model. Providing uncertainties of geometrical properties such as distances, diameters, angles, etc., can therefore be accomplished making use of user actions such as dragging points along edges or specifying axes of deformation graphically.

The next step in the simulation workflow is the assignment of materials and their properties. Typically, material properties are entered as deterministic values or chosen from a material library. Unfortunately, most material libraries also only yield deterministic values. A simple way to include uncertainties would be to provide the possibility of adding uncertainty parameters such as mean or variance right next to the deterministic values.

The last step in setting up the simulation is the definition of boundary conditions and loads. Normally this involves specifying numerical values (scalar or vector) for each condition. Similar to the preceding steps, uncertainties can be either introduced by prescribing numerical values for the statistical properties of a quantity (e.g., in the case of Dirichlet boundary condition) or by graphical interaction (e.g., varying the length and direction of a load vector).

Having defined the model, the user typically enters the numerical parameters for the solution (solution and solver type, convergence criteria, etc.). In this step, the necessary input for the UQ solver must be given in the same manner before the solver is finally started and the solutions are obtained. Additionally the post-processing of the results remains the same as in a classical workflow, with the only difference that additional results for UQ are available for the user.

4.3 An exemplary implementation

In this section we present an implementation of a workflow that covers some of the most important requirements listed in Section 4.2. We show the different aspects using a simple cantilever beam example, similar to the one presented in Section 2.3, modeled using 2D finite elements. A distributed load is applied to the tip of the beam in negative z-direction effectively bending it downwards. In this example we are interested in the stress distribution in the beam due to bending. We assume that the load magnitude is not deterministic but rather normally distributed with a mean value of $800\,\mathrm{N}$ and a standard deviation of $80\,\mathrm{N}$. Due to this variability the resulting von Mises stress field is a (dependent) RV as well. Denoting the spatial coordinates by x and the random load by ζ, the stress field is then denoted by $\sigma(x, \zeta)$. Note that, for the spatially discretized model, the solution stress vector contains a finite number of components, i.e., a single value for each element. To simplify notation, we will refrain from writing this stress vector explicitly. Instead, it should be understood that a relation containing the spatial variable x should hold for all (finitely many) elements.

4.3.1 Graphical user interface (GUI)

As mentioned before, model parameters should be declarable to be uncertain as intuitively as possible, e.g., directly within the GUI of the simulation tool. Figure 2 shows a possible realization of such a GUI. After the load is defined in the usual way, an additional dialog box can be opened that lets the user select any defined load and declare it to be a RV. Additionally, the type of probability distribution as well as the corresponding distribution parameters, e.g., mean and standard deviation in the case of a Gaussian distribution, are defined directly in this dialog box. The RVs and their distribution parameters are stored as internal expressions in the simulation software.

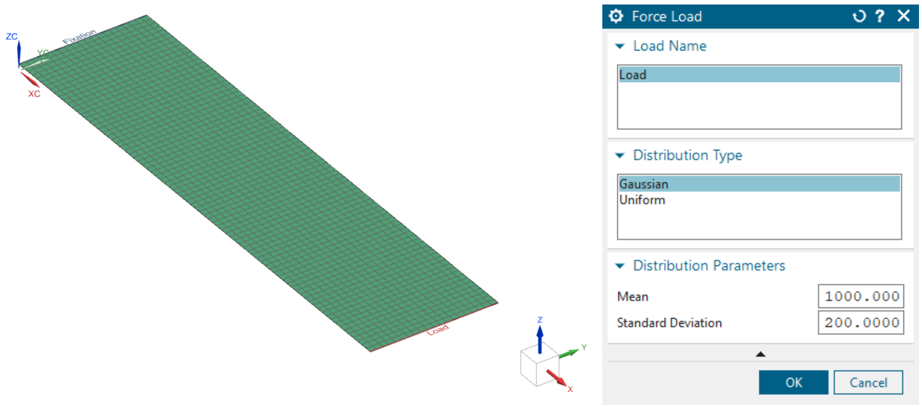

Figure 2: Left: The example cantilever beam model. A load on the free end pointing towards the negative z-direction is defined (red edge). The other end of the beam is fixed (blue edge). Right: Dialog box directly integrated into the model simulation setup. The load is selected and distribution type as well as distribution parameters can be defined.

4.3.2 UQ computation

As already described in Section 3.2, a great deal of effort is typically spent establishing the link between the simulation software and an external UQ solver. Therefore, any implementation of a UQ workflow targeting simulation engineers needs to automatically access those parameters and avoid manual interaction. This is easily realizable if the UQ calculations are part of the simulation software itself (in this case, intrusive methods may be used as well), but for coupling an external UQ solver, an application programming interface (API) for the extraction of parameters declared to be uncertain as in Section 4.3.1 needs to be readily available. Our implementation uses UQpy as external UQ tool and directly extracts the relevant internal expressions of the simulation software to create the corresponding probability distributions.

Next, we use polynomial chaos expansion (PCE) [4] to obtain an approximation of the random stress field of the form

$$\sigma(x, \zeta) = \sum_{q=1}^{Q} c_q(x) \Phi_q(\zeta), \tag{1}$$

where $c_q(x)$ are the PCE coefficients defined as

$$c_q(x) = \int \sigma(x, \zeta) \Phi_q(\zeta) \rho(\zeta) \, \mathrm{d}\zeta. \tag{2}$$

Here, $\rho(\zeta)$ denotes the probability density function (PDF) of ζ. The polynomials Φ_q are chosen according to the Wiener-Askey scheme [4] to fulfill the orthogonality condition

$$\int \Phi_q(\zeta) \Phi_p(\zeta) \rho(\zeta) \, \mathrm{d}\zeta = \delta_{qp}. \tag{3}$$

After drawing load samples $\zeta_1, \ldots, \zeta_p$ from the underlying load distribution, the corresponding solution fields $\sigma(x, \zeta_i), i = 1, \ldots, p$, are computed. To that end, the simulation software must provide an API to call the numerical solver with user-defined inputs and needs to make the results accessible. Usually this is realized in a way that inputs and outputs are written to files.

Equation (2) is usually impractical for computing the PCE coefficients as it contains the solution itself. Therefore, in our example the PCE coefficients were obtained using linear regression. Let $\mathbf{\Phi}$ denote a matrix with elements $\mathbf{\Phi}_{ij} = \Phi_j(\zeta_i)$ and $\sigma(x) = (\sigma(x, \zeta_1), \ldots, \sigma(x, \zeta_p))$. Then, the coefficient vector

$\mathbf{c}(x) = (c_1(x), \dots, c_Q(x))$ is obtained by minimizing

$$\mathbf{c}(x) = \arg\min_{\hat{\mathbf{c}}} \|\sigma(x) - \mathbf{\Phi}\hat{\mathbf{c}}\|_2^2. \tag{4}$$

Approximations of the mean μ and variance Var of the random stress field are then readily obtained by post-processing the PCE, i.e., by using the relations [5]

$$\mu(x) \approx c_1(x), \quad \mathrm{Var}(x) \approx \sum_{q=2}^{Q} c_q^2(x). \tag{5}$$

Note that, in order to avoid confusion, we do not use the common symbols σ, σ^2 for standard deviation and variance, respectively.

We stress that the computation of probabilistic solutions should be performed with minimum user interaction. This includes the selection of appropriate algorithms for the given problem. For the beam example, the analysis of a deterministic solution would typically include a visual assessment of the bending of the beam. The analysis of the probabilistic counterpart should therefore also include visualization aspects, which are addressed in the next section.

4.3.3 Visualization

As already mentioned, the visual assessment of simulation results presents an important component of post-processing. Significant research effort has been directed towards the scientific visualization of probabilistic results [18, 17, 16].

First, the user needs to assess 1D results, such as the variation in the maximum von Mises stress. Additionally, the user must be able to visually inspect the whole model including the stress field and its probabilistic variations. The former can be achieved by standard methods such as probability curves or box plots. Figure 3 shows yet another method: a histogram of 1500 sampled input loads alongside the corresponding results for the maximum von Mises stress in the cantilever beam. Unsurprisingly, the maximum stress is always located at an element that is adjacent to the fixed end of the beam. The information on the location of this element has to be conveyed alongside the plots, e.g., by highlighting the corresponding element, see Fig. 4.

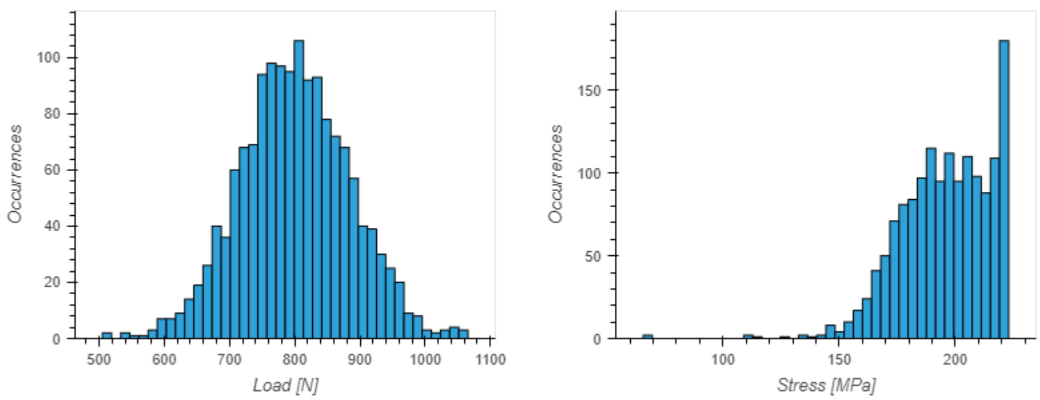

Figure 3: Left: Histogram of 1500 load samples drawn from the input distribution. Right: Corresponding maximum von Mises stresses.

Displaying 3D results is more challenging, as already noted in the past [18]. Nevertheless, several visualization techniques have been proposed. Choosing the most suitable one certainly depends on the complexity of the model, as well as on the type of data and variability (1D or 3D). For the beam example, the mean stress field and its standard deviation can be displayed using color coding, as shown in Fig. 5.

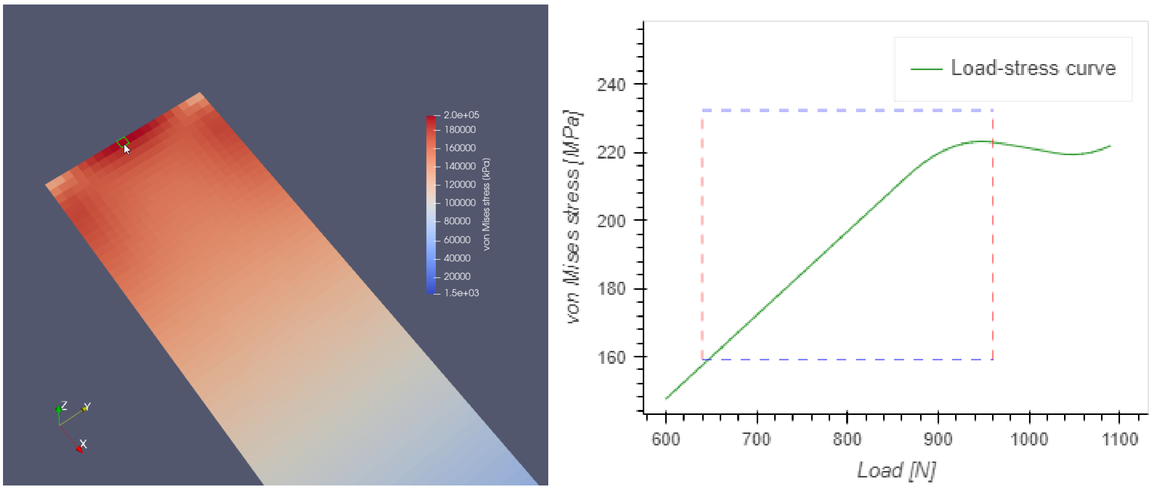

Figure 4: Selection of elements in post-processing. Here, the user selects the element with the highest von Mises stress (green highlight in the left image). Confidence intervals of the output are shown (blue borders in the right image) corresponding to twice the standard deviation on either side of the mean.

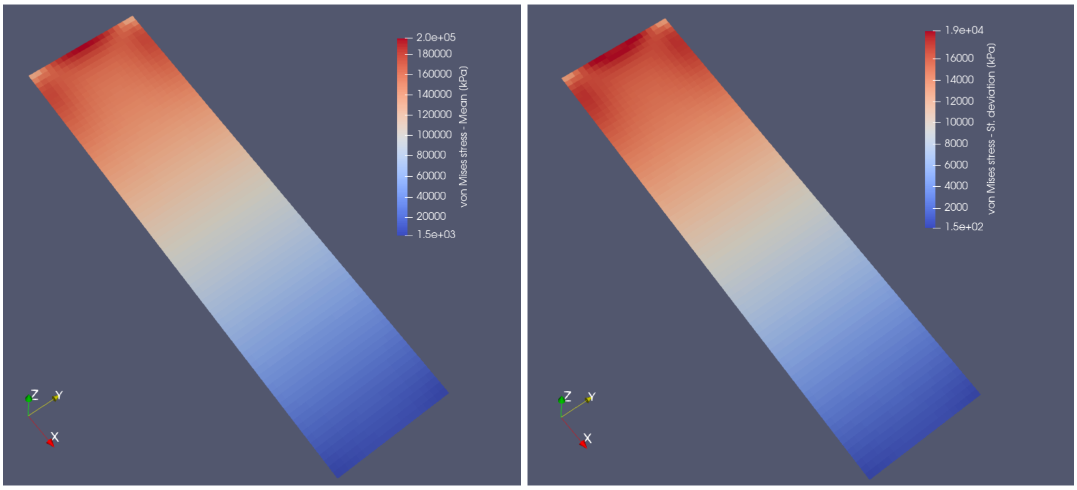

Figure 5: Left: Mean von Mises stresses on the beam are visualized. The color indicates the stress magnitude. Right: Standard deviation of the von Mises stresses.

Alternatively, a glyph based approach can be chosen, where the color indicates the mean stress and the standard deviation is illustrated by the size of semi-transparent glyphs. In Fig. 6, the glyphs are chosen as balls that are attached to each element. In the end result, the fuzzier areas indicate higher uncertainty regarding the stress value.

5 Discussion and outlook

In this work we presented a workflow for the intuitive integration and application of UQ in industrial simulation and engineering tool-chains. Based on the premise of the omnipresence of uncertainty in engineering design, we concluded that simulation software should make use of the powerful mathematical methods that are available to quantify the effect of model uncertainties on the simulation results. This observation has generally been ignored in the past, as conventional simulation tools mostly remain purely deterministic. This lead to the formation of a gap between academic research in UQ and its application for industrial problems, but also to a systematic overlook of the impact of uncertainty onto industrial design products.

We showed how this gap can potentially be bridged by integrating UQ straightforwardly and seamlessly into

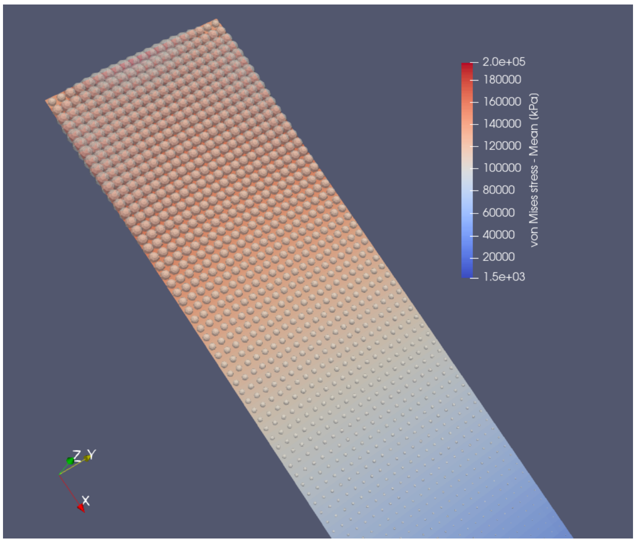

Figure 6: Simultaneous visualization of mean and standard deviation of the von Mises stresses in the beam. Color represents the mean whereas the size of the semi-transparent balls indicates the magnitude of the standard deviation. In this visualization, regions with high variability appear ”fuzzy”.

the conventional modeling and simulation process. Our workflow spans model generation via intuitive GUI dialogues, probabilistic solution via non-intrusive UQ methods performed by external solvers, and visual assessment of the probabilistic solution using different visualization formats.

To illustrate these steps we provided an exemplary implementation of this workflow using a simple cantilever beam model. Note that only a small set of features are covered by the implementation and there are many conceivable extensions that can further assist users in applying UQ methods.

During the modeling process, information on uncertainties could be extracted directly from metadata contained in the CAD files. A prerequisite is of course the availability of such data, i.e. the consideration of typical uncertainties, e.g., manufacturing tolerances, during CAD setup. Selection of suitable algorithms for UQ calculations based on model complexity, number of parameters, required accuracy, etc., is another extension that would significantly facilitate usage of UQ in CAE. If only rough predictions are needed, simple rules of thumb or heuristics could be proposed by the software. Finally, there is still a need for research in the field of scientific visualization of uncertain 3D data. Providing useful results in an easily conceivable way is a key for widespread adoption and requires further investigation.

The adoption of UQ methods in industrial design requires a shift of mind in the way engineers address the various aspects of simulation. At the moment, the current mind set is to solve problems deterministically. Our thesis is that, providing the means to use probabilistic methods within simulation software without significantly altering conventional simulation workflows will accelerate the usage of UQ, further build trust in simulation results, and eventually foster theoretical advancements and practical improvements by bridging the gap between academic research and industrial needs.

References

[1] B. Bergman, J. De Maré, T. Svensson, and S. Loren, *Robust design methodology for reliability: exploring the effects of variation and uncertainty*. John Wiley & Sons, 2009.

[2] C. Hirsch, D. Wunsch, J. Szumbarski, J. Pons-Prats *et al.*, “Uncertainty management for robust industrial design in aeronautics,” *Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Springer*, 2019.

[3] M. Rafiei and L. A. Ricardez-Sandoval, “Integration of design and control for industrial-scale applications under uncertainty: A trust region approach,” *Computers & Chemical Engineering*, vol. 141, p. 107006, 2020.

[4] D. Xiu and G. E. Karniadakis, “The wiener–askey polynomial chaos for stochastic differential equations,” *SIAM journal on scientific computing*, vol. 24, no. 2, pp. 619–644, 2002.

[5] D. Xiu, “Fast numerical methods for stochastic computations: A review,” *Communications in Computational Physics*, vol. 5, pp. 242–272, 2009.

[6] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo method.* John Wiley & Sons, 2016.

[7] I. M. Sobol, *A primer for the Monte Carlo method.* CRC press, 1994.

[8] J. Zhang, “Modern monte carlo methods for efficient uncertainty quantification and propagation: A survey,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 13, no. 5, p. e1539, 2021.

[9] K. Potter, J. Kniss, R. Riesenfeld, and C. R. Johnson, “Visualizing summary statistics and uncertainty,” in *Computer Graphics Forum*, vol. 29, no. 3, 2010, pp. 823–832.

[10] G.-P. Bonneau, H.-C. Hege, C. R. Johnson, M. M. Oliveira, K. Potter, P. Rheingans, and T. Schultz, “Overview and state-of-the-art of uncertainty visualization,” in *Scientific Visualization*. Springer, 2014, pp. 3–27.

[11] K. Potter, P. Rosen, and C. R. Johnson, “From quantification to visualization: A taxonomy of uncertainty visualization approaches,” in *Uncertainty Quantification in Scientific Computing*, A. M. Dienstfrey and R. F. Boisvert, Eds. Springer, 2012, pp. 226–249.

[12] A. Der Kiureghian and O. Ditlevsen, “Aleatory or epistemic? does it matter?” *Structural safety*, vol. 31, no. 2, pp. 105–112, 2009.

[13] F. Augustin, A. Gilg, M. Paffrath, P. Rentrop, and U. Wever, “Polynomial chaos for the approximation of uncertainties: chances and limits,” *European Journal of Applied Mathematics*, vol. 19, no. 2, pp. 149–190, 2008.

[14] A. Olivier, D. G. Giovanis, B. Aakash, M. Chauhan, L. Vandanapu, and M. D. Shields, “Uqpy: A general purpose python package and development environment for uncertainty quantification,” *Journal of Computational Science*, vol. 47, p. 101204, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877750320305056

[15] R. Richardson, D. Wright, W. Edeling, V. Jancauskas, J. Lakhlili, and P. Coveney, “Easyvvuq: A library for verification, validation and uncertainty quantification in high performance computing,” *Journal of Open Research Software*, vol. 8, 2020.

[16] K. Pöthkow and H.-C. Hege, “Nonparametric models for uncertainty visualization,” *Computer Graphics Forum*, vol. 32, no. 3pt2, pp. 131–140, 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12100

[17] S. Djurcilov, K. Kim, P. Lermusiaux, and A. Pang, “Visualizing scalar volumetric data with uncertainty,” *Computers & Graphics*, vol. 26, no. 2, pp. 239–248, 2002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0097849302000559

[18] C. R. Johnson and A. R. Sanderson, “A next step: Visualizing errors and uncertainty,” *IEEE Computer Graphics and Applications*, vol. 23, no. 5, pp. 6–10, 2003.