Uncertainty Quantification Module

User's Guide

COMSOL

Uncertainty Quantification Module User's Guide

Contact Information

Visit the Contact COMSOL page at www.comsol.com/contact to submit general inquiries or search for an address and phone number. You can also visit the Worldwide Sales Offices page at www.comsol.com/contact/offices for address and contact information.

If you need to contact Support, an online request form is located on the COMSOL Access page at www.comsol.com/support/case. Other useful links include:

- Support Center: www.comsol.com/support
- Product Download: www.comsol.com/product-download
- Product Updates: www.comsol.com/support/updates
- COMSOL Blog: www.comsol.com/blogs
- Discussion Forum: www.comsol.com/forum
- Events: www.comsol.com/events
- COMSOL Video Gallery: www.comsol.com/videos
- Support Knowledge Base: www.comsol.com/support/knowledgebase

C o n t e n t s

C h a p t e r   1 :   I n t r o d u c t i o n

C h a p t e r   2 :   U n c e r t a i n t y   Q u a n t i f i c a t i o n   T h e o r y

C h a p t e r   3 :   U n c e r t a i n t y   Q u a n t i f i c a t i o n   M o d e l i n g

Chapter 4: Glossary

1

Introduction

Welcome to the *Uncertainty Quantification Module User's Guide*. The capabilities of the Uncertainty Quantification Module can be used in conjunction with any combination of other COMSOL® products. This guide is a supplement to the *COMSOL Multiphysics Reference Manual*, with the present chapter containing an overview of the Uncertainty Quantification Module.

Uncertainty Quantification Module Overview

What Can the Uncertainty Quantification Module Do?

The Uncertainty Quantification Module can be used throughout the COMSOL product family. It provides a general interface for characterizing uncertainties, propagating input uncertainties in COMSOL Multiphysics models, and statistically analyzing output quantities of interest.

The Uncertainty Quantification Module can use any model parameters as inputs. The analyses performed are global in nature. In other words, the variations in the inputs need not be small. This ability makes the functionality very versatile for a number of investigation scenarios. A COMSOL model is, for most applications, a mathematical model of reality. Such a model will have inputs that are related to:

- The physics — For example, boundary conditions, sources, and coupling terms, such as the velocity magnitude or direction, for an inflow boundary condition in a transport problem.

- The materials — For example, material coefficients or details in their specification, such as Young's modulus for a structural mechanics problem.

- The geometry — For example, manufacturing specifications, such as a hole radius or plate thickness.

- The discretization — For example, mesh properties such as the maximum and minimum mesh element sizes. Another example is the relative tolerance used to solve a time-dependent problem.

For a particular model, it is often expected that the *variation* of a few inputs will have a significant influence on the results. This can be a perfectly natural effect of the design of the model and the physics involved. This should lead to an expected sensitivity in the results to the variation in these inputs. At the same time, it is also expected that other inputs have a small or even negligible influence; that is, there is an expected insensitivity to the results. The Uncertainty Quantification Module can be used to test the validity of such expectation and verify that some key input variations affect the key outputs in the model while other input variations do not. Most experienced modelers routinely perform these types of exercises before they can trust their model. When there are just a few inputs, this can be done, for example, by performing a

well-designed parametric sweep. In contrast, for more than a few parameters, this cannot be done in a simple and efficient way, and simply assuming that a model is correct can be dangerous. Uncertainty quantification is a convincing way to show the correctness of a model.

Furthermore, once you have verified the key parameters that account for most of the variations of the quantities of interest, you can take advantage of this knowledge to simplify further analysis by considering only those parameters.

The motivations for doing further, more rigorous uncertainty quantifications can be to answer questions like:

• Which of the key parameters are more important than others for my quantities of interest? Remember that a high sensitivity to a variation can be both a desirable and an undesirable property. For example, a sensor is often designed to be sensitive to a certain input signal; sensitivity is thus desirable. A hi-fi amplifier's characteristics should not be sensitive to the input frequency; sensitivity is here undesirable. How does a modification of the input uncertainties affect this? If the input uncertainty cannot be reduced (or increased), perhaps the underlying design can be altered to modify it to improve the sensitivity.

• How does the uncertainty distribution of input parameters translate to the uncertainty distribution of a quantity of interest? What are the confidence intervals for the quantities of interest?

• Given a nominal design and some specific uncertain inputs, what is the probability that the design fails? The failure can be a complete breakdown of the design, but it can also be phrased in terms of a quality criterion.

Given the available resources, these questions can be answered by performing different types of uncertainty quantification studies.

Where Do I Access the Documentation and Application Libraries?

Uncertainty quantification model examples that do not require any other COMSOL add-on modules are located in the Uncertainty Quantification Module folder in the Application Libraries window. To find all related examples, including any models in the add-on modules, if available, type `uncertainty quantification` in the search field at the top of the Application Libraries window.

A number of online resources have more information about COMSOL, including licensing and technical information. The electronic documentation, topic-based (or context-based) help, and the Application Libraries are all accessed through the COMSOL Desktop.

> If you are reading the documentation as a PDF file on your computer, the blue links do not work to open an application or content referenced in a different guide. However, if you are using the Help system in COMSOL Multiphysics, these links work to open other modules, application examples, and documentation sets.

THE DOCUMENTATION AND ONLINE HELP

The *COMSOL Multiphysics Reference Manual* describes the core physics interfaces and functionality included with the COMSOL Multiphysics license. This book also has instructions on how to use COMSOL Multiphysics and how to access the electronic Documentation and Help content.

Opening Topic-Based Help
The Help window is useful as it is connected to the features in the COMSOL Desktop. To learn more about a node in the Model Builder, or a window on the Desktop, click to highlight a node or window, then press F1 to open the Help window, which then displays information about that feature (or click a node in the Model Builder followed by the **Help** button (?)). This is called *topic-based* (or *context) help*.

> **Win**
>
> To open the **Help** window:
>
> - In the **Model Builder**, **Application Builder**, or **Physics Builder**, click a node or window and then press F1.
> - On any toolbar (for example, **Home**, **Definitions**, or **Geometry**), hover the mouse over a button (for example, **Add Physics** or **Build All**) and then press F1.
> - From the **File** menu, click **Help** (?).
> - In the upper-right corner of the COMSOL Desktop, click the **Help** (?) button.

	To open the **Help** window:
(Mac) (Linux)	• In the **Model Builder** or **Physics Builder**, click a node or window and then press F1. • On the main toolbar, click the **Help** (?) button. • From the main menu, select **Help>Help**.

Opening the Documentation Window

	To open the **Documentation** window:
(Win)	• Press Ctrl+F1. • From the **File** menu, select **Help>Documentation** (▮).

	To open the **Documentation** window:
(Mac) (Linux)	• Press Ctrl+F1. • On the main toolbar, click the **Documentation** (▮) button. • From the main menu, select **Help>Documentation**.

THE APPLICATION LIBRARIES WINDOW

Each model or application includes documentation with the theoretical background and step-by-step instructions to create a model or application. The models and applications are available in COMSOL Multiphysics as MPH-files that you can open for further investigation. You can use the step-by-step instructions and the actual models as templates for your own modeling. In most models, SI units are used to describe the relevant properties, parameters, and dimensions, but other unit systems are available.

Once the Application Libraries window is opened, you can search by name or browse under a module folder name. Click to view a summary of the model or application and its properties, including options to open it or its associated PDF document.

🔍	The Application Libraries Window in the *COMSOL Multiphysics Reference Manual*.

Opening the Application Libraries Window
To open the **Application Libraries** window (▥):

Win	From the **File** menu, select **Application Libraries**. To include the latest versions of model examples, from the **File>Help** menu, select (▦) **Update COMSOL Application Library**.

Mac Linux	Select **Application Libraries** from the main **File** or **Windows** menus. To include the latest versions of model examples, from the **Help** menu, select (▦) **Update COMSOL Application Library**.

CONTACTING COMSOL BY EMAIL

For general product information, contact COMSOL at info@comsol.com.

COMSOL ACCESS AND TECHNICAL SUPPORT

To receive technical support from COMSOL for the COMSOL products, please contact your local COMSOL representative or send your questions to support@comsol.com. An automatic notification and a case number will be sent to you by email. You can also access technical support, software updates, license information, and other resources by registering for a COMSOL Access account.

COMSOL website	www.comsol.com
Contact COMSOL	www.comsol.com/contact
COMSOL Access	www.comsol.com/access
Support Center	www.comsol.com/support
Product Download	www.comsol.com/product-download
Product Updates	www.comsol.com/support/updates
COMSOL Blog	www.comsol.com/blogs
Discussion Forum	www.comsol.com/forum
Events	www.comsol.com/events
COMSOL Application Gallery	www.comsol.com/models
COMSOL Video Gallery	www.comsol.com/video
Support Knowledge Base	www.comsol.com/support/knowledgebase

2

Uncertainty Quantification Theory

This chapter discusses the theory for uncertainty quantification.

Uncertainty Quantification Theory

This section contains theory useful for understanding and applying uncertainty quantification studies. Topics explained in this section:

- Data sampling methods
  - The *Latin hypercube sampling* (LHS) method is used to generate sample data with good space filling in the input parameter space. In addition, sequential LHS is used to generate additional sample points based on existing samples in an uncertainty quantification (UQ) study that requires more quantities of interest (QoIs).
  - The *Morris sampling* method is used to generate sample points for the Morris one-at-a-time (MOAT) method, where the next sample point only varies in one dimension from the current point of the input parameter space within each trajectory.
  - The *importance sampling* method is used to do Monte Carlo analysis with a multimodal sampling density created near the region where the criteria for reliability analysis are satisfied in the input parameter space.
  - Input distributions are used to set up the input parameters in a UQ study. This module supports a wide range of continuous independent distributions.
- Surrogate models
  - A key feature of the Uncertainty Quantification Module is its ability to train and use a so-called surrogate model, also known as a *metamodel*, for a particular UQ analysis to save computation resources. A surrogate model is a compact mathematical model constructed to represent and evaluate the QoIs in the domain of interest defined by the input parameters. This model is completely independent of the underlying COMSOL model and can, when trained properly, be used instead of the COMSOL model to predict values for the QoIs for other values of the input parameters than those solved for. The surrogate model evaluation (or prediction) has a very low cost. This is of paramount importance for UQ analysis when a Monte Carlo-type analysis is being used because a large number of evaluations are often required to achieve high accuracy. This is particularly true for more realistic problems where a COMSOL model evaluation

might require significant resources and where the UQ analysis involves several parameters.

- The *sparse polynomial chaos expansion* (SPCE) and *adaptive sparse polynomial chaos expansion* (ASPCE) models are used to model QoIs where each QoI is expanded with multivariate orthonormal polynomials that are specified for a particular type of probability distribution of input parameters. SPCE and ASPCE models are most commonly used for sensitivity analysis where the Sobol indices can be directly computed from the coefficients in the surrogate model.

- The *Gaussian process* (GP) and *adaptive Gaussian process* (AGP) are popular surrogate models widely used for sensitivity analysis, uncertainty propagation, and reliability analysis. GP and AGP are probabilistic in that they provide the variance of the prediction at each point sampled from the input parameter space.

- Screening
  - The *Morris one-at-a-time* (MOAT) method is a lightweight global screening method that gives a qualitative measure of the importance of each input parameter. The method is purely sample based and does not require a surrogate model. This is an ideal method when the number of input parameters is too large to allow the application of more computationally expensive UQ studies.

- Sensitivity analysis
  - The *Correlation* method is a widely used method for determining the linear relationship between each input parameter and the QoI. The method is purely sample based and does not require a surrogate model. There are four types of correlations available: bivariate correlation (also known as Pearson's correlation), rank bivariate correlation (also known as Spearman's rank correlation), partial correlation, and rank partial correlation.

  - The *Sobol* method (or variance decomposition method, analysis of variance (ANOVA)) looks at the entire input parameter distribution and decomposes the variance of each QoI into a sum of contributions from the input parameters and their interactions. In general, the Sobol method is a sample-based method and requires a large number of samples to achieve good accuracy. Here, the first-order Sobol index and the total Sobol index based on surrogate models are computed.

- Uncertainty propagation
  - Uncertainty propagation, also known as forward propagation of uncertainty, maps uncertainty of the input parameters to uncertainty of the QoIs. A Monte Carlo analysis is conducted based on a surrogate model to generate the approximated probability density function (PDF) of each QoI.

- Reliability analysis
  - Reliability analysis computes the probability that the QoIs satisfy the reliability criteria defined with thresholds corresponding to the QoIs. The *efficient global reliability analysis* (EGRA) method is a global reliability analysis method that uses an AGP model to reduce the number of COMSOL model evaluations by only constructing a model that accurately separates the region where the reliability criteria is satisfied from the region where such criteria is not satisfied.

- Inverse uncertainty quantification
  - Inverse uncertainty quantification computes the posterior distribution of input parameters — the calibration parameters that best reflects their distribution based on the experimental data and the prior knowledge on the calibration parameters. The *Markov chain Monte Carlo* (MCMC) method infers the posterior distribution of the calibration parameters by constructing Markov chains whose stationary distribution is the posterior distribution. This analysis is called inverse because it propagates information of experimental data backwards to gain information on the calibration parameters.

Data Sampling — Latin Hypercube Sampling

The accuracy of the UQ analysis and the surrogate model approximation is directly related to the data sampling for model evaluations. A Latin hypercube sampling (LHS) is a design given by an $n \times m$ matrix in which each column is a random permutation of $\{1, 2, \ldots, n\}$. Here, n is the number of model evaluations, and m is the number of input parameters. A LHS has good projection properties on any single dimension, which means that, in any one-dimensional input parameter space, the sampled data always has a good space-filling property. For UQ studies using LHS for input parameter sampling, an optimal LHS is generated for n number of model evaluations in the $[0, 1]^m$ hypercube and then mapped to the input parameters' probability distribution. An optimal LHS is space filling, which means that the sample points should spread over the entire input parameter space as evenly as possible. Meanwhile, it has no replications and sample points clustering in the projection onto any subspaces of the input parameter space. This is important because many problems could have only a subset of significant input parameters. The process of constructing an optimal LHS is formulated as a global optimization problem. The Uncertainty Quantification Module provides an optimal LHS algorithm that is efficient both in finding the global optimum and in computing the optimality criteria in every iteration. It uses the global optimization method and centered L_2 discrepancy criterion described in Ref. 1. Note that depending on the starting design, the optimal LHS obtained from the

optimization could be of lower quality. Therefore, the global optimization is repeated from different starting LHS designs.

In addition to constructing optimal LHS for a one-stage study, it is important to generate sequential (also known as follow-up) optimal LHS based on the current sampling for both ASPCE surrogate models and appending more model evaluation to improve a UQ analysis. The optimization of the sequential sampling uses the same criterion as for optimizing the initial optimal LHS. During the optimization, the sequential sampling does not modify the initial samples and tries to spread the sequential samples away from them. Because it is computationally expensive to run many model evaluations, it is possible to use any number of model evaluations for sequential sampling following the sequential optimization method described in Ref. 2.

For both initial and sequential sampling, you can choose the number of restart points and maximum number of iterations for optimizing LHS where more computational resources are required if a larger number is used. Note that an optimal LHS is more crucial when the number of model evaluations is small.

Data Sampling — Morris Sampling

The Morris sampling method is a special sampling method used for the Screening, MOAT UQ analysis. The Morris sampling is generated on a $[0, 1]^m$ hypercube and then mapped into the input parameter distribution. The MOAT method varies one input parameter at a time, where each input parameter is constructed to have a discrete number of values, called levels. The value at each level is in the set $[1/(n - 1), 2/(n - 1), ..., 1 - 1/(n - 1)]$, where n is the number of levels. The entire m-dimensional input parameter space is discretized into an n-level grid in each dimension. The data sampling starts by randomly selecting a base value, and each component in $\mathbf{x}$ is sampled from the set $[1/(n - 1), 2/(n - 1), ..., 1 - 1/(n - 1)]$. Note that the base value vector is only used to generate other sampling points. The next sampling point is obtained by increasing or decreasing one component of $\mathbf{x}$ by Δ, defined as

$$\Delta = \frac{n}{2(n - 1)} = \frac{1}{2} + \frac{1}{2(n - 1)}$$

The choice between increasing or decreasing Δ is conditioned by $\mathbf{x}$ still being in the $[0, 1]^m$ hypercube. Starting from a base value, m sampling points are added by moving in one dimension at a time; the path following all the sampling points is called

a trajectory. A schematic plot of two Morris trajectories of a 3D input space consisting of p_1, p_2, and p_3 is shown in Figure 2-1.
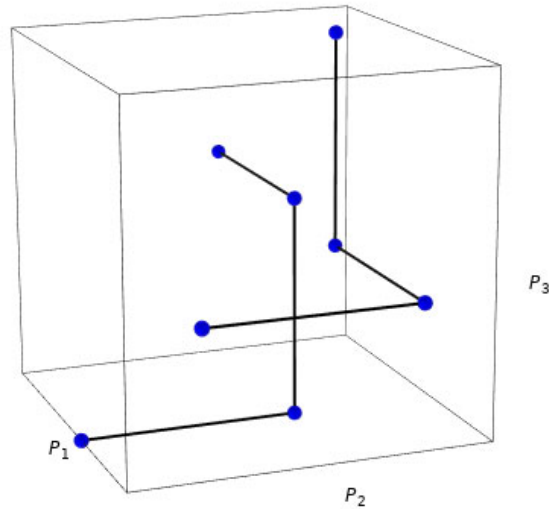
Figure 2-1: Morris trajectories of a 3D input space.

> The MOAT method is based on the construction of r trajectories in the input parameter space. The design is based on generating a random starting point for each trajectory and then completing it by moving one factor at a time in random order.

The Uncertainty Quantification Module uses the sampling method designed in Ref. 3, which gives better coverage of the entire input parameter space. The method selects r trajectories together in a way that is designed to maximize their dispersion in the input space. It starts with a large number of different trajectories and chooses the trajectories with the highest "spread", which is defined based on the distances between the sample points in the trajectories.

> When the dimension m of the input parameter space is low, there could be repeated sample points in different trajectories. The repeated samples are only evaluated once during the COMSOL model evaluation.

Data Sampling — Importance Sampling

Importance sampling is used in reliability analyses that contain an importance region (the region defined by the relationship between the QoIs and the thresholds for a reliability analysis) in the input parameter space. Although Monte Carlo analysis can be applied to reliability analysis with desired accuracy, the response of the limit state is analyzed with a large number of samples, which can be time consuming and expensive, especially for problems with low probability of importance. In reliability analysis, by default, a multimodal adaptive importance sampling method is used in combination with the AGP surrogate model. The AGP model provides information about the initial location of the importance region in the input parameter space, which ensures efficiency in constructing the multimodal sampling density. The importance sampling method starts with using the representative points (the points located near the limit state where the QoIs are equal to the thresholds) from adaptive GP with the highest probability density to iteratively construct a multimodal sampling density with the representative points. After the multimodal sample density converged, the importance sampling method iteratively samples on the multimodal sampling density function and computes the probability of the reliability analysis until convergence. The probability computed with importance sampling is defined as

$$p_f = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{\varphi(x_i)} I(x_i)$$

where x_i are the points sampled with the multimodal sampling density, N is the number of samples, φ is the multimodal sampling density, f is the original input distributions density, and I is an operator that evaluates to 1 if the reliability criteria is satisfied and 0 otherwise. You can specify the initial number of samples, maximum number of samples, and relative tolerance for the importance sampling method.

If the relative tolerance is not satisfied for the importance sampling, you can try to increase the initial number of samples and the maximum number of samples. Given that an accurate AGP model has been built for the problem, the importance sampling method can be used without recomputation of COMSOL model evaluations. More details about the importance sampling method are described in Ref. 4.

Data Sampling — Input Distributions

Probability distributions are used for specifying input parameters used for sampling input data points used in model evaluations and Monte Carlo-type analyses as well as

in response surfaces. For all UQ analysis types and methods, you can use uniform, normal (Gaussian), log-normal, gamma, beta, Gumbel, and Weibull distributions for the inputs. For these distributions, except the uniform and beta distributions, lower and upper bounds are automatically computed based on cumulative distribution function levels. There is also a possibility to manually specify the bounds.

- The uniform distribution is defined by the upper and lower bounds, and the *probability density function* (PDF) of the uniform distribution

$$f(x) = \frac{1}{(b-a)}, x \in [a, b].$$

- The normal distribution is defined by the mean μ and the standard deviation σ, and the PDF of the normal distribution is

$$(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\sqrt{\frac{(x-\mu)^2}{2\sigma^2}}\right), x \in (-\infty, \infty).$$

- The log-normal distribution is defined by the mean μ and the standard deviation σ, and the PDF of the log-normal distribution is

$$(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\sqrt{\frac{(\ln x - \mu)^2}{2\sigma^2}}\right), x \in (0, \infty).$$

- The gamma distribution is defined by the shape parameter k and the scale θ, and the PDF of the gamma distribution is

$$f(x) = x^{k-1} \frac{\exp\left(-\frac{x}{\theta}\right)}{\theta^k \Gamma(k)}, x \in (0, \infty).$$

- The beta distribution is defined by the shape parameter α, the shape parameter deviation β, the upper and lower bounds a and b, respectively, and the PDF of the beta distribution is

$$f(x) = \frac{(x-a)^{\alpha-1}(b-x)^{\beta-1}}{(b-a)^{\alpha+\beta-1}B(\alpha, \beta)}, x \in [a, b].$$

- The Weibull distribution is defined by the shape parameter k and the scale λ, and the PDF of the Weibull distribution is

$$(x) = \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} \exp\left(-\left(\frac{x}{\lambda}\right)^k\right), x \in (0, \infty).$$

- The Gumbel distribution is defined by the location μ and the scale β, and the PDF of the Gumbel distribution is

$$f(x) = \frac{1}{\beta}\exp\left(-\left(\frac{x-\mu}{\beta} + \exp-\left(\frac{x-\mu}{\beta}\right)\right)\right), x \in (-\infty, \infty).$$

These input distributions assume that the input parameters are independent. During the Latin hypercube sampling, Morris sampling, Monte Carlo sampling, as well as the importance sampling, the algorithms always first sample data in a unit interval for each parameter, $\phi_i \in [0, 1]$, where $i = 1, \ldots, m$, and m is the number of input parameters. The sampling can therefore be seen as selecting points in a unit hypercube of dimension m, $[0, 1]^m$. The boundaries of the hypercube correspond to the bound values for each physical parameter value p_i. The *cumulative distribution function* (CDF) F_i and its inverse F_i^{-1} are used to map the values between the unit interval to the input parameter values such that

$$p_i = F_i^{-1}(F_i(p_{i,l}) + \phi_i(F_i(p_{i,u}) - F_i(p_{i,l})))$$

where $p_{i,l}$ and $p_{i,u}$ are the lower and upper bounds, respectively. For the uniform distribution, this takes the simple form $p_i = p_{i,l} + \phi_i(p_{i,u} - p_{i,l})$. The lower and upper bounds can be given manually in the user interface. However, for all distributions, except uniform and beta, where manual bounds are needed, they are by default computed automatically from a CDF level. For example, if a lower bound CDF level of 0.1% is used, the lower bound is computed by the corresponding inverse CDF function $p_{i,l} = F_i^{-1}(0.001)$.

If there are existing QoIs or important input parameter points that need to be evaluated, specified values can be used as the input parameter points when a GP-type surrogate model is selected. For UQ analysis computed with the specified values, the input parameter space is a space defined by the maximum and minimum values of the specified values in each dimension, which is used for the Monte Carlo-type analysis associated with the study. Note that specified values can also be used for the correlation method in sensitivity analysis.

Surrogate Models — Polynomial Chaos Expansion

Polynomial chaos expansion (PCE) is an efficient surrogate model that provides approximations to the QoIs using a spectral representation on a basis of polynomial functions with respect to the input parameter distributions. Consider a model represented by $\mathbf{y} = M(\mathbf{x})$. Here, $\mathbf{x}$ is the vector of input parameters, and $\mathbf{y}$ is the vector

of QoIs. For a vector-valued y, one PCE is computed for each QoI. For clarity, a scalar y is considered from now on. The input parameters $\mathbf{x}$ are independent variables with a prescribed PDF $f(\mathbf{x})$. The QoI, y, can be expanded into an orthonormal polynomial basis,

$$y = \sum a_\alpha \psi_\alpha(\mathbf{x}).$$

Here, a_α are unknown coefficients, and ψ_α are multivariate polynomials. For input parameters of each distribution type, there is a family of orthonormal polynomials, $\{\varphi_j^{(i)}, j \in N\}$, in terms of which the multivariate polynomial ψ_α is defined as

$$\psi_\alpha = \prod_{i=1}^{m} \varphi_\alpha^{(i)}(x_i),$$

where α is a multidimensional index, and m is the dimension of input parameters.

The classical family of univariate orthonormal polynomials and the distributions to which they are orthonormal are given in Table 2-1.

TABLE 2-1: LIST OF CLASSICAL POLYNOMIAL FAMILIES FOR POLYNOMIAL CHAOS EXPANSION.

INPUT PARAMETER DISTRIBUTION	EXPRESSION	POLYNOMIAL FAMILY
Uniform	$x/2$, $x \in [-1, 1]$	Legendre
Normal	$\exp(-x^2/2)/(2\pi)$, $x \in [-\infty, \infty]$	Hermite
Gamma	$\exp(-x)x^\alpha/\Gamma(\alpha + 1)$, $x \in [0, \infty]$	Laguerre
Beta	$(1-x)^\alpha(1+x)^\beta/B(\alpha + 1, \beta + 1)$, $x \in [-1, 1]$	Jacobi

In case the input parameters do not have a distribution given in the table, the algorithm defines an isoprobabilistic transformation to a uniform distribution and builds the PCE with the transformed input parameter. Further details about the orthonormal polynomial bases can be found in Ref. 5. Given the definition of the PCE, a truncation is defined such that the total degree of all polynomials is less than or equal to p. In addition, a hyperbolic truncation scheme that uses a parametric q-norm, $q \in [0, 1]$, is used to further truncate the polynomial expansions. For $q = 1$, the truncation method truncates all polynomials with total degrees smaller than p. For $q < 1$, hyperbolic truncation includes all the high-degree terms in each input parameter, but it removes equivalently high-order interaction terms. For $q = 0$, the truncation method truncates all interaction terms between input parameters. A sparse representation of PCE refers to a sparsely selected polynomial basis from all the truncated polynomial expansions. An adaptive algorithm based on a least-angle regression method is used to select the

significant sparse coefficients from all the truncated polynomial expansions. During the iterative process, the algorithm only adds new polynomial bases that are most correlated with the residual built with the existing polynomial basis. The sparse representation is both used for reducing the computational cost and, more importantly, for avoiding overfitting. Here, the leave-one-out cross-validation error estimation is defined with a correction factor that is used to consider overfitting and ensure that the generalization error estimate is not underestimated. The generalization error refers to the error for the sample points with unknown model evaluations. Additional details on the sparse PCE representation and the error estimation can be found in Ref. 6.

In COMSOL Multiphysics, both SPCE and ASPCE are trained adaptively based on the maximum total degree p. For ASPCE, the q-norm is also learned adaptively. Meanwhile, depending on whether the *a posteriori* cross-validation error estimation satisfies the tolerance, the ASPCE adds more input points selected by sequential optimal LHS to compute more model evaluations and train a new PCE in every adaptive step. For both SPCE and ASPCE, the truncated sparse coefficients are computed with an ordinary-least-square method in each adaptive step. You can use a specified q-norm for sparse PCE. If you have *a priori* information that the input parameter interaction is limited, a lower q-norm can be used to reduce the risk of overfitting.

Surrogate Models — Gaussian Process

The Gaussian process (GP) model, also known as the Kriging model, is a popular surrogate model that gives an approximate prediction of the QoIs, is probabilistic, and computes the variance of the prediction at every sample point in the input parameter space. A GP is specified by its mean, also known as the trend, and a covariance function, also called a kernel or correlation function. The hyperparameters of the mean and covariance functions are optimized while training the GP model. Consider input parameters $\mathbf{x}$ and a scalar QoI y. The GP model can then be expressed as

$$y = m(\mathbf{x}) + f(\mathbf{x})$$

where $m(\mathbf{x})$ is the mean function and $f(\mathbf{x})$ is a GP with zero mean and covariance $k_\theta(\mathbf{x}, \mathbf{x}')$. Here, θ represents the hyperparameters of the covariance function.

It is then possible to write the joint distribution of the evaluated data and the predicted data as

$$\begin{bmatrix} f \\ f' \end{bmatrix} = N\left(0, \begin{bmatrix} k_\theta(\mathbf{x}, \mathbf{x}) & k_\theta(\mathbf{x}, \mathbf{x}') \\ k_\theta(\mathbf{x}', \mathbf{x}) & k_\theta(\mathbf{x}', \mathbf{x}') \end{bmatrix}\right)$$

where $f = y - m(\mathbf{x})$, y is the evaluated QoI, and f' and $\mathbf{x}'$ are the predicted QoI and its corresponding input parameters.

The posterior distribution of the GP is written as

$$y(\mathbf{x}') \sim N(\mu(\mathbf{x}'), \mathrm{cov}(\mathbf{x}, \mathbf{x}'))$$

where $\mu(\mathbf{x})$ and $\mathrm{cov}(\mathbf{x}, \mathbf{x}')$ are the mean and covariance of the model, respectively:

$$\mu(\mathbf{x}') = m(\mathbf{x}') + k_\theta(\mathbf{x}', \mathbf{x}) k_\theta(\mathbf{x}, \mathbf{x})^{-1} f(\mathbf{x})$$
$$\mathrm{cov}(\mathbf{x}', \mathbf{x}) = k_\theta(\mathbf{x}', \mathbf{x}') - k_\theta(\mathbf{x}', \mathbf{x}) k_\theta(\mathbf{x}, \mathbf{x})^{-1} k_\theta(\mathbf{x}, \mathbf{x}')$$

Covariance functions are a crucial ingredient of GPs that determine the shape of the prior and posterior of the model. They encode the assumption that the function is being learned by defining the similarity of two input parameter points (stationary kernel) or specific values of the input parameter points (nonstationary kernel). The module provides three common stationary kernels and one nonstationary kernel. The stationary kernels are the spectral exponential kernel

$$k_{\mathrm{SE}}(\mathbf{x}, \mathbf{x}') = \exp\left(-\sum_{d=1}^{m} \frac{(x_d - x'_d)^2}{2l_d^2}\right)$$

and the Matérn class kernel with $\nu = 3/2$ and $\nu = 5/2$

$$k_{\mathrm{Matérn}_{\frac{3}{2}}}(r) = (1 + \sqrt{3}r)\exp(-\sqrt{3}r)$$
$$k_{\mathrm{Matérn}_{\frac{5}{2}}}(r) = \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right)\exp(-\sqrt{5}r)$$

where r is a function of $\mathbf{x}$ and $\mathbf{x}'$ defined as

$$r(\mathbf{x}, \mathbf{x}') = \sqrt{\sum_{d=1}^{D} \frac{(x_d - x'_d)^2}{l_d^2}}$$

The nonstationary kernel is the single-layer neural network kernel

$$k_{NN}(\bar{\mathbf{x}}, \bar{\mathbf{x}}') = \frac{2}{\pi} \operatorname{asin}\left(\frac{2\bar{\mathbf{x}}^T \Sigma \bar{\mathbf{x}}'}{\sqrt{(2\bar{\mathbf{x}}^T \Sigma \bar{\mathbf{x}})(2\bar{\mathbf{x}}'^T \Sigma \bar{\mathbf{x}}')}} \right)$$

Here, $\bar{\mathbf{x}}$ is defined as $\bar{\mathbf{x}} = [1, \mathbf{x}]$ and $\Sigma = \operatorname{diag}(\sigma_0^2, \sigma^2)$, where σ_0 and $\sigma = (\sigma_1, ..., \sigma_m)$ are the hyperparameters. Further details about the covariance function and the Gaussian process can be found in Ref. 7.

The covariance functions are assumed to be anisotropic. The *automatic relevance determination* (ARD) method is used to define different length scales for each input parameter. The ARD method is given its name because estimating the length scale parameters implicitly determines the "relevance" of each dimension. Input dimensions with relatively large length scales imply relatively little variation along those dimensions in the QoI function being modeled.

The spectral exponential kernel has strong smoothness where the Matérn kernel with $\nu = 5/2$ is less smooth, and the Matérn kernel with $\nu = 3/2$ is more rough. The GP model with a spectral exponential kernel is infinitely differentiable, whereas the GP model with a Matérn kernel is differentiable a finite number of times. Comparing the Matérn kernel and a spectral exponential kernel with the same length scale, the Matérn kernel has weaker correlation and is more oscillatory. Samples from a GP with the one-layer neural network kernel can be viewed as superpositions of functions with a single-layer neural network with $\operatorname{erf}(\mathbf{x}^T \Sigma \mathbf{x}')$ as the activation function. The GP model created with a single-layer neural network is smooth. In the region with a large absolute value of $\mathbf{x}$, they approximate to a constant value.

Regardless of which covariance matrix is used, it is often the case that it needs to be inverted while training the GP model. The inversion is known to suffer from numerical instabilities. To circumvent such limitations, a nugget factor defined as $\sigma^2 I$ is added to the covariance matrix. Here, σ is learned as an additional hyperparameter during the training of the GP.

There are three types of mean functions for GPs:

- Constant:

$$m_{\text{const}} = b$$

- Linear:

$$m_{\text{linear}}(\mathbf{x}) = b + \sum_{i=1}^{m} h_i x_i$$

- Quadratic:

$$m_{\text{quadratic}}(\mathbf{x}) = b + \sum_{i=1}^{m} h_i x_i + \sum_{i=1}^{m} \sum_{j=1}^{m} h_{ij} x_i x_j$$

GP models with constant and linear mean functions are also known as ordinary Kriging and universal Kriging, respectively. The coefficients for the mean functions are also learned during the training of the GP.

If there is *a priori* information of the mean function, it can be used to increase the modeling accuracy. One example of using a mean function is that when a GP is used for sensitivity analysis and uncertainty propagation, the extrapolation of the GP far from the sample points might be computed. If there is *a priori* information of the mean, using a linear or quadratic mean function can be beneficial for the extrapolation.

While training the GP, the hyperparameters of covariance functions, mean functions, as well as the nugget factor are learned by maximizing the log-likelihood of the posterior of the GP:

$$\log p(f|\mathbf{x}) = -\frac{1}{2} f k_\theta(\mathbf{x}, \mathbf{x}) f - \frac{1}{2} \log \left| k_\theta(\mathbf{x}, \mathbf{x}) + \sigma^2 I \right| - \frac{n}{2} \log 2\pi$$

Since the log-likelihood may have multiple local optima, the local optimizer for training the hyperparameters needs to be started repeatedly by specifying different starting points in the hyperparameter space. This surrogate model provides an automatic number of restart points for training the hyperparameters. These points are chosen such that they are in the regions where the hyperparameters render relatively small negative log-likelihood. Also, the restarting points are space filling in the hyperparameter space. These points are obtained by first generating a large amount of possible choices and then using a K-cluster mean method to select different starting points following the method described in Ref. 8 from the Journal of Statistical Software.

For adaptive GPs, the GP is constructed adaptively by adding a new input parameter point at the location of the maximum entropy, when it is used for sensitivity analysis or uncertainty propagation, or at the location of the maximum expected feasibility function, when it is used for reliability analysis. For sensitivity analysis or uncertainty propagation, the GP is trained to approximate the underlying model accurately in the entire input parameter space. It follows the heuristics in Ref. 9, which adds a new point at the location with the highest entropy. Here, the entropy is defined as the standard deviation of the GP model. For reliability analysis, the GP is trained to accurately

approximate the underlying model only near the regions that satisfy the reliability conditions, where the maximum expected feasibility function is used to find the adaptation points. More details about the expected feasibility function can be found in Reliability Analysis — Efficient Global Reliability Analysis.

To explore the region in the input parameter space and exploit the most-recently-built GP, a global optimization problem is solved in every adaptation step in finding the location corresponding to the maximum entropy or maximum expected feasibility function. There are two types of global optimization methods, the dividing rectangles (DIRECT) method (see Ref. 10) and the Monte Carlo method. The default choice is the DIRECT method. For nonadaptive GPs, this global optimization procedure is used to determine the maximum entropy as the error estimation of the model. Note that nonadaptive GP models are not used for reliability analysis.

Note that adaptive GP models require enough initial model evaluations for the adaptation method to locate "good" adaptation points that can improve the accuracy of the GP model. Given an initial dataset with too few sample points, exploration of the unsearched region may not be properly done. One way to fix this problem is using the improve and analyze functionality to append more samples using sequential optimal LHS, where the optimal LHS samples randomly fill the entire input parameter space.

Also note that for the adaptive GP built on input parameters with long tails (near-zero lower CDF and near-one upper CDF), most initial sample points will be located at the region where input parameters have high probability. The accuracy of the GP model in the region where input parameters have low probability will be low due to the lack of sample points, thus most adaptation points will probably be placed in the low probability region. A better way to create a GP model with high accuracy in the entire input-parameter space in this circumstance would be using a uniform distribution from the minimum and maximum value corresponding to the lower and upper CDF.

Screening — Morris One-at-a-Time Method

The Screening, MOAT study uses the Morris one-step-at-a-time (MOAT) method, which means that in each run, only one input parameter is given a new value. This method is purely sample based and does not rely on a surrogate model. The Screening, MOAT study method first samples r trajectories with the Morris sampling method, where r is the repetition number. In the Morris sampling method, the input parameter space is partitioned into n levels for each parameter ϕ_i . Then, it picks r trajectories, where all the points on a trajectory are located from the n^m positions, where the

sampled data points are $\phi_{i,j}, j = 1, ..., r$, and m is the dimension of input parameters. More details about the Morris sampling method are described in Data Sampling — Morris Sampling. The input parameter values are mapped so as to lie in the range [0, 1]. Each value corresponds to an actual input parameter value $p_{i,j} = p_{i,j}(\phi_{i,j})$. From the evaluated QoI data, the elementary effect for the ith input can be computed as

$$d_{i,j} = \frac{f(\phi_{1,j}, \phi_{2,j}, ..., \phi_{i,j} \pm \Delta, ..., \phi_{m,j}) - f(\phi_{1,j}, \phi_{2,j}, ..., \phi_{m,j})}{\Delta},$$

where

$$\Delta = \frac{n}{2(n-1)} = \frac{1}{2} + \frac{1}{2(n-1)}.$$

The sign in the above formula is chosen such that the perturbed point is inside the hypercube. From the elementary effect for all the r replication points, the MOAT mean for the ith input becomes

$$\bar{\mu}_i = \frac{1}{r} \sum_{j=1}^{r} |d_{i,j}|$$

and the MOAT standard deviation for the ith input

$$\sigma_i = \sqrt{\frac{1}{r} \sum_{j=1}^{r} (d_{i,j} - \mu_i)^2},$$

where

$$\mu_i = \frac{1}{r} \sum_{j=1}^{r} d_{i,j}.$$

The MOAT mean estimates the overall effect of an input parameter on the QoI, and the MOAT standard deviation measures the nonlinear effects of the input parameter and the interaction effect of this input parameter and others.

Note that the choice of n is strictly linked to the choice of r. If you consider a high value of n, the possible levels to be explored in the input parameter space are increased. If r is not also changed to a higher value, the effort of increasing the resolution in the input parameter space is wasted.

The MOAT method is economic in the sense that it requires a relatively small number of model evaluations. As a drawback, the method relies on the elementary effect, which uses finite differences. However, the final measures are obtained by averaging several

elementary effects computed at different points of the input space, which lose the dependence on the specific points at which the elementary effects are computed. Therefore, this method is considered a global screening method. Note that MOAT provides the qualitative measure because the method provides the ranking in the order of importance of input parameters.

Sensitivity Analysis — Correlations

The correlation method is a widely used method to determine the linear relationship between each input parameter and QoI. The method is purely sample based and does not require a surrogate model. There are four type of correlations: bivariate correlation (also known as Pearson's correlation), rank bivariate correlation (also known as Spearman's rank correlation), partial correlation, and rank partial correlation.

The *bivariate correlation* is defined as

$$r_i = \frac{\mathrm{cov}(x_i, y)}{\sqrt{V(x_i)V(y)}}$$

Here, $\mathrm{cov}(x_i, y)$ is the covariance between the scalar QoI and the ith input parameter, and $V(x_i)$ and $V(y)$ are the variances of x_i and y, respectively. Note that for a UQ study with more than one QoI, the correlations between QoIs and input parameters are computed for each QoI. The correlation result, ranging from −1 to 1, measures the linear relationship between x_i and y, where a correlation equals 1 when y linearly grows with x_i. Correlation is equal to −1 when y linearly decreases with x_i.

The *rank correlation* is defined as the bivariate correlation between the rank values of x_i and y. The rank correlation computes the monotonic relationship between x_i and y. A rank correlation equals 1 when the QoI monotonically grows with x_i and it equals −1 when the QoI monotonically decreases with x_i.

Note that the bivariate correlation does not take into account the possible effect that other input parameters might have on the QoI. For instance, healthcare funding and disease rate could be positively correlated according to their bivariate correlation. Such an observation is contradictory because healthcare visits increasing with healthcare funding is ignored. On the other hand, *partial correlation* can be calculated to determine the linear relationship between x_i and y where all the linear effects from the other input parameters are removed:

$$r_{\text{partial}, i} = \frac{\text{cov}(x_i, y|(\mathbf{x}_{\backslash i}))}{\sqrt{V(x_i|(\mathbf{x}_{\backslash i}))V(y|(\mathbf{x}_{\backslash i}))}}$$

Here, $\mathbf{x}_{\backslash i}$ denotes all parameters but x_i. The *rank partial correlation* is defined as the bivariate correlation between the rank values of x_i and y. In terms of linear regression, the partial correlation can be interpreted as the correlation between the residuals resulting from the linear regression of x_i with $\mathbf{x}_{\backslash i}$ and y with $\mathbf{x}_{\backslash i}$.

Note that, compared to the screening method, sensitivity analysis provides a quantitative measure of the relationship between the input parameters and the QoI.

Sensitivity Analysis — Sobol Index

Sensitivity analysis is the study of how uncertainty in the QoI can be apportioned to different sources of uncertainty in the input parameters. Unlike correlations where the effects of parameters are estimated based on the model evaluation data, the Sobol method looks at the entire input parameter distribution and decomposes the variance of the QoI into a sum of contributions from the input parameters and their interactions. The number of terms in the Sobol indices of the variance of the QoI with m input parameters grows as 2^m. It is customary to compute only the m first-order effects (first-order Sobol indices) and the m total effects (total Sobol indices).

Given a scalar QoI defined as $y = M(x_1, x_2, ..., x_m)$, the variance-based first-order Sobol index is defined as

$$S_i = \frac{V_{x_i}(E_{\mathbf{x}_{\backslash i}}(y|x_i))}{V(y)}$$

where x_i is the ith input parameter and $\mathbf{x}_{\backslash i}$ denotes all the other parameters but x_i. The meaning of the inner expectation value is that the mean of y is taken over all possible values of $\mathbf{x}_{\backslash i}$ while fixing x_i. The outer variance is then computed over all possible values of x_i.

The total Sobol index is defined as

$$S_{T, i} = 1 - \frac{V_{\mathbf{x}_{\backslash i}}(E_{x_i}(y|(\mathbf{x}_{\backslash i})))}{V(y)}$$

which measures the summation of the first-order effect and interactions with other parameters of the ith input parameter. The second term in the equation can be seen as the first-order effect of all parameters but the ith parameter. So, one minus the second

term must give the contribution of all terms related to x_i. Note that for a UQ study with multiple QoIs, the Sobol indices are computed for each QoI.

There are two types of methods to compute the Sobol indices. One is through postprocessing of a polynomial chaos expansion (PCE) model. Given the definition of the PCE, the contributions from each input parameter and their interactions are readily separable. Thus the Sobol indices can be computed solely based on the coefficients trained from the PCE. More details on the computation of Sobol indices based on the PCE model can be found in Ref. 6.

The other method of computing the Sobol indices is through Monte Carlo analysis. As it is computationally expensive to perform Monte Carlo simulation directly with model evaluation, it provides a Gaussian process as the surrogate model for this type of analysis. The algorithm follows the best practice of simultaneous computation of the first and total Sobol index described in Ref. 11.

Uncertainty Propagation — Monte Carlo Analysis

Studying the uncertainty propagation of a model is equivalent to approximating the PDF of the QoI. To construct the PDF of the QoI accurately, a large number of samples is required. Such an operation is very expensive and the analysis therefore uses the surrogate model. Both PCE and the GP model are used for uncertainty propagation. To approximate the PDF, the KDE method is used, in which density is formulated in terms of known kernel functions, which have the form

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K_h\left(\frac{x - x_i}{h}\right)$$

where K_h is a specified symmetric PDF. Here, the standard normal distribution is used, and h is a smoothing parameter termed the bandwidth.

By default, the Uncertainty Quantification Module uses Silverman's rule for the smoothing parameter h. More details related to the KDE can be found in Ref. 12. In addition to the approximation of the probabilistic distribution, the confidence interval table of each QoI is computed. In this table, it gives the mean, maximum, minimum, and standard deviation of each QoI. It also computes the value of the QoI corresponding to different CDF levels. Such information can also serve as a lookup table to a rough estimate of finding the probability that a QoI is larger than a certain threshold.

The goal of reliability analysis is to determine the probability that a model will satisfy certain criteria given by the user-defined threshold. Thus, the probability, p_f, is defined by

$$p_f = \int\limits_{\text{COND}(y,\, z)} \cdots \int f(x)dx$$

where $f(\mathbf{x})$ is the probability density function for the input parameters $\mathbf{x}$, and the integration is performed over the region where the condition $\text{COND}(\mathbf{y}, \mathbf{z})$ is satisfied. Here, $\mathbf{y}$ is the vector of the QoIs, and $\mathbf{z}$ is the value of the threshold corresponding to each QoI.

For the jth QoI, the probability condition $\text{cond}(y_j, z_j)$ is true if $y_j > z_j$ or $y_j < z_j$.

For reliability analyses with multiple QoIs, the probability condition can be formed as "all true", where the condition is defined as

$$\text{COND}(y, z) = \bigcap_{j=1}^{n} \text{cond}(y_j, z_j)$$

or "any true" where the condition is defined as

$$\text{COND}(y, z) = \bigcup_{j=1}^{n} \text{cond}(y_j, z_j)$$

where n is the number of QoIs. In general, this integration is impossible to obtain with analytical methods.

Therefore, the integration is approximated, for example, with Monte Carlo analysis. Note that when comparing existing local reliability analysis methods, numerical integration is generally more accurate but it requires a large number of function evaluations. Similar to surrogate-based sensitivity analysis and uncertainty propagation, the reliability analysis is built with the use of a surrogate model. It uses the efficient global reliability analysis (EGRA), which is built on an adaptive Gaussian process (GP). The method balances the exploration of the region where the GP provides a good prediction and exploitation of the region where the GP has large variance and more model evaluations are required. Furthermore, it does not require the GP model to have high accuracy in the region far from the limit state where the

QoIs are equal to the thresholds. This is achieved by focusing on adding adaptive model evaluations around the limit state that reduces the number of model evaluations.

The expected feasibility function (EFF), which is used as the adaptive error estimation in EGRA, is used to select the location of the next input parameter point. The EFF defines the expectation of the sample lying within the vicinity, defined by $\pm\varepsilon(\mathbf{x})$, around the limit state. The feasibility function at $\mathbf{x}$ is defined as being positive within the vicinity around the limit state and 0 otherwise. For problems with a scalar value QoI y, the EFF is defined as the expectation of being within the vicinity around the limit state given by

$$\text{EFF}(\hat{y}(x)) = \int_{z-\varepsilon}^{(z+\varepsilon)} (\varepsilon - |z - \hat{y}|)f_{\hat{y}}d\hat{y}$$

where $f_{\hat{y}}$ is the probability of the AGP and y is realizations of $f_{\hat{y}}$. Note that ε defines the vicinity around z and is set to $2\sigma_y$, where σ_y is the standard deviation of the AGP model.

Note that EFF provides a balance between exploration and exploitation: Points where the expected value is close to the threshold and points with a large uncertainty in the prediction will have a larger expected feasibility value. The adaptation point is defined as

$$x_{\text{adp}} = \text{argmax}(\text{EFF}(\hat{y}(x)))$$

The most efficient way of finding the solution to system-level EGRA problems is to construct a GP for each individual quantity of interest and then construct a composite EFF. Additional details about the EFF can be found in Ref. 13.

There are two sampling methods: One is to use importance sampling and the other is simply using Latin hypercube sampling. For the Lain hypercube sampling method, some number of samples of the input parameters are first randomly generated based on their distributions. Then, the surrogate models are evaluated at their input points. The values of the surrogate model at these points are compared with the threshold to determine if the predicted QoIs satisfy the probability conditions. The probability p_f is then calculated as the ratio of the number of predicted QoIs that satisfy the criteria for the total number of samples:

$$p_f = \frac{N_f}{N}$$

Here, N_f is the number of predicted QoIs that satisfy the criteria, and N is the number of all samples. One drawback of this method is that the majority of the samples lie in the high-probability region of the input parameter space. If the region defined by the threshold is a low-probability region, a very large number of samples is required to ensure enough samples to be located in such a region. Another method is the multimodal adaptive importance sampling method, which combines the surrogate model and Latin hypercube sampling. Compared to the Latin hypercube method, the importance sampling method uses fewer samples and provides an error estimation.

The procedure of performing EGRA with importance sampling is as follows. First, an initial set of model evaluations is computed and the initial GP is constructed based on these data. Then the point with maximum expected feasibility is searched as a global optimization problem using the GP model. Next, the adaptation model evaluation is evaluated at the point, and the enhanced dataset is used to construct an updated GP. The adaptation procedure is repeated until the maximum expected feasibility is smaller than the relative tolerance or the maximum number of model evaluations is reached. The final GP provides information on the data points located near the limit state, which serves as a good initial state for the importance sampling method.

To ensure that the initial GP covers a sufficiently large region in the input parameter space, instead of sampling according to the input parameter distribution, the reliability analysis uniformly samples in the region between the minimum and maximum values of each input dimension where the minimum and maximum is user specified or given by the input parameter distribution and the lower and upper cumulative distribution function. Note that when the initial GP has too little information (too few model evaluations) or the global optimization does not cover a sufficiently large space, the EFF may not locate the limit state. Increasing the initial number of model evaluations and the maximum surrogate evaluations for optimizations for the global optimization procedure of finding maximum EFF could improve the accuracy of the analysis.

Inverse Uncertainty Quantification— Markov Chain Monte Carlo

The goal of inverse uncertainty quantification analysis is to determine the probability distribution of unknown model parameters. These parameters can be unknown for example because they cannot directly be measured. These parameters are known as the calibration parameters and they are estimated based on experimental data that relates to the parameters through a computational model. Inverse uncertainty quantification

propagates the experimental data information backwards to gain knowledge of the calibration parameters. Compare this to Uncertainty Propagation that propagates a known parameters distribution forward to gain knowledge of the QoIs.

Consider a computational model represented by $\mathbf{y} = M(\mathbf{x}, \theta)$. Here, $\mathbf{x}$ is the vector of experimental parameters, θ is the vector of calibration parameters, and $\mathbf{y}$ is the vector of QoIs. The experimental parameters $\mathbf{x}$ are independent variables with a known prescribed PDF $p(\mathbf{x})$. The calibration parameters θ are parameters to be calibrated from the inverse uncertainty quantification study, the existing knowledge on the calibration parameters are used to prescribe their prior distribution $p(\theta)$. Note that the calibration parameters are considered as inputs to the computer model but are unknown when conducting the experiments. Experiment parameters are known for both the computational model and physical experiment, and they are used to describe the conditions under which the experiments have been conducted.

To connect the computational model and the experimental observation $\mathbf{y}_E$, a discrepancy model is introduced as,

$$\mathbf{y}_E \ = \ M(\mathbf{x}, \theta) + \delta$$

where $\delta \sim N(0, \varepsilon)$ represents an additive error and is assumed to be normally distributed with mean equals to $\mathbf{0}$ and an unknown diagonal covariance matrix $\varepsilon = \sigma^2 \mathbf{I}$, where σ is inferred with the calibration parameters. The definition assumes that the instrumentation, or the computer model, has no systematic bias.

Bayesian inference theory is used to determined the posterior distribution of the calibration parameters which is defined as $p(\theta | \mathbf{y}, \mathbf{y}_E)$, where $\mathbf{y}$ and $\mathbf{y}_E$ represents the computational model and experimental data respectively. According to the Bayesian theory, we have

$$p(\theta | \mathbf{y}, \mathbf{y}_E) \sim p(\mathbf{y}, \mathbf{y}_E | \theta) p(\theta)$$

where $p(\theta)$ is the prior distribution of the calibration parameter, and $p(\mathbf{y}, \mathbf{y}_E | \theta)$ is the likelihood function. Here the prior and posterior distribution reflects the degree of belief on possible values of θ before and after considering the experimental data. The likelihood function based on all the experimental data is defined as

$$p(y, y_E | \theta) = \prod_{i=1}^{N} \frac{1}{\sqrt{(2\pi\sigma^2)^m}} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y}_{E,i} - \mathbf{y}_i)^T (\mathbf{y}_{E,i} - \mathbf{y}_i)\right)$$

where N is the number of experimental data, m is the number of QoIs. Here, the experimental data is considered as independent samples conditioned on θ.

The computational model is a surrogate model constructed with the known distribution of the experimental parameters and the prior distribution of the calibration parameters. In practice, the posterior distribution defined by Bayesian theory do not have a closed-form solution. We rely on MCMC method to solve the inverse problem. The basic idea of MCMC is to build a Markov chain with an invariant distribution that equals to the posterior distribution. MCMC method produces chains of sample points that follows the posterior distribution. In practice, users need to make decision about the termination of the Markov chain based on the number of samples and the step size between two consecutive samples. To diminish the influence of the starting values, we discard the first half of each chain and only keep the second half of the chain. The discarded samples are considered as burn-in samples or warm-up samples. More details related to MCMC can be found in Ref. 14.

You can perform a uncertainty propagation study after the inverse uncertainty quantification by using the calibrated parameters to do the forward Monte Carlo analysis and study the calibrated QoIs. Note that inverse uncertainty quantification can also be used to calibrate the predicted QoIs to real world observations. By changing the status from input parameter with a known distribution, to a calibration parameters, we can use experimental observations to fine-tune these inputs and reduce the difference between the computational model and observations.

References

1. R. Jin, W. Chen, and A. Sudjianto. "An efficient algorithm for constructing optimal design of computer experiments," *J. Stat. Plan. Inference*, vol. 134, no. 1, pp 268–287, 2005.

2. X. Kong, M. Ai, and K.L. Tsui, "Design for Sequential Follow-Up Experiments in Computer Emulations," *Technometrics*, vol. 60, no. 1, pp. 61–69, 2018.

3. F., Campolongo, J. Cariboni, and A. Saltelli, "An effective screening design for sensitivity analysis of large models," *Environ. Model. Softw.*, vol. 22, no. 10, pp. 1509–1518, 2007.

4. T. Zou, S. Mahadevan, Z. Mourelatos, and P. Meernik, "Reliability analysis of automotive body-door subsystem," *Reliab. Eng. Syst.*, vol. 78, no. 3, pp. 315–324, 2002.

5. D. Xiu and G.E. Karniadakis, "The Wiener–Askey polynomial chaos for stochastic differential equations," *SIAM J. Sci. Comput.*, vol. 24, no. 2, pp. 619–644, 2002.

6. G. Blatman, and B. Sudret, "Adaptive sparse polynomial chaos expansion based on least angle regression," *J. Comput. Phys.*, vol. 230, no. 6, pp. 2345–2367, 2011.

7. C.K. Williams and C.E. Rasmussen, *Gaussian processes for machine learning*, MIT Press, Cambridge, MA, vol. 2, no. 3, p. 4, 2006.

8. B. MacDonald, P. Ranjan, and H. Chipman, "GPfit: An R package for fitting a Gaussian process model to deterministic simulator outputs," *J. Stat. Softw.*, vol. 64, pp .1–23, 2015.

9. M.D. McKay, R.J. Beckman, and W.J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.

10. D.E. Finkel and C.T. Kelley, "Additive scaling and the DIRECT algorithm," *J. Glob. Optim.*, vol. 36, no. 4, pp. 597–608, 2006.

11. A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola, "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index," *Comput. Phys. Commun.*, vol. 181, no. 2, pp. 259–270, 2010.

12. S.J. Sheather, "Density estimation," *Stat. Sci.*, vol. 19, no. 4, pp. 588–597, 2004.

13. B.J. Bichon, M.S. Eldred, L.P. Swiler, S. Mahadevan, and J.M. McFarland, "Efficient global reliability analysis for nonlinear implicit performance functions," *AIAA J.*, vol. 46, no. 10, pp. 2459–2468, 2008.

14. A. Gelman, J.B. Carlin, H.S. Stern and D.B. Rubin, Bayesian data analysis. Chapman and Hall/CRC. 1995.

3

Uncertainty Quantification Modeling

The Uncertainty Quantification Module is designed to facilitate setting up and solving uncertainty quantification (UQ) problems. You can specify a number of UQ study types for computations of uncertainties and sensitivities and specify input parameters and their statistical distributions as well as lower and upper bounds. This chapter describes the UQ study types and their applications. It also contains a description of the settings for the Uncertainty Quantification study.

In this chapter:

Overview

The Uncertainty Quantification study provides the tools and functionality for UQ studies of varying types. See the following sections for more information about each UQ study type.

About Input Parameters and Data Sampling

For most input parameters in an uncertainty quantification study, the parameter name, distribution type, specifications that define the distribution, lower and upper CDF or bounds of the distribution, as well as parameter units are needed to specify how the inputs are sampled. For screening, the Morris sampling method is used, and for other UQ study types, Latin hypercube sampling is used. Additional details about the input distribution and data sampling are provided in the theory section (see Data sampling methods).

In addition to generating input parameter data from analytic probability distributions, specified values and data column from result table can be used for sensitivity analysis, uncertainty propagation, and reliability analysis.

For verification and surrogate-based Monte Carlo analysis, the input parameters are usually the same as the input parameters used to generate quantities of interest through COMSOL model evaluations.

About Surrogate Model Settings

The sensitivity analysis, uncertainty propagation, reliability analysis and inverse uncertainty quantification rely on accurate surrogate models to deliver accurate UQ analysis. The relative tolerance is provided for all kinds of surrogate models. For a SPCE model, the relative tolerance is used to terminate the model training process. For an ASPCE model, the relative tolerance is used to terminate the model training process for each adaptive step combined with the maximum number of model evaluations, which is used to terminate the adaptive procedure. For a GP model, the relative tolerance is used to check against the error estimates. For an AGP model, the relative tolerance combined with the maximum number of model evaluations is used to terminate the adaptive process. For the SPCE model, you can also set the q-norm, which determines the truncation level of the polynomial basis. For the GP and AGP methods, different kinds of covariance and mean functions and different types of

optimization methods for finding the maximum error estimates are provided. The covariance and mean functions are used to determine the shape of the GP and AGP models. Further details about the surrogate model are given in the theory section (see Surrogate models).

To use a Surrogate model for a UQ study, choose a method from the **Surrogate model** list. The available methods are: **Adaptive sparse polynomial chaos expansion**, **Sparse polynomial chaos expansion**, **Adaptive Gaussian process**, and **Gaussian process**. Note that reliability analysis only works with **Adaptive Gaussian process**. A detailed explanation is given in the theory section (see Reliability Analysis — Efficient Global Reliability Analysis). For models of the polynomial chaos expansion type, both **Adaptive sparse polynomial chaos expansion** and **Sparse polynomial chaos expansion** models have a **Relative tolerance**, which is the leave-one-out cross-validation error that is used to find the best multivariate polynomial basis. If this error is not smaller than the **Relative tolerance**, a warning is given in the **Log** window. For **Adaptive sparse polynomial chaos expansion**, if the leave-one-out cross-validation error for the current adaptive iteration step is larger than the **Relative tolerance** and the total number of model evaluations is smaller than the **Maximum of input points**, then more input parameter points will be sampled with the LHS method, and more model evaluations will be computed at these points to create a new polynomial chaos expansion model. The adaptive version only has a setting for the **Relative tolerance**. Here, the polynomial degree and the q-norm are computed iteratively in each adaptation process. For GP-type models, both **Adaptive Gaussian process** and **Gaussian process** models have a **Relative tolerance**. For **Gaussian process**, the **Relative tolerance** is used to check if the error estimation is small enough. For **Adaptive Gaussian process**, the adaptation process finds the next sample point corresponding to the location of the largest error estimation. The searching procedure is a global optimization where **DIRECT** and **Monte Carlo** methods can be used. For the **DIRECT** method, the optimization is stopped when the **Maximum surrogate evaluations for optimization** and the **Maximum number of optimization iteration** are reached. For the **Monte Carlo** method, the optimization computes surrogate model evaluations on all the **Surrogate evaluations for optimization** sample points. For both **Adaptive Gaussian process** and **Gaussian process**, if this error is not smaller than the **Relative tolerance**, a warning is given in the **Log** window.

Adding an Uncertainty Quantification Study

To add an **Uncertainty Quantification** study node (🔲) to the current study, right-click the main Study node and choose **Uncertainty Quantification** from the **Uncertainty Quantification** submenu.

You can also add an **Uncertainty Quantification** study node to a new study with a study reference to the current study. To do so, right-click the main Study node and choose **Add Uncertainty Quantification Study Using Study Reference** from the **Uncertainty Quantification** submenu. The **Uncertainty Quantification** study node is then added under a new Study node together with a **Study Reference** node that refers back to the original study. This way, you can set up separate UQ studies with different study types and run them independently and, at the same time, use the original study.

If you have a Study node with a **Uncertainty Quantification** study node and you want to place the **Uncertainty Quantification** study node under a new Study node, together with a **Study Reference** node that refers back to the original study, you can right-click the **Uncertainty Quantification** study node and choose **Move Uncertainty Quantification to New Study**.

To add a new type of UQ study and keep most settings in the current **Uncertainty Quantification** study node, add a new study with a UQ and a study reference to the current study. To do so, right-click the **Uncertainty Quantification** study node and choose one of the UQ study types from the **Add New Uncertainty Quantification Study For** submenu. The **Uncertainty Quantification** study node with the selected UQ study type is then added under a new Study node together with a copy of all the other nodes under the original study. If the original UQ study is **Screening, MOAT**, you can choose sensitivity analysis, uncertainty propagation, or reliability analysis. If the original study has a **Quantities of interest** table or **Verification quantities of interest** table, the newly created UQ study will copy the data in these tables to new tables. When the new UQ study is computed, new quantities of interest data will be written to the new tables. If **Output table group usage** is **Automatic**, the new UQ analysis results will be saved in a newly created **Output table group**. The newly created UQ study uses **Improve and analyze** as the **Compute action** if the original study has a **Quantities of interest** table, such that the new UQ study could reuse the quantities of interest data from the original study. Note that, if you want to change the input parameter or quantities of interest from the original study, the quantities of interest data cannot be used for **Improve and analyze**; instead, use **Compute and analyze**. Similarly, the newly created UQ study uses **Verify only** as the **Verify action** if the original study has a **Verification quantities of interest** table, such that the new UQ study cannot reuse the verification quantities of interest data from the original study. Note that if you want to change the input parameter or quantities of interest from the original study and create a new surrogate model, the verification quantities of interest data cannot be used for **Verify only**; instead, use **Compute and verify**.

Using the Screening, MOAT Study Type

Due to a potentially high number of input parameters, a pure sample-based screening method with low computational demand, such as the Screening, MOAT method, is useful to identify the most influential input parameters.

Background

The Screening, MOAT study uses the Morris one-step-at-a-time (MOAT) method, which means that in each run, only one input parameter is given a new value. The Screening, MOAT study method first samples the m parameters with the Morris sampling method. For each replication point, the parameter value is perturbed in every parameter dimension, such that $r(m+1)$ total model evaluations are needed. From the evaluated data, this study type can compute the elementary effect for each input parameter and each quantity of interest. From the elementary effect for all the r replication points, you get the MOAT mean and the MOAT standard deviation. The higher the MOAT mean μ_i, the more sensitive the ith input parameter is. The higher the MOAT standard deviation σ_i, the more interaction the ith input parameter has with other parameters. More details about the MOAT method and Morris sampling method are described in the theory section under Data Sampling — Morris Sampling and Screening — Morris One-at-a-Time Method, respectively.

Settings

Select **Screening, MOAT** from the **UQ study type** list to perform a UQ study with a screening using the MOAT method. No additional specific settings are needed for this study type. However, the settings for the **Repetition number** and **Partition level** are used for Morris sampling and are discussed above in the Background section above Background. As a convenience, to estimate the total cost of the analysis, the **Number of input points** is computed based on the setting for the repetition number and the number of input parameters. For a full overview of the settings for the **Uncertainty Quantification** node, see The Uncertainty Quantification Study.

Results

The output for the screening study includes two tables: the quantities of interest table and the MOAT table. The quantities of interest table contains all the sampled input parameters from the Morris method and quantities of interest computed with the COMSOL model. The MOAT table lists the MOAT mean and MOAT standard deviation for the input parameters, discussed in the Background section. The MOAT table data is also shown in 1D plot groups, where table annotation plots show the MOAT table data, with one plot for each quantity of interest.

Using the Sensitivity Analysis Study Type

The sensitivity analysis study is useful for identifying which of the input parameters has the largest impact on the quantities of interests. This analysis provides insight into how sensitive the quantities of interest are with respect to variance in the input parameters. It can also help identify the important input parameters, which is useful in connection with uncertainty propagation and reliability analysis.

Background

The Sensitivity Analysis study provides global sensitivity analysis and supports both the Sobol indices method and the correlation method.

For the Sobol indices method, the most efficient method is to generate a polynomial chaos expansion (PCE) surrogate model for each quantity of interest (QoI). The Sobol indices can readily be computed from the coefficients of the PCE model without sampling through a Monte Carlo procedure. The Sobol indices can also be computed through Monte Carlo analysis when a Gaussian process (GP) surrogate model is used. For details on the surrogate models and how the Sobol indices are computed, see the following sections in the Theory chapter: Surrogate Models — Polynomial Chaos Expansion, Surrogate Models — Gaussian Process, and Sensitivity Analysis — Sobol Index.

For the correlation method, the sensitivity analysis is performed through pure sampling-based statistical analysis with no surrogate model. There are four types of correlations:

- The bivariate correlation, also known as the Pearson's correlation, which computes the linear relationship between each input parameter and QoI.

- The rank bivariate correlation, also known as Spearman's correlation, which computes the monotonicity between the each input parameter and QoI.

- The partial correlation, which computes the linear relationship between each input parameter and QoI where all linear effects from other input parameters are removed.

- The rank partial correlation, which computes the monotonic relationship between each input parameter and QoI where all linear effects from other input parameters

are removed. For details on how each correlation is computed, see Sensitivity Analysis — Correlations in the Theory chapter.

Settings

Select **Sensitivity analysis** from the **UQ study type** list to perform a UQ study with a sensitivity analysis.

For the sensitivity analysis method, choose **Sobol** (the default) or **Correlation** from the **Method** list.

For the Sobol method, the most efficient surrogate model is of the polynomial chaos expansion (PCE) type. But you can also use the type Gaussian process (GP). See the Theory sections Surrogate Models — Polynomial Chaos Expansion and Surrogate Models — Gaussian Process. If a GP model is used, you also need to choose a Monte Carlo analysis.

For a full overview of the settings for the **Uncertainty Quantification** node, see The Uncertainty Quantification Study.

Results

The output for the sensitivity analysis study includes the QoI table, which contains the sampled input parameters with the LHS method as well as the QoI computed with the COMSOL model.

For the Sobol method, there is also the Sobol table as well as the Sobol training error table. For the adaptive training method, there are also Adaptive Sobol and Adaptive Sobol training error tables. The adaptive tables have results (updated while solving) for the Sobol indices as well as the error estimates. These tables can be useful to inspect how the process is going and also verify that the errors are reduced during the adaptation process. This method also adds Sobol index plots, one per QoI, in 1D plot groups. These plots use the Sobol table as their data source. Both the first and total index are given with different colors for the different inputs. The plot is by default sorted so that the most influential inputs are shown first.

For the Correlation method, there are tables for bivariate and rank bivariate correlation, as well as partial and rank partial correlation.

Using the Uncertainty Propagation Study Type

In an uncertainty propagation study, probability distribution for the input parameters is propagated to probability distributions for the QoI. Another, less statistical, way of expressing this: In a uncertainty propagation study, the variation for the outputs is computed from given variations for the inputs. The distribution (or variation) for the inputs is well defined through the input parameter specification for the UQ study, but the output distribution is in general far from simple to compute or estimate, in particular for nonlinear QoIs. The method used for the Uncertainty Quantification Module is to first train a generally valid surrogate model, and then use Monte Carlo methods to get raw data for a KDE. The kernel density estimation itself is performed by a new dataset under Results and a new plot type, both automatically added by this analysis type.

Background

The Uncertainty Propagation study type performs a number of model evaluations from LHS sampling data; such data is used to generate a surrogate model. Then a number of predictions are made by using the surrogate model in a Monte Carlo analysis. A KDE is generated with this sampled data to produce the estimation of the probability density for each QoI. For details on the KDE and surrogate method, see the Theory sections Uncertainty Propagation — Monte Carlo Analysis, Surrogate Models — Polynomial Chaos Expansion, and Surrogate Models — Gaussian Process.

Settings

Select **Uncertainty propagation** from the **UQ study type** list to perform a UQ study with uncertainty propagation. Alternatively, select **Add New Uncertainty Quantification Study For>Uncertainty Propagation** from the context menu for another UQ study in the same model. The advantage with the latter method is that you will start from a copy of the QoI and input parameter settings from the UQ study. Another advantage is that the QoI table will also be copied so that you will by default start with the method **Improve and analyze**, which will include your previous data in the training. If you are adding (or removing) input parameters or QoIs, this method will not work, and you have to change **Compute action** to **Compute and analyze**.

For a full overview of the settings for the **Uncertainty Quantification** node, see The Uncertainty Quantification Study.

Results

By performing an uncertainty propagation (UP) analysis using the nonadaptive surrogate models, four tables are added to the output table group for the UP analysis.

- A **QoI Confidence interval** table. It contains one row per QoI, with columns for the mean, standard deviation (STD), minimum, and maximum, followed by confidence intervals for 90%, 95%, and 99% likelihood.

- A **UP predicted QoI** table. It contains the surrogate-model-predicted values for the QoIs for the Monte Carlo sampling points. This data table is used by KDE datasets that are also added by the analysis, one per QoI.

- A **UP predicted STD** table. It contains the surrogate-model-predicted standard deviation for the Monte Carlo sampling points. This can be seen as the built-in surrogate model error estimation.

- Finally, a **Maximum entropy** table. It contains the maximum relative standard deviation, one per QoI.

When an adaptive surrogate model is used, an **Adaptive maximum entropy** table is also added to the output table group. It contains the maximum relative standard deviation, one per QoI for all the adaption steps.

An Uncertainty Quantification plot group is also added with KDE plots, one for each QoI.

Using the Reliability Analysis Study Type

Reliability analysis provides an efficient and accurate way of studying the probability that the QoIs satisfy reliability conditions. It answers the following question: Given a set of uncertain input parameters and quantities of interests, what is the probability that the QoIs satisfy certain conditions compared to given thresholds?

Background

The Reliability Analysis study type is used to compute the probability that all QoIs satisfy the reliability conditions or any QoIs satisfy the reliability conditions using the EGRA method. The EGRA method is constructed upon an AGP surrogate model where the EFF is used as an estimation of where the AGP model should append the next adaptation point. The EFF is constructed to balance the exploitation of the AGP model and the exploration of the unobserved input parameter space. More details of EGRA can be found in Reliability Analysis — Efficient Global Reliability Analysis.

Settings

Select **Reliability analysis, EGRA** from the **UQ study type** list to perform a UQ study with reliability analysis. You can also specify settings for the surrogate model. Only the adaptive GP is an applicable surrogate model for this type of analysis.

For a full overview of the settings for the **Uncertainty Quantification** node, see The Uncertainty Quantification Study.

Results

Six tables are generated in an efficient global reliability analysis (RA) and added to the result output table group for RA:

- The **Probability for conditions** table contains the important number that is the result of the EGRA problem.
- The **Adaptive probability for conditions** table, with the probabilities for all the adaptive steps.

- The **Maximum expected feasibility function** table gives the value for the EFF for the final step of the adaptive process.

- The **Adaptive maximum expected feasibility function** table contains the EFF for all the adaptive steps.

- The **RA predicted QoI** table provides the predicted values for the QoIs for the sample points produced by the Monte Carlo process.

- The **Adaptive RA predicted QoI** table contains the predicted values of the QoIs for all the adaptive steps.

- Finally, a **Maximum entropy** table. It contains the maximum relative standard deviation, one per QoI.

When an adaptive surrogate model is used, an **Adaptive maximum entropy** table is also added to the output table group. It contains the maximum relative standard deviation, one per QoI for all the adaption steps.

An Uncertainty Quantification plot group is also added with KDE plots, one for each QoI.

Using the Inverse Uncertainty Quantification Study Type

Inverse uncertainty quantification estimates the calibration parameters using the experimental data and prior knowledge on the calibration parameters. It can be seen as parameter estimation with a Bayesian perspective.

Background

The Inverse Uncertainty Quantification study type is used to compute the posterior distribution of all calibration parameters based on the experimental data and the prior distribution of the calibration parameters. The MCMC method is used to construct Markov chains where the sampled data from a converged Markov chain is based on the posterior distribution. More details of Inverse Uncertainty Quantification can be found in Inverse Uncertainty Quantification— Markov Chain Monte Carlo.

Settings

Select **Inverse Uncertainty Quantification** from the **UQ study type** list to perform a UQ study with inverse uncertainty quantification. You can also specify settings for the surrogate model, MCMC and experimental data. The settings for the **Number of samples** and **Step size** are used for sampling data from posterior distribution of calibration parameters, more details about MCMC are discussed in the Theory section in Inverse Uncertainty Quantification— Markov Chain Monte Carlo. For a full overview of the settings for the **Uncertainty Quantification** node, see The Uncertainty Quantification Study.

Results

Four tables are generated in an inverse uncertainty quantification study and added to the result output table group for inverse uncertainty quantification:

- A **MCMC samples** table. It contains the samples of all calibration parameters from their posterior distributions, the samples of the standard deviation of the added discrepancy between the surrogate model and experimental data, the log-likelihood of the posterior and the acceptance rate of the Markov chain. From a converged Markov chain, the log-posterior should be smoothly varying and close to its

maximum value. The acceptance rate measures the percentage of random samples accepted by the Markov chain iteration. A small acceptance rate close to 0 indicates that the step size might be too large where the new samples are mostly rejected by the Markov chain; a large acceptance rate close to 1 indicates that the step size might be too small where the new samples does not cover enough space from parameter space defined by the prior distributions.

- A **Calibrated confidence interval** table. It contains one row per calibrated parameter, with columns for the mean, standard deviation (STD), potential scale reduction factor (Rhat) and number of efficient samples (Neff), minimum, and maximum, followed by confidence intervals for 90%, 95%, and 99% likelihood. The potential scale reduction factor measures the ratio of the variance of samples within a Markov chain to the variance of the samples across chains computed with different starting points. The potential scale reduction factor should decline to 1 when the sample size approaches infinity and the chain converges. The number of efficient samples measures the effectiveness of a Markov chain, it estimates the number of samples as if they are independent random draws from the posterior distribution. The number of efficient samples should be at least 20 for each calibration parameter to be considered a converged Markov chain.

- A **Posterior mean around experimental data** table. It contains the experimental data, the mean and the standard deviation of the predicted QoIs computed with the calibrated parameters from the Markov chain in the **MCMC samples** table. The experimental data of the QoIs at the experimental data points should be contained within the bound of the predicted QoIs mean and standard deviation.

- Finally, a **Maximum entropy** table. It contains the maximum relative standard deviation from the GP surrogate model, one per QoI.

When an adaptive GP surrogate model is used, an **Adaptive maximum entropy** table is also added to the output table group. It contains the maximum relative standard deviation, one per QoI for all the adaption steps.

The Uncertainty Quantification Study

The **Uncertainty Quantification** study (▦) contains tools for setting up UQ studies of different types for evaluating the uncertainty and sensitivity in a simulation model with respect to some input parameters with variations described by some statistical distribution. To add an **Uncertainty Quantification** study, right-click a Study node and choose **Uncertainty Quantification**. You can only have one **Uncertainty Quantification** node in each study.

At the top of the **Settings** window, you can use the **Compute** action button to run the study. The effect of this action is determined by the **Compute action** settings as described below.

The **Settings** window contains the following sections:

UNCERTAINTY QUANTIFICATION SETTINGS

At the top of this section, use the **Compute action** list to select one of the following: **Compute and analyze**; **Improve and analyze**; **Analyze only**; or **Analyze only, including verification data**. When there is no previous data to use, the only option will be **Compute and analyze**, but when the QoI table has been populated, also **Improve and analyze** and **Analyze only** can be selected. **Improve and analyze** appends data (to the **Quantities of Interest** table) and analyzes the total amount of data. **Analyze only** performs the UQ analysis based on the data as is, and does not perform any new COMSOL model evaluations. When verification has been performed, **Analyze only, including verification data** is also an option. This option considers the union of data from the QoI table and the verification table.

From the **UQ study type** list, choose one of the following main study types for the UQ:

- **Screening, MOAT** (the default). See Using the Screening, MOAT Study Type.
- **Sensitivity analysis**. See Using the Sensitivity Analysis Study Type. For this study type, also choose a method from the **Method** list: **Sobol** (the default) or **Correlation**.
- **Uncertainty propagation**. See Using the Uncertainty Propagation Study Type.
- **Reliability analysis, EGRA**. See Using the Reliability Analysis Study Type.
- **Inverse uncertainty quantification, MCMC**. See Using the Inverse Uncertainty Quantification Study Type.

From the **Output table group usage** list, choose **Automatic** (the default) or **Manual**. For **Manual**, you can choose any existing table group or **New** from the **Output table group** list to use an existing table group or create a new table group for the table data output from the UQ study. **Automatic** automatically generates an output table group for each UQ study type. When you switch between different UQ study types, the result table group automatically points to the output table group corresponding to the study type. The **Clear previous result tables** check box is selected by default. Clear it to keep existing result tables.

For all UQ study types except **Screening, MOAT**, there are also settings under **Surrogate model settings**:

From the **Surrogate model** list, choose one of these surrogate models:

The following surrogate models are available for the **Sensitivity analysis** and **Uncertainty propagation** UQ study types.

- **Sparse polynomial chaos expansion**. This method compares the leave-one-out cross-validation error for its polynomial to the **Relative tolerance**. The method starts from the lowest possible polynomial order and then increases it, while still fulfilling the **Q norm** (default: 0.5), until the error is smaller than the tolerance and the surrogate model construction is done. If the maximum polynomial degree is reached without the tolerance criteria being reached, the surrogate model construction terminates, but a warning is printed to the **Log** window. The setting for the maximum polynomial degree can be set on the **Job Configurations** level; see Uncertainty Quantification Job Configurations. The q-norm is real valued in the interval $[0, 1]$, which determines the mixed terms included in the expansion. Mixed terms means terms that involve more than one parameter. A q-norm of 0 means that no mixed terms are used for the polynomial, while 1 allows mixed terms of the same combined order as the nonmixed terms. See the Theory section Surrogate Models — Polynomial Chaos Expansion for more information.

- **Adaptive sparse polynomial chaos expansion** (the default for Sensitivity analysis). This method finds the q-norm automatically during the surrogate model construction. This mechanism favors a small q-norm. The leave-one-out cross-validation error cannot be used to find new sampling points for the adaptation process. Instead, new sampling points are added with the LHS method, and a new error estimate is computed. The adaptation process is terminated if the error is smaller than the **Relative tolerance**. See the theory section Surrogate Models — Polynomial Chaos Expansion for more information.

- **Gaussian process**. This surrogate model uses the standard deviation as its built-in error estimate. This error estimate can be computed for any GP evaluation point. An optimization approach is used to find the maximum error in the input parameter domain. Use the **Optimization method for error estimation** list: **Direct** (the default) or **Monte Carlo**. The **Direct** method is normally preferred because it is faster and gives a better estimate. In each iteration of this method, the number of GP evaluations can change. Initially, only a few evaluations are done, but the number can quickly grow. Therefore, you can set a bound, both on the number of iterations and the total number of GP evaluations. Enter values in the **Maximum number of surrogate evaluations for optimization** (default: 10,000) and **Maximum number of optimization iterations** fields (default: 500). The **Monte Carlo** method can be used for comparison with the **Direct** method. If the error obtained from the optimization approach is not smaller than the **Relative tolerance**, a warning is printed to the **Log** window. For this surrogate model, also specify a covariance and a mean. From the **Covariance** list, choose **Squared exponential**, **Matérn 3/2** (the default), **Matérn 5/2**, **Neural network**, or **Spectral mixture**. From the **Mean** list, choose **Constant** (the default), **Linear**, or **Quadratic**. See the theory section Surrogate Models — Gaussian Process for more information.

- **Adaptive Gaussian process**. This is the only applicable surrogate model for **Reliability analysis, EGRA**. This method also uses the GP standard deviation for error estimation. Here, the same optimization methods as for the GP can be used to find the maximum. If the error obtained from the optimization approach is smaller than the **Relative tolerance** (default: 0.001), the adaptation iterations are terminated. The error estimate is used to add new sampling points, one at a time. For **Reliability analysis, EGRA**, the error estimation can be seen as a weighted error where the region for the QoIs away from the threshold are down-weighted to be less important. This means that most new points will be added close to the threshold, to enhance the quality of the surrogate model for this analysis. See the Theory section Reliability Analysis — Efficient Global Reliability Analysis for more information on this weighting.

When a surrogate model has been computed, you will, at the bottom of this section, find a **Verify action** list: **Compute and verify**, **Improve and verify**, and **Verify only**. Next to this list, you find an action button that executes the action selected. The **Compute and verify** action independently adds new sampling points, considering the ones you have already used for the surrogate model buildup. For these, new COMSOL model evaluations are performed. This data is then used to compute error estimates for the surrogate model. These error estimates are done independently of any surrogate model built-in error estimates and should be seen as an independent quality test of the

surrogate model. Notice that no new UQ analysis data is produced by this action. It will therefore not affect these results. The **Improve and verify** action can be used when an initial verification computation has already been done. It can be used to further add verification sampling points and COMSOL model evaluations to extend the verification. The **Verify only** action uses an existing verification table and only performs the verification error estimation, without adding any new sampling points or performing any new COMSOL model evaluations. This action can be useful, for example, if there is some COMSOL table with computational results for the same parameters and the same QoIs from a previous computation. See the section Surrogate Model Verification for more information.

QUANTITIES OF INTEREST

From the **Quantities of interest table** list, choose **New** to create a new table, or choose any of the available tables. This table will be used to store not only sampling points but also the model evaluation of the QoIs for these points. This data serve as a building block for the surrogate models. This table is used when the compute action is **Analyze only**. It can also be used as a starting point for the compute action **Improve and analyze**. This table is used for all UQ analyses, and is not put in the Output table group.

The **Solution to use** list contains the solution in a COMSOL model to use as the QoI. Choose **Automatic** (the default), **Summation**, **Minimum**, **Maximum**, **Use first**, or **Use last**. The **Automatic** solution to use will be the last solution for time-dependent and parametric solutions, while for eigenvalue and eigenfrequency solutions, it will be the first solution. You can override this mechanism by selecting any of the other methods. For **Summation**, the QoI is defined as the summation of the **Expression** over all the solutions. For **Maximum** (or **Minimum**), the QoI is defined as the maximum (or minimum) of the expression taken over all the solutions. Notice also that evaluation operators like at() and with() can be used in the expression, making it possible to evaluate even more general quantities from dynamic solutions.

The **Outer solution to use** list is available when the UQ study is using a parametric sweep, function sweep, or material sweep in the same study. The choice refers to the solutions in the parametric sweep type study to use as the QoI. Choose **Summation** (the default), **Minimum**, or **Maximum**. For **Summation**, the QoI is defined as the summation of the QoIs over all the solutions computed in the parametric-sweep-type studies. For **Maximum** (or **Minimum**), the QoI is defined as the maximum (or minimum) of the QoIs taken over all the parametric sweep solutions.

For **Reliability analysis, EGRA** with multiple QoIs, choose **All true** (the default) or **Any true** from the **Probability for conditions** list. With **All true**, the condition for each QoI must be fulfilled. With **Any true**, the condition for at least one QoI must be fulfilled.

For the **Screening, MOAT**; **Sensitivity analysis**; and **Uncertainty propagation** UQ studies, the QoI table includes the following columns:

- **Expression** — The expression of the QoI, a scalar expression.
- **Description** — An optional description of the expression.
- **Individual solution to use** — From the list in this column, you can use a specific solution for that QoI. The default, **From "Solution to use"**, takes the solution from the **Solution to use** list above the table.

For **Reliability analysis, EGRA**, in addition to the **Expression**, **Description**, and **Individual solution to use** settings in the **True if** and **Threshold** columns are also required. The value for **True if** defines the relationship between the expression and the threshold. **Larger than threshold** means that the reliability probability is defined as the condition that the value of the expression is larger than the threshold. **Smaller than threshold** is the opposite condition.

INPUT PARAMETERS

In this section, you define the input parameters for the UQ study and their related settings.

For each input parameter, from the **Source type** list, choose **Analytic** (the default) or **Data** (for the **Sensitivity analysis, Correlation, Uncertainty propagation** and **Reliability analysis, EGRA** study methods only).

Data Generation from Analytic source type

When you have chosen **Analytic** for all input parameters, the following additional settings are available (except for the **Screening, MOAT** study):

From the **Number of input points type** list, choose **Automatic** (the default) or **Manual**. The **Manual** type gives you the possibility to specify the **Initial number of input points**, which is the number of points generated and simulated (with the COMSOL model) before the adaptation starts. The **Maximum number of input points** is the total number of input points used by the adaptation. Both of these numbers are important for the adaptation process. The initial number needs to be large enough to cover the sampling space and the complexity in the QoIs. Otherwise, there is a risk that the adaptation algorithm is not adding new input points correctly. An appropriate number depends not only on the dimension of this space, which is the same as the number of input

parameters, but also on how large this space is in relation to the variation of the QoIs. The maximum number of input points puts a limit to the whole process. If the adaptation process has not terminated fulfilling the tolerance criteria when this limit is reached, a warning is printed in the **Log** window. When adaptation is not used, there is only one setting, the **Number of input points**, which dictates the exact size of the training data. Choosing this number will naturally be a tradeoff between surrogate model quality and training cost. Notice that the number of COMSOL model evaluations will be the same as the number of input points. The **Automatic** method chooses these numbers based on the number of input parameters and the compute action used. The number used for these properties can be inspected in the user interface, but they cannot be edited. Table 3-1 lists the default and automatic values for the different compute actions.

TABLE 3-1: DEFAULT AND AUTOMATIC VALUES FOR THE INPUT POINTS PROPERTIES.

COMPUTE ACTION	NUMBER OF INPUT POINTS TYPE	MAXIMUM NUMBER OF INPUT POINTS (ADAPTATION)	INITIAL NUMBER OF INPUT POINTS (ADAPTATION)	NUMBER OF INPUT POINTS (NONADAPTATION)
Compute and analyze	Automatic	$20m$	$10m$	$20m$
Compute and analyze	Manual	20	10	20
Improve and analyze	Automatic	$10m$	$5m$	$10m$
Improve and analyze	Manual	10	5	10

For the **Screening, MOAT** study, the following additional settings are available:

- The number of input points appears as a convenience below the **Input parameter data generation method** list. This number is equal to the number of COMSOL model evaluations. The formula is $(m + 1)r$, where r is the repetition number and m the number of input parameters.

- In the **Repetition number** field, enter the desired number of repetitions. Each repetition uses different start points for the Morris sampling technique from which one-at-a-time is changed to produce new sampling points (default: 4). See the Theory section Data Sampling — Morris Sampling.

- Choose a value for the **Partition level**, which specifies how many grid lines are used in each parameter dimension: 4 (the default), 6, 8, 10, or 12. See Data Sampling — Morris Sampling in the Theory chapter.

For all UQ study types where at least one input parameter has chosen **Analytic** from the **Source type**, specify the following additional settings:

From the **Random seed type** list, choose **Automatic** (the default) to generate a random seed automatically (which is then displayed below), **Manual** to enter a seed in the **Initial random seed** field, or **Current computer time** to use that time as the random seed. A random seed is used not only the first time sampling data is generated but also for subsequent generation, for example, for the **Improve and analyze** compute action. To avoid the same sampling data being generated over and over again, another random seed is used for subsequent runs. The **Automatic** method adds 1 to the seed each time you run the study, so that subsequent runs never use the same seed.

Data Generation Using Data source type
When you have chosen **Data**, you can choose the data source.

From the **Data source** list, choose **Specified values** (the default) or **Result table**.

When you have chosen **Result table**, you can chose the input parameter data source table and select the data **Column** from the table where the **Column** list is automatically populated with the table's column headers.

When you have chosen **Specified values** for all the input parameters, you can choose the sweep type.

From the **Sweep type** list, choose **Specified combinations** (the default) or **All combinations**. These sweep types work exactly as if a parametric sweep study type was used for the input parameters.

The **Number of input points** is computed based on the number of data points from **Specified values** or number of data points in the selected table **Column**, where the number of **Specified values** and number of data points in table **Column** should be the same. When **All combinations** is chosen from the **Sweep type** list, the **Number of input points** are the multiplication of the number of data points of each **Specified values**.

The Input Parameters Table
Under **Input parameters**, specify the input parameters for the UQ study in the table below, which contains the following columns:

• In the **Parameter** column, choose any existing global parameter as an input parameter.

• In the **Source type** column, choose **Analytic** or **Data**.

• In the **Parameter description** column, the distribution type and bound are automatically populated when **Analytic** is chosen from the **Source type** list, the data

source type and the number of input data points are automatically populated when **Data** is chosen from the **Source type** list.

When **Analytic** is chosen for **Source type**, the analytic distribution settings are shown under the table:

- In the **Distribution** list, choose the distribution for the input parameter: **Uniform** (the default), **Normal(μ,σ)**, **LogNormal(μ,σ)**, **Gamma(k,θ)**, **Beta(α,β)**, **Weibull(λ,k)**, or **Gumbel(μ,β)**. All distributions except the uniform distribution have two distribution parameters shown under the **Distribution** list, such as the **Mean** and **Standard deviation** for a normal distribution, **Shape** and **Scale** for a gamma distribution etc. You specify the distribution parameters in the corresponding edit field. All distributions except the uniform distribution and beta distribution have **CDF-Lower**, **CDF-Upper** shown under the **Distribution** list.

- In the **CDF-Lower** list, choose the cumulative distribution function level for your lower bound: **30%**, **10%**, **1%**, **0.1%** (the default), **1E-4**, **1E-5**, **1E-6**, **1E-7**, or **Manual**. These bounds automatically compute a lower bound by using the inverse cumulative distribution function.

- In the **CDF-Upper** list, choose the cumulative distribution function level for your upper bound: **70%**, **90%**, **99%**, **99.9%** (the default), **1-1E-4**, **1-1E-5**, **1E-6**, **1-1E-7**, or **Manual**. These bounds automatically compute an upper bound by using the inverse cumulative distribution function.

- For Manual bounds, you can enter bounds and units for the input parameter in the **Lower bound**, **Upper bound**, and **Unit** columns. For the bounds, you can use unit syntax such as `0.45[mm]`, and for the unit, add its abbreviation, such as `Pa` for pascal.

You can edit the table using the buttons under the table:

- In general, use the **Move Up** (↑), **Move Down** (↓), and **Delete** (≣) buttons and the fields under tables to edit the table contents. Or right-click a table cell and select **Move Up**, **Move Down**, or **Delete**.

- The **Add** button (+) adds a new input parameter to the list.

- Use the **Clear Table** button (✎) to clear the entire table.

SURROGATE-BASED MONTE CARLO ANALYSIS

The **Surrogate-Based Monte Carlo Analysis** section is available for surrogate models used in **Sensitivity analysis**, **Uncertainty propagation**, and **Reliability analysis, EGRA**. For **Sensitivity analysis**, it is available for the **Adaptive Gaussian process** and **Gaussian process** surrogate models. **Sensitivity analysis**, **Uncertainty propagation**, and **Reliability analysis, EGRA** are all sample-based analysis, which means a large number of samples are often

required to achieve high accuracy of the analysis results. Given the computational cost, it is always forbidden to run a large number of COMSOL model evaluations to a large-scale simulation problem. Here, Monte Carlo analysis performs repeated evaluations using the surrogate model with random input parameter values, according to their distribution settings. A COMSOL model evaluation is only needed to build the surrogate model and is not needed in this analysis.

By default, the **Monte Carlo parameters source** is set to **From surrogate model quantities of interest table**. Change it to **Manual** enable a separate Monte Carlo parameters table below, where you can specify parameters for the Monte Carlo analysis.

When **From surrogate model quantities of interest table** is chosen for the **Monte Carlo parameters source** or **Source type** for all parameters are **Analytic** when **Manual** is chosen for the **Monte Carlo parameters source**, you can enter a suitable number of samples for the Monte Carlo analysis (default: 10,000 samples) in the **Number of samples** field. Otherwise, the **Number of samples** field is automatically populated based on the number of samples from **Specified value** or data **Column** selected from the surrogate-based Monte Carlo analysis data source table.

When you have selected a **Reliability analysis, EGRA** study type and **From surrogate model quantities of interest table** is chosen for the **Monte Carlo parameters source** or **Source type** for all parameters are **Analytic** when **Manual** is chosen for the **Monte Carlo parameters source**, you can choose **Importance sampling** (the default) or **Latin hypercube sampling**. With the latter, the same **Number of samples** field appears. With **Importance sampling** selected, you can enter an **Initial number of samples** (default: 1000), a **Maximum number of samples** (default: 10,000), and a **Relative tolerance** (default: 0.05).

If desired, select the **Monte Carlo random seed** check box to enter a random seed in the **Monte Carlo random seed** field. Another seed can be used to generate another set of random samples, to check how sensitive the UQ analysis result is to the particular set used. Ideally, the results should not change much. If they do, you should consider increasing the number of samples.

When **Manual** is chosen for the **Monte Carlo parameters source**, the Monte Carlo parameters setting table is shown under **Monte Carlo parameters source** list.

From the **Source type** list, choose **Analytic** (the default) or **Data**. All settings related to the **Manual** settings for the **Monte Carlo parameters source** are the same as those in the **Input Parameters** section with one difference, where the difference is that **Nominal value** can be selected from the **Data source** list. When **Nominal value** is selected, you can specify a nominal value of the corresponding parameter to be used for the Monte Carlo analysis.

Using **Manual** settings for the Monte Carlo analysis other than what has been specified under input parameters is an advanced feature, and needs to be used with some care. The specification given here will be used for the UQ analysis. It is therefore possible to modify the specification for the input parameters and get new UQ analysis results through the compute action **Analyze only**, without making any new COMSOL model evaluation. Notice, however, that this will only be accurate if the surrogate model is also an accurate model for the new input parameter specification. One simple case where this approach can fail is if the surrogate model is constructed or trained for a smaller domain with stricter bounds, and here is used for a significantly larger domain that involves extrapolation.

SURROGATE MODEL VERIFICATION

The error estimation for the surrogate models are estimations based on the known QoI data used to train the model. The verification error here defined the error, which directly measures how well the surrogate model predicts the QoIs unknown to the model. The **Surrogate Model Verification** section is only available for surrogate models used in **Sensitivity analysis** and **Uncertainty propagation**. It is not available for **Reliability analysis, EGRA** because the surrogate model for reliability analysis is not optimized for the global accuracy.

From the **Verification quantities of interest table** list, choose **New** for a new table, or choose an existing table. This table will contain the verification sample points, as well as COMSOL model evaluation of the QoIs, followed by the surrogate model predictions. The verification analysis will also produce the root mean square (RMS) of the difference between the model evaluations and the predictions as well as the relative RMS difference. These numbers can be found in the **UP verification error** table under the UQ analysis table group, one row per QoI. The RMS difference can be found in the **Verification error** column and the relative RMS difference in the **Relative verification error** column.

By default, the **Verification parameters source** is set to **From surrogate model quantities of interest table**. Change it to **Manual** enable a separate verification parameters table below, where you can specify parameters for the surrogate model verification. This can be useful, for example, if you want to verify the surrogate model in a certain part of the input parameter domain by using other bounds. Notice, however, that the surrogate model is trained under certain conditions and assumptions regarding the input parameters, so it is natural that the predictability (and error) for other input parameter settings is not as good.

When **From surrogate model quantities of interest table** is chosen for the **Verification parameters source** or **Source type** for all parameters are **Analytic** when **Manual** is chosen for the **Verification parameters source**, you can enter the desired number of verification points (default: 10) in the **Number of verification points** field. This is the number of independent COMSOL model evaluations performed for the verification of a surrogate model. The sampling of points will be different from the one used to build the model. For example, another random seed will be used. Otherwise, the **Number of verification points** field is automatically populated based on the number of points from **Specified value** or data **Column** selected from the surrogate model verification data source table.

From the **Random seed type** list, choose **Automatic** (the default) to generate a random seed automatically (that initial random seed is then display below), choose **Manual** to enter a seed in the **Initial random seed** field, or choose **Current computer time** to use that time as the random seed.

SURROGATE-BASED RESPONSE SURFACE

In this section, you can generate response surface data for your surrogate model, which can be used from results for visualization. In this section, you can use the **Response Surface** button and generate a filled COMSOL table. The filled format of the data makes it possible to use a **Table Surface** plot and select two parameters at a time and one QoI. When the table and plot are generated, it is possible to change the two parameters to use and the QoI to visualize. All this is set up for you by the action.

The filled-data structure can be controlled in the four-column table found at the top of the section. The first column, **Parameter**, contains the input parameters. This column is dictated by the surrogate model and cannot be changed. Here, the surrogate model is defined as the one last constructed from the **Quantities of interest table** and is therefore related to the UQ analysis. If the table has surrogate model information stored for this UQ study about how it was constructed and so on, then this column will be populated automatically. If the table does not have such information, it will be empty, and no response surface data can be generated. The second column is the **Point generation method**. For each parameter in this column, you can select **Distribution**, **Specified values**, or **Both**. For **Distribution**, the **Distribution resolution** column can be used to prescribe how many points should be used in this parameter dimension. The actual distribution function will here be used to add points close to the mean value. For the **Specified values** method, the **Parameter value list** column can be used to specify freely which points to use. The **Both** method combines the methods and adds points based on the last two columns. Notice that filled data means that all combinations of

values for the parameters will be generated or predicted. So, for example, if 5 points are generated for m parameters, 5^m rows are generated in the **Response surface data table**.

The target table for the data generation can be set by **Response surface data table**. Since the table data can be large, there are also settings for how to store the table. Select one of the methods in the **Store table** list: **In Model**, **On file**, or **In model and on file**. See Table 3-1 for more information. When a file is used, you also have to give a filename in the **Filename** edit field. There is also a **Maximum file size (MB)** setting where you can limit how much data is stored from the data generation.

The Table Surface plot can be selected from the **Response surface plot group**.

EXPERIMENTAL DATA SETTINGS

In this section, you can specify the experimental data source for the inverse uncertainty quantification study.

From the **Experimental data table** list, choose an existing table that contains the experimental data. This table should contains data columns of measured experimental parameters and the corresponding QoIs.

Under **Experimental data table**, specify the data **Type** for each data column of the table:

- In the **Column** column, the table's column header is automatically populated.
- In the **Type** column, choose **Quantity of interest**, **Experimental parameters** or **Ignored column**.
- In the **Setting** column, the user selected QoI from the **Expression** in the quantities of interests settings and experimental parameter from the **Parameter** in the input parameters settings will be automatically populated.

When **Quantity of interest** is chosen from the **Type** list, the **Name** list is shown under the table, where the list is automatically populated with the **Expression** in the quantities of interests settings; when **Experimental parameters** is chosen from the **Type** list, the **Name** list and **Unit field** are shown under the table, where the **Name** list is automatically populated with the **Parameter** in the input parameters settings; when **Ignored column** is chosen from the **Type** list, the corresponding data column is ignored in the uncertainty quantification study.

No repeated QoI expressions or input parameters should be used as the **Quantity of interest** and **Experimental parameters** respectively. Once the experimental data settings are specified, the **Parameter description** column from the input parameters settings table will be populated with label of calibration parameter or experimental parameter.

The input parameters that are not selected as experimental parameters are labeled as calibration parameters. Since the goal of the inverse uncertainty quantification is to compute the posterior distribution of the calibration parameters, at least one input parameters should be considered as calibration parameter.

OUTPUT WHILE SOLVING

In this section, you control what to output when solving the UQ study.

Select the **Plot** check box to choose a plot group for plotting from the **Plot group** list. This plot group will be updated while solving the underlying COMSOL model for the model evaluations. No plots will be updated during Monte Carlo analysis. When using the **Default** choice, the first automatically added plot group will be used.

Select the **Show UQ results in table graph** check box to display numerical results from the UQ study in a table graph group that you choose from the **Table graph group** list. Choose **New** to create a new table graph group, or choose any existing table graph group.

From the **Probes** list, choose which probes to output values from: **All** (the default), **None**, or **Manual**. If you choose **Manual**, add any available probes to the **Probes** table below.

To output to an accumulated probe table, select the **Accumulated probe table** check box and then choose a probe table for the output from the **Output table** list. Choose **New** to create a new output table, or choose any existing output table.

ADVANCED SETTINGS

From the **Error handling** list, choose **Stop immediately** (the default) to stop the solution process directly if an error occurs, or choose **Skip problematic parameters** to skip problematic parameters where the COMSOL model evaluation fails, and continue the solution process for the UQ study. When **Skip problematic parameters** is chosen, the possible errors are collected and retained in the Uncertainty Quantification Job Configuration node. This node has a warning overlay for its icon when there are collected errors.

From the **Keep model evaluations in memory** list, choose **Only last** (the default) to only keep the last model evaluation, or choose **All** to keep all model evaluations.

From the **Default solver sequence generation** list, choose **Using global parameters** (the default) to use the values of the global parameters or choose **Use each parameter tuple** for a parametric run with parameter tuples. See Parametric Sweep in the *COMSOL Multiphysics Reference Manual* for more information.

Select the **Reuse solution from previous step** check box if desired. This option can be useful for cases where the solutions for the input parameters are close to each other and it benefits the nonlinear solution process to start from the previously computed solution. Notice that because the UQ analysis is global, there can be large differences between these solutions so this option is not always beneficial.

Select the **Distribute model evaluation** check box if you run COMSOL on a distributed system and want to make use of a distributed evaluation. The benefit of distributed evaluation is that during the sampling with the COMSOL model evaluations, these can be done in parallel. However, for the AGP, this is only done for the initial sample points, since only one model evaluation at a time will be performed for subsequent adaptation steps. Without adaptation, or when using the SPCE surrogate model, there is no such limitation.

From the **Uncertainty Quantification log** list, choose **Normal** (the default) to only keep the log from the uncertainty quantification study, choose **Minimal** to only keep minimal log from the uncertainty quantification study, or choose **Detailed** to keep the log from the uncertainty quantification study and the model evaluations from the uncertainty quantification study.

Uncertainty Quantification Job Configurations

Design of Experiments

To display this option, click the **Show More Options** button (👁) and select **Solver and Job Configurations** in the **Show More Options** dialog box.

Add a **Design of Experiments** node to evaluate outputs with the Latin hypercube or Morris sampling methods. The evaluation is done by solving the COMSOL model for certain parameter values. The values for these parameters are generated with these sampling methods. The outputs are any scalar quantity that can be evaluated for the underlying study. Click the **Run** button (=) to run the design of experiments.

GENERAL

When a **Design of Experiment** node is used by UQ, you will only see a disabled setting, **Defined by Uncertainty Quantification job**, which shows which node uses this node and controls all its settings. When this is the case, a **Disconnect from UQ Job and Edit** button is available. By clicking this button, you can take over the control of the node.

The main setting at the top of the section is the **Sample method** list. Choose between **Latin hypercube** (the default), **Morris**, and **Specified**. For the **Latin hypercube** method, you can use the **Number of model evaluations** to set how many model evaluations you want to do in total. A global optimization method is used to find a Latin hypercube of good quality. Use **Maximum number of iterations for LHS** to specify how many steps to allow for the global optimizer for each initial configuration. Use **Number of restart points for LHS** to prescribe how many initial configurations to optimize. The best one of these will be used for the sampling. See the section Data Sampling — Latin Hypercube Sampling for more information.

From the **Random seed type** list, choose **Automatic** (the default) to generate a random seed automatically (the random seed used is displayed), choose **Manual** to enter a seed in the **Random seed** field, or choose **Current computer time** to use that time as the random seed. The **Automatic** method adds 1 to the seed for each run. This method is useful if you want reproducible results, but not identical sampling for repeated runs. The **Manual** method gives the same sampling each time you run with the same random seed. The **Current computer time** method typically never gives the same sampling when you run repeatedly.

Choose one of the **Output data methods**: **Recompute** or **Compute and append**. The **Recompute** method clears the **Design** table in the **Output table group** and adds the sample parameter values and the corresponding output variable evaluations to this table. The **Compute and append** method does not clear the table but instead appends to it.

At the bottom, you find a table with the parameter settings. These are described in the Input Parameters settings section for the The Uncertainty Quantification Study.

GLOBAL OUTPUTS

These outputs work as the QoIs for the The Uncertainty Quantification Study. See the Quantities of Interest section.

RESULTS WHILE SOLVING

See The Uncertainty Quantification Study and the Output While Solving section.

ERROR

See the **Error** section for Parametric Sweep (Job Configuration).

CLUSTER SETTINGS

See the **Cluster** section for Parametric Sweep (Job Configuration).

LOG

See the **Log** section for Parametric Sweep (Job Configuration).

Uncertainty Quantification (Job Configurations)

To display this option, click the **Show More Options** button (👁) and select **Solver and Job Configurations** in the **Show More Options** dialog box.

The **Uncertainty Quantification** (🔢) node under **Job Configurations** is used by the UQ study. Most of the settings are shared and synchronized from the study level to the Job Configuration. Another related node is the Design of Experiment Job Configuration, which also shares many of its settings with the study. The organization of the settings is very much the same. The differences between the nodes are mentioned below.

GENERAL

The synchronization from study is dictated by the **Defined by study step** setting at the top of the **General** section. When this setting points to a UQ study, all synchronized properties are grayed out and cannot be edited. By changing this setting to **User**

defined, you can change these and run the UQ analysis directly from the Job Configuration.

This section also has a setting for which **Design of Experiments job** to use. By default, this points to **New** and a node of type Design of Experiments is added. You can also use an existing node of this type.

UNCERTAINTY QUANTIFICATION SETTINGS

For the SPCE and ASPCE surrogate models, you can find a **Maximum polynomial degree** setting (default value: 30), which terminates the increase of order for the SPCE construction.

For the GP and AGP surrogate models, you can find three settings for the training of the GP. These settings are related to the so-called hyperparameters, which can be seen as an internal tuning of the GP to a fixed dataset. These are the **Method for the number of restart points for training** list: **Automatic** (the default) and **Manual**. The **Automatic** method will use $40m$ restart points. For each of these points, a local optimization problem is solved to find the best parameters. With the **Manual** method, specify the **Number of restart points for training**. With the **Relative tolerance for training**, you can set the termination tolerance for the local optimizer.

All the other settings are the same as for The Uncertainty Quantification Study.

QUANTITIES OF INTEREST

All settings are the same as for The Uncertainty Quantification Study.

INPUT PARAMETERS

There are two Latin hypercube sampling settings that are used by the used **Design of Experiments** node: the **Maximum number of iterations for LHS** and **Number of restart points for LHS**. These settings are related to the problem of finding an optimal set of sampling points. See Design of Experiments for more information.

All the other settings are the same as for The Uncertainty Quantification Study.

SURROGATE-BASED MONTE CARLO ANALYSIS

All settings are the same as for The Uncertainty Quantification Study.

SURROGATE-BASED RESPONSE SURFACE

All settings are the same as for The Uncertainty Quantification Study.

EXPERIMENTAL DATA SETTINGS

All settings are the same as for The Uncertainty Quantification Study.

RESULTS WHILE SOLVING

All settings are the same as those in the **Output While Solving** section for The Uncertainty Quantification Study.

CLUSTER SETTINGS

The check box **Distribute parameters** is synchronized to the **Distribute model evaluation** setting in the **Advanced** section setting for The Uncertainty Quantification Study. There is also a **Maximum number of groups** setting, which is not used when synchronized to a UQ study. When in **User defined** mode, you can change this and prescribe a value. Groups can be used to split all the evaluations into groups, so that each cluster node evaluates a group of model evaluations together.

ERROR

The main setting in this section is the **Error handling** list. It is synchronized to the setting in the **Advanced** section with the same name for The Uncertainty Quantification Study. All warnings and errors thrown within an uncertainty quantification study are collected in the table in this section. If errors occur for the **Skip problematic parameters** method, the failing parameter values and the corresponding error messages are both collected in the table. When there are warning or error messages collected, this node is also decorated with a Warning overlay on its icon.

4

Glossary

This Glossary of Terms contains modeling terms in an uncertainty quantification context. For general mathematical and finite element terms, and geometry and CAD terms specific to the COMSOL Multiphysics software and documentation, see the glossary in the *COMSOL Multiphysics Reference Manual*. For references to more information about a term, see the index.

Glossary of Terms

adaptive Gaussian process (AGP) A Gaussian process method that adaptively adds new sample points based on the location of the maximum error estimation of the most recent Gaussian process model.

adaptive sparse polynomial chaos expansion (ASPCE) A sparse polynomial chaos expansion surrogate model that adaptively adds new sample points from a *Latin hypercube sampling (LHS)* if error estimation is larger than the relative tolerance.

bivariate correlation (Pearson's correlation) A sample-based sensitivity analysis method that computes the linear relationship between a *QoI (quantity of interest)* and an input parameter.

calibration parameters Global parameters with unknown probability distributions. The distributions of these are determined (calibrated) in an inverse uncertainty quantification study.

compute action An action type that defines how to generate and use sampled input parameter data and model evaluations to perform UQ analysis.

covariance A covariance function encodes the assumption of the function you want to learn with a GP model, for instance, its smoothness and length scale.

cumulative distribution function (CDF) The probability that an uncertain variable is small or equal to a certain value.

efficient global reliability analysis (EGRA) A global reliability analysis method that balances the exploitation of the surrogate model and exploration of the unobserved region to maximize the accuracy of the surrogate model near the location where the QoIs are close to the threshold, thereby efficiently increasing the accuracy of the reliability analysis.

expected feasibility function (EFF) The error estimation used to find an adaptation point for the AGP used in reliability analysis. The EFF defines the expectation of the sample lying in the vicinity around the limit state where the QoIs are equal to the thresholds.

experimental parameters Global parameters with known probability distributions. These are the parameters both used to create the surrogate model and to produce experimental data in the inverse uncertainty quantification study.

Gaussian process (GP) A popular probabilistic surrogate model that provides both the prediction and the variance of the prediction at every point sampled from the input parameter space.

importance sampling A method that, through sampling, forms a distribution that overweights an importance region. The method is used for reliability analysis.

input parameters Global parameters with uncertainties. These need to be specified before the model can be used to conduct uncertainty analysis.

inverse uncertainty quantification A UQ analysis that computes the posterior distribution of the calibration parameters based on experimental data and prior distributions of the calibration parameters.

kernel density estimation (KDE) A numerical technique for estimating the probability density function based on sample data. The method is formulating the probability density functions by using a known kernel function.

Latin hypercube sampling (LHS) A method used to generate sample data with good space filling in the input-parameter space.

Markov chain Monte Carlo (MCMC) A method that forms a chain of samples from the posterior distribution for the calibration parameters in the inverse uncertainty quantification study.

model evaluation A COMSOL Multiphysics computation for the sampled input parameter values and the QoIs evaluated based on the corresponding solution.

Morris one-at-a-time method (MOAT) The sample-based global screening method used to rank the importance of the input parameters.

partial correlation A sample-based sensitivity analysis method that computes the monotonic relationship between a QoI and an input parameter.

probability density function (PDF) The probability weight function for an uncertain variable.

quantity of interest (QoI) The output from the COMSOL Multiphysics model evaluation that provides data for the uncertainty analysis.

rank bivariate correlation (Spearman's correlation) A sample-based sensitivity analysis method that computes the linear relationship between the ranking of a QoI and the ranking of an input parameter.

rank partial bivariate correlation A sample-based sensitivity analysis method that computes the monotonic relationship between the ranking of a QoI and the ranking of an input parameter.

reliability analysis A UQ analysis that computes the probability that the QoIs satisfy a condition defined by thresholds.

response surface A postprocessing method that displays the relationship between each QoI and all the input parameters. Two parameters at a time can be visualized in a Table Surface Plot.

screening An UQ analysis that qualitatively ranks the importance of all input parameters for each of the QoIs separately.

sensitivity analysis A UQ analysis that quantitatively computes the influence of all input parameters for each of the QoIs separately.

sparse polynomial chaos expansion (SPCE) A polynomial chaos expansion surrogate model that finds the sparse representation of a multivariate orthonormal polynomial basis.

Sobol method A sensitivity analysis method that decomposes the variance of each QoI into a sum of contributions from the input parameters and their interactions.

surrogate model A class of inexpensive-to-evaluate models used instead of a COMSOL Multiphysics model for Monte Carlo-type UQ analysis.

uncertainty propagation A UQ analysis that propagates the uncertainty in inputs to the uncertainty in the outputs (QoIs).

verify action An action type that defines how to generate and use sampled data and to verify the accuracy of a surrogate model.

Index