National Physical Laboratory

NPL REPORT MS 23

HOW TO IMPLEMENT METHODS FOR UNCERTAINTY EVALUATION, WITH AN APPLICATION TO THE MEASUREMENT OF BRAIN PERFUSION USING A CFD-MRI SIMULATION

NADIA SMITH

AUGUST 2018

How to implement methods for uncertainty evaluation, with an application to the measurement of brain perfusion using a CFD-MRI simulation

Nadia Smith
Data Science Department

ISSN 1754-2960

National Physical Laboratory
Hampton Road, Teddington, Middlesex, TW11 0LW

Approved on behalf of NPLML by
Louise Wright, Science Area Leader, Data Science

CONTENTS

CONVENTIONS AND NOTATIONS

Abbreviations

The following abbreviations are used in this report

A4I	Analysis for Innovators
ASL	Arterial Spin Labelling
CBF	Cerebral Blood Flow
CFD	Computational Fluid Dynamics
JCGM	Joint Committee for Guides in Metrology
GSP	Gold Standard Phantoms
GUM	Guide to the Expression of Uncertainty in Measurement
LHS	Latin hypercube sampling
MCM	Monte Carlo method
MRI	Magnetic Resonance Imaging
NEL	National Engineering Laboratory
NPL	National Physical Laboratory
PDF	Probability density function
RSM	Response surface methodology
VIM	International Vocabulary of Metrology – Basic and general concepts and associated terms

Notation

The following notation is used in this report

$\widehat{D}_i$	Estimator of the variance of the conditional expectation of Y given X_i
$\widehat{D}$	Estimator of the total variance of Y
E_j	Main effect associated with input quantity X_j
$E(Y)$	Expectation of Y
$\hat{F}_0$	Empirical mean of y
M	Number of trials of a Monte Carlo method
m_{Ei}	Mean effect for the ith input quantity of an input screening design
$\mathrm{N}(\mu, \sigma^2)$	Gaussian (normal) distribution with expectation μ and variance σ^2
S_i	First order sensitivity index
$S_{i,j}$	Second order sensitivity index
T_i	Total sensitivity index
$u(y)$	Standard measurement uncertainty associated with y
$u(x_i, x_j)$	Covariance associated with the pair of estimates x_i and x_j
$V(Y)$	Variance of Y
X_i	Input quantity, upon which the measurand depends
x_i	Estimate of the input quantity X_i
Y	Measurand, defined as the output quantity or quantity intended to be measured
y	Estimate of the measurand Y

1 SCOPE

This report explains the implementation of methods for uncertainty evaluation, with an application to the measurement of brain perfusion using a Computational Fluid Dynamics (CFD)-Magnetic Resonance Imaging (MRI) simulation. This report is the main output of the activity between the National Physical Laboratory (NPL) and Gold Standard Phantoms (GSP) as part of the Innovate UK Analysis for Innovators (A4I) project to create a *traceable calibration for an MRI perfusion measurement*. The project is a collaboration between GSP, NPL and the National Engineering Laboratory (NEL). Further details can be found on GSP's web site [1].

2 DESCRIPTION OF THE PROJECT

2.1 GOLD STANDARD PHANTOMS' PROBLEM

Perfusion is the rate of delivery of arterial blood to an organ, and is a useful biomarker of health and disease in the brain, liver, heart and kidneys, and is of clinical importance for dementia, stroke, cerebrovascular disease, and cancer. It can be measured by MRI using a technique known as Arterial Spin Labelling (ASL), which, unlike other medical imaging perfusion measurement techniques, does not require the injection of a gadolinium-based contrast agent, and can therefore be repeated without risk for the patient. ASL perfusion measurements are quantitative - the images obtained represent a physical quantity that has a clinical significance. However, due to the lack of a standard with which to validate and calibrate such a measurement, ASL has not yet seen major clinical uptake, despite its advantages over other techniques.

GSP have developed a unique product in the form of a physical phantom that simulates the process of perfusion, and is compatible with an ASL perfusion scan. It provides a stable source of perfusion, and can be used to assess the variability and linearity of a perfusion measurement made with an MRI scanner. Importantly, to be able to quantify, and associate uncertainties with, an MRI perfusion measurement in accordance with good metrology practice, the perfusion within the phantom should be known.

GSP do not understand well enough how to account for all the uncertainties in the physical phantom. They cannot say with confidence what is the uncertainty on the MRI measurand (that is, the quantity intended to be measured, i.e., perfusion), and cannot therefore provide the absolute calibration of a perfusion measurement.

GSP have attempted to solve this problem of uncertainty analysis by building a large computational model that relates flow within the phantom to simulated MRI images. It consists of a physical and geometrical model, in which fluid velocity and distribution are simulated using CFD software. The output velocity map from this stage is then used to compute contrast agent kinematics with a particle simulation, from which a simulated ground truth MRI data set is obtained. There are many stages in this process (see Figure 1), all of which have assumptions, associated uncertainties, and systematic effects, and because of the sheer complexity of the problem they do not know how to account for all of these. Since they feel they lack domain knowledge in measurement and uncertainty evaluation they do not know how to simplify the problem so that it can be solved while retaining a meaningful result.

2.2 THE ANALYSIS FOR INNOVATORS PROJECT

The project aimed to address GSP's challenge by validating key aspects of their CFD-MRI model, and building an uncertainty budget. They partnered with NPL and NEL to address this challenge and both partners provided consultancy to GSP, for GSP to implement and learn as much of the process as possible.

NEL's role was to validate GSP's CFD model, reviewing assumptions and quantifying their impacts, cross-validating with their own CFD package, and provide guidance on the uncertainty coming from this stage of the process.

NPL's role was to support GSP in developing an uncertainty budget, by explaining and teaching them how to use existing methods for quantifying uncertainty developed within the metrology community. In Section 3 a brief introduction to uncertainty evaluation is given, with pointers to the main references. In Section 4 methods for uncertainty evaluation are briefly explained, including the GUM uncertainty framework and a Monte Carlo method. In Section 5 methods for uncertainty evaluation for computationally expensive models, when a Monte Carlo method may not no longer be practical, are described. Finally, in Section 6 a suggestion of how to tackle the problem of implementing methods for uncertainty evaluation for the measurement of brain perfusion using a CFD-MRI simulation is given.

3 INTRODUCTION TO UNCERTAINTY EVALUATION

The approach to uncertainty evaluation adopted for this project, and throughout this report, is in accordance with the *Guide to the Expression of Uncertainty in Measurement* (GUM) [2] and its Supplements [3] [4]. The metrology vocabulary employed in this report follows that of the *International Vocabulary of Metrology* (VIM) [5].

3.1 THE GUM, ITS SUPPLEMENTS AND COMPLEMENTARY REPORTS

The GUM [2] is the primary document regarding the evaluation and reporting of uncertainty in measurement. It was prepared and published by the Joint Committee for Guides in Metrology (JCGM), alongside its supplementary documents [3] [4]. All of these documents can be downloaded from the BIPM website [6].

Although the GUM is mainly concerned with measurement, its methodology can be applied to evaluate the uncertainty for a measurand defined by a mathematical model that is implemented as a computer-based simulation of a measurement, as is the case here. In the work reported here we have aimed as far as possible to follow the procedure for evaluating and expressing uncertainty that is set out in clause 8 of the GUM [2]. In addition, we aim to characterise completely the input quantities to the model that links the measurand to those input quantities by means of probability distributions, as this allows us to employ Monte Carlo methods for uncertainty evaluation [3].

NPL has produced several good practice guides and reports on software support for uncertainty evaluation [7] [8], which provide software specifications for the evaluation of measurement uncertainty. These can be downloaded from the NPL website [9].

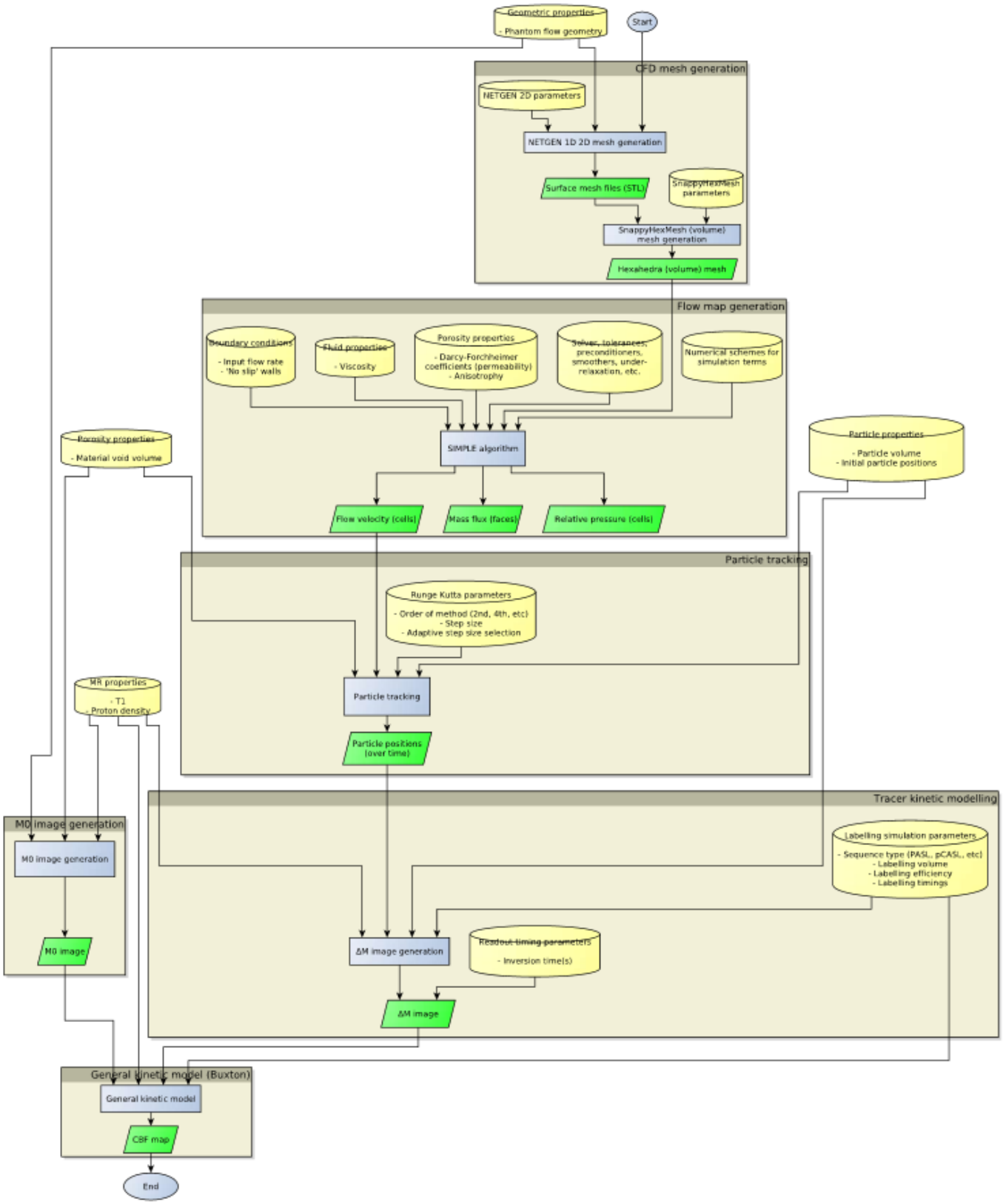

Figure 1: Flowchart of the different steps within the CFD-MRI simulation of a cerebral blood flow (CBF) measurement developed by GSP.

3.2 SOME UNDERLYING PRINCIPLES

The following paragraphs are some useful extracts from the GUM about:

- Why uncertainty analysis is needed: "When reporting the result of a measurement of a physical quantity, it is obligatory that some quantitative indication of the quality of the result be given so that those who use it can assess its reliability. Without such an indication, measurement results cannot be compared, either among themselves or with reference values given in a specification or standard. It is therefore necessary that there be a readily implemented, easily understood, and generally accepted procedure for characterizing the quality of a result of a measurement, that is, for evaluating and expressing its uncertainty." ([2], Introduction, paragraph 0.1).

- How to approach uncertainty evaluation: "Although (the GUM) provides a framework for assessing uncertainty, it cannot substitute for critical thinking, intellectual honesty and professional skill. The evaluation of uncertainty is neither a routine task nor a purely mathematical one; it depends on detailed knowledge of the nature of the measurand and of the measurement. The quality and utility of the uncertainty quoted for the result of a measurement therefore ultimately depend on the understanding, critical analysis, and integrity of those who contribute to the assignment of its value." ([2], paragraph 3.4.8).

- Working definition of uncertainty: "Uncertainty of measurement is an expression of the fact that, for given measurand and given result of measurement of it, there is not one value but an infinite number of values dispersed about the result that are consistent with all of the observations and data and one's knowledge of the physical world, and that with varying degrees of credibility can be attributed to the measurand" ([2], D5.2).

- Main stages of uncertainty evaluation: "The main stages of uncertainty evaluation constitute formulation, propagation, and summarizing

  a) Formulation:
     1) define the output quantity Y, the quantity intended to be measured (the measurand);
     2) determine the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^{\mathrm{T}}$ upon which Y depends;
     3) develop a model relating Y and $\boldsymbol{X}$;
     4) on the basis of available knowledge assign PDFs—Gaussian (normal), rectangular (uniform), etc.—to the X_i. Assign instead a joint PDF to those X_i that are not independent;
  b) Propagation: propagate the PDFs for the X_i through the model to obtain the PDF for Y;
  c) Summarizing: use the PDF for Y to obtain
     1) the expectation of Y, taken as an estimate y of the quantity;
     2) the standard deviation of Y, taken as the standard uncertainty $u(y)$ associated with y ([2], E.3.2);
     3) a coverage interval containing Y with a specified probability (the coverage probability)."([3], section 5.1).

3.3 SOME UNDERLYING CONCEPTS

The following concepts are useful to understanding uncertainty and its evaluation using the approach described in the GUM and its supplements:

- Measurand (or output quantity in a measurement model): The quantity intended to be measured ([5], clause 2.3).

- Measurement uncertainty: Non-negative parameter characterizing the dispersion of the quantity values being attributed to a measurand, based on the information used ([5], clause 2.26).

- Measurement model: Mathematical relation among all quantities known to be involved in a measurement ([5], clause 2.48).

- Input quantity (in a measurement model): Quantity that must be measured, or a quantity the value of which can be otherwise obtained, in order to calculate a measured quantity value of a measurand ([5], clause 2.50). The input quantities are the influence quantities that are the sources of uncertainty for the measurand.

4 METHODS FOR UNCERTAINTY EVALUATION

4.1 THE PROPAGATION OF PROBABILITY DISTRIBUTIONS

The basis for the evaluation of measurement uncertainty is the propagation of probability distributions. In order to apply the propagation of probability distributions, a model of the generic form $Y = f(X_1, \dots, X_N)$ relating input quantities $X_1, \dots, X_N$, about which information is available, and the output quantity Y, about which information is required, is formulated. The model would generally be based on the physical properties underlying the measurement, and incorporate corrections and other effects that influence the measurement. In addition, information concerning the input quantities is encoded as probability density functions (PDFs) for those quantities. The PDFs would be derived from the knowledge available (repeated indications, suppliers' specifications, calibration certificates, expert knowledge, etc.).

Then an implementation of the propagation of probability distributions provides a PDF for the output quantity, Y, from which a best estimate of that quantity can be obtained, as can the standard uncertainty associated with that estimate and a coverage interval for Y corresponding to a stipulated coverage probability. The best estimate and associated standard uncertainty are typically taken as the expectation and standard deviation of Y calculated in terms of the PDF for Y. The PDF also defines a coverage interval for Y that is an interval that contains values of Y with the stated probability. The coverage interval is not uniquely defined, but particular choices include a ‘probabilistically symmetric interval’ (for which the probability that Y takes values less than the left-hand endpoint of the interval is equal to the probability that Y takes values greater than the right-hand endpoint) and the ‘shortest coverage interval’.

Particular implementations of the approach are the GUM uncertainty framework [2] described in section 4.2, and a Monte Carlo method [3], described in section 4.3.

4.1.1. Example of applying the propagation of distributions

We consider two simple examples that are not particularly representative of real measurement problems but serve to compare the propagation of distributions with the GUM uncertainty framework and a Monte Carlo method as particular implementations of the propagation of distributions.

The first example is the simple non-linear model $Y = X^2$ in the case that X is characterised by the standard Gaussian (normal) distribution $\mathrm{N}(0,1)$ having an expectation of zero and a variance of one.

Then, the distribution for the measurand Y provided by the propagation of distributions is known to be the chi-squared distribution with one degree of freedom, and the expectation and standard deviation of Y are, respectively, one and $\sqrt{2}$.

The second example is the simple linear model $Y = X_1 + X_2$ in the case that X_1 and X_2 are independent and each is described by the rectangular distribution on the interval from $-\sqrt{3/2}$ to $\sqrt{3/2}$. Then, the distribution for the measurand Y is known to be the triangular distribution on the interval from $-\sqrt{6}$ to $\sqrt{6}$, and the expectation and standard deviation are, respectively, zero and one.

4.2 THE GUM UNCERTAINTY FRAMEWORK

The GUM presents a framework for uncertainty evaluation based on the use of the law of propagation of uncertainty and the central limit theorem. The law of propagation of uncertainty provides a means for 'propagating uncertainties' through the measurement model, i.e. for evaluating the standard uncertainty $u(y)$ associated with the estimate $y = f(x_1, \ldots, x_N)$ of Y given the standard uncertainties $u(x_i)$ associated with the estimates x_i of X_i (and, when they are non-zero, the covariances $u(x_i, x_j)$ associated with pairs of estimates x_i and x_j). The central limit theorem is applied to characterize Y by a Gaussian distribution (or, in the case of finite effective degrees of freedom, by a scaled and shifted t-distribution), which is used as the basis of providing a coverage interval for Y.

There are some practical issues that arise in the application of the GUM uncertainty framework [7] [8]. Firstly, although the GUM uncertainty framework can be expected to work well in many circumstances, it is generally difficult to quantify the effects of the approximations involved, which include the linearisation of the model in the application of the law of propagation of uncertainty, the evaluation of effective degrees of freedom using the Welch–Satterthwaite formula, and the assumption that the output quantity is Gaussian (or a scaled and shifted t-distribution). Secondly, the procedure relies on the calculation of model sensitivity coefficients (the values of the partial derivatives of the model with respect to the input quantities at the estimates of those quantities) as the basis of the linearisation of the model. Such calculation can be difficult when (a) the model is (algebraically) complicated, or (b) the model is specified as a numerical procedure for calculating a value of Y, for example, as the solution to a differential equation.

4.2.1 Example of applying the GUM framework

In the first example of section 4.1.1, applying the law of propagation of uncertainty gives $y = x^2$ for the best estimate of Y, and $u(y) = |2x|u(x)$ for the standard uncertainty associated with y, where $x = 0$ and $u(x) = 1$, i.e., $y = 0$ and $u(y) = 0$. It is unreasonable that the uncertainty for Y is zero when Y depends on a quantity whose value is not known exactly, but this arises because the linearisation of the measurement model used in the law of propagation of uncertainty is inadequate in this case.

In the second example of section 4.1.1, applying the law of propagation of uncertainty gives $y = x_1 + x_2$ for the best estimate of Y and $u^2(y) = u^2(x_1) + u^2(x_2)$ for the (squared) standard uncertainty associated with y, where $x_1 = x_2 = 0$ and $u(x_1) = u(x_1) = \sqrt{1/2}$, i.e., $y = 0$ and $u(y) = 1$. In this case the law of propagation of uncertainty reproduces exactly the results from the propagation of distributions, which is expected for a linear model. However, applying the GUM uncertainty framework, Y is characterized by the Gaussian distribution $\mathrm{N}(y, u^2(y))$, which does not reproduce the triangular distribution provided by the propagation of distributions. Figure 2 compares the probability density functions for the two distributions and the positions of the endpoints of, respectively, a 95 % and 99 % coverage interval. The differences between the endpoints for a 95 % coverage probability are not too large. However, the 99 % coverage interval is wider than the triangular distribution provided by the propagation of distributions and, consequently, achieves a 100 % coverage probability for Y.

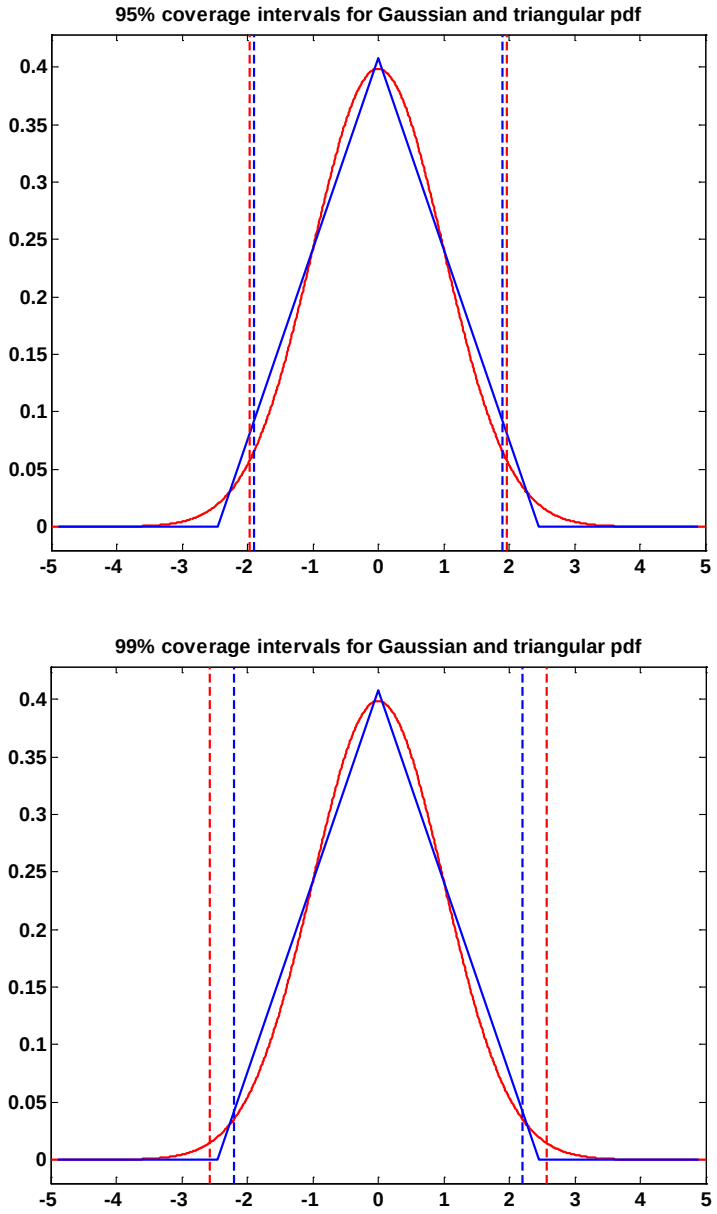

Figure 2: Comparison of the probability distributions returned by the propagation of distributions (triangular distribution in blue) and the GUM uncertainty framework (Gaussian distribution in red) for the example of a simple additive model $Y = X_1 + X_2$ in which X_1 and X_2 are characterised by rectangular distributions. The vertical lines (also in blue and red, respectively) show the endpoints of the coverage intervals having coverage probabilities of 95 % (top) and 99 % (bottom) obtained from the two distributions.

4.3 A MONTE CARLO METHOD

A Monte Carlo method for uncertainty evaluation operates in the following manner [7] [10]. A random draw is made from the probability distribution for each X_i and the corresponding value of Y is formed by evaluating the model for these values. Many Monte Carlo trials are performed, i.e. the process is repeated many times, to obtain M, say, values $y_r, r = 1, \ldots, M$, of Y. Finally, the values y_r are used to provide an approximation to the probability distribution for Y.

The method has a number of features [7], including (a) that it is applicable regardless of the nature of the model, i.e. whether it is linear, mildly non-linear or highly nonlinear, (b) that there is no requirement to evaluate effective degrees of freedom and (c) that no assumption is made about the distribution for Y, for example, that it is Gaussian. In consequence, the method provides results that are free of the approximations involved in applying the GUM uncertainty framework, and it can be expected, therefore, to provide an uncertainty evaluation that is reliable for a wide range of measurement problems. Additionally, the method does not require the calculation of model sensitivity coefficients since the only interaction with the model is to evaluate the model for values of the input quantities.

However, there are also some practical issues that arise in the application of a Monte Carlo method [8]. The degree of numerical approximation obtained for the distribution for Y is controlled by the number M of trials, and a large value of M (perhaps 10^5 or 10^6 or even greater) may sometimes be required. One issue, therefore, is that the calculation for large values of M may not be practical, particularly when a (single) model evaluation takes an appreciable amount of time. Another issue is that the ability to make random draws from the distributions for the X_i is central, and the use of high-quality algorithms for random-number generation gives confidence that reliable results are provided by an implementation of the method. In this regard, the ability to draw pseudo-random numbers from a rectangular distribution is fundamental in its own right, and also as the basis for making random draws from other distributions using appropriate algorithms or formulae.

4.3.1 Example of applying a Monte Carlo method

In the first example of section 4.1.1, applying a Monte Carlo method with $M = 10^6$ trials involves making M draws x_r independently from $\mathrm{N}(0,1)$ and evaluating $y_r = x_r^2, r = 1, \ldots, M$. The mean and standard deviation of the values y_r are, respectively, 1.00 and 1.41 (to three significant decimal digits), which agree well with the expectation and standard deviation (of one and $\sqrt{2}$, respectively) of the chi-squared distribution provided by the propagation of distributions. The top graph in Figure 3 compares the probability density function of the chi-squared distribution with an approximation to the probability density function for Y constructed from the histogram of y_r values.

In the second example of section 4.1.1, applying a Monte Carlo method with $M = 10^6$ trials involves making M draws $x_{1,r}$ and M draws $x_{2,r}$ independently from the rectangular distribution on the interval from $-\sqrt{3/2}$ to $\sqrt{3/2}$ and evaluating $y_r = x_{1,r} + x_{2,r}, r = 1, \ldots, M$. The mean and standard deviation of the values y_r are, respectively, –0.00208 and 0.999 (to three significant decimal digits), which agree well with the expectation and standard deviation (of zero and one, respectively) of the triangular distribution provided by the propagation of distributions. The bottom graph in Figure 3 compares the probability density function of the triangular distribution with an approximation to the probability density function for Y constructed from the histogram of y_r values.

4.4 MULTI-STAGE MEASUREMENT MODELS

Multi-stage measurement models are widespread and relevant to the GSP problem. Any situation in which the output quantities from one uncertainty evaluation become the input quantities to a subsequent evaluation constitute (part of) a multi-stage measurement model. Within a measurement model there are frequently sub-models, and therefore multi-staging arises also in this context. In the first stage of a multi-stage measurement model, it is necessary to provide information about all the input quantities. In subsequent stages, the input quantities constitute some or all of the output quantities from previous stages plus, possibly, further input quantities. When applying a Monte Carlo method in the context of a multi-stage model, information about the output quantity in the first stage takes the form of a (large) set of values $z_r, r = 1, \ldots, M$. In subsequent stages, the quantity (now considered as an input quantity) is treated by sampling at random, and with replacement, from the set of values. Alternatively, the set of

values can be used to construct an approximation to the distribution function for the quantity (for example, in the form of a piecewise straight-line function) and used as the basis for making random draws from the probability distribution for the quantity.

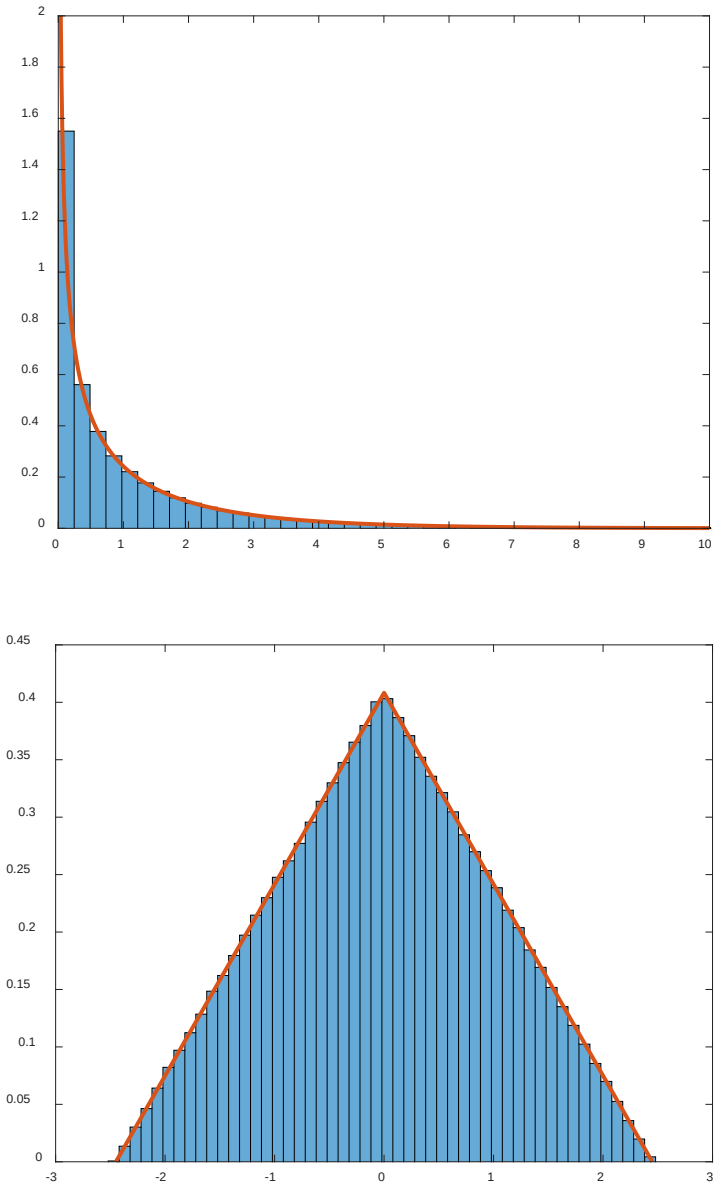

Figure 3: For the two examples, comparison of the probability distributions provided by the propagation of distributions (in red) and a Monte Carlo method (in blue) with $M = 10^6$ trials.

5 UNCERTAINTY EVALUATION FOR COMPUTATIONALLY EXPENSIVE MODELS

The use of computationally expensive models presents a challenge for uncertainty evaluation because such models are frequently nonlinear, so the GUM approach cannot be applied reliably, and their expense makes a Monte Carlo method unfeasible. This section offers some information on uncertainty evaluation for such models, and Figure 4 shows a flow chart illustrating the steps one should take to do so.

In this section it is assumed we have a model of the form

$$\mathbf{Y} = \mathbf{F}(\mathbf{X})$$

where $\mathbf{X} = (X_1, X_2, \ldots, X_N)^{\mathrm{T}}$ is an input vector of quantities, described as random variables of known joint probability distribution, and $\mathbf{Y} = \left(Y_1, X_2, \ldots, Y_{M_{out}}\right)^{\mathrm{T}}$ is an output vector of quantities for which it is required to determine the joint distribution. It is assumed that the function **F** is a black box, so that evaluation of **F** is possible but algebraic evaluation of its derivatives with respect to the inputs is not.

As in section 4, values or estimates of an input or an output quantity are denoted in lower case, so that x_i is a value of X_i the variable. The use of this notation becomes important when distinguishing between a quantity that is a function of the input quantities X_i, $i = 1, 2, \ldots, N$, and a value that has been calculated using a set of sample values x_i, $i = 1, 2, \ldots, N$.

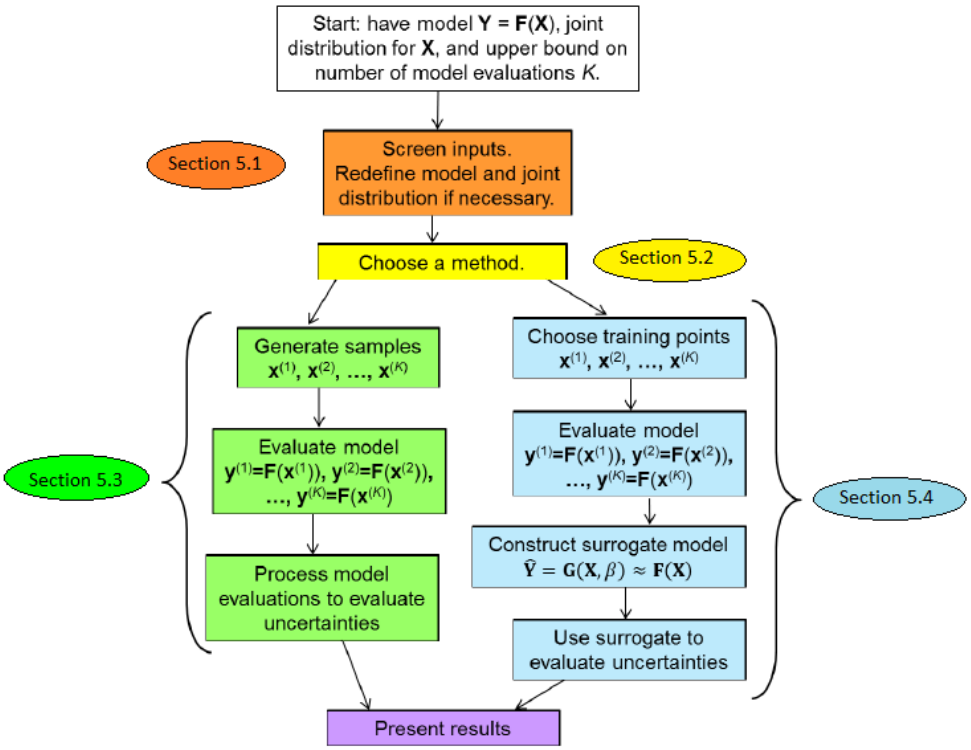

Figure 4: Flow chart illustrating the steps when evaluating the uncertainties associated with the results of a computationally expensive model, addressed in this section.

5.1 SENSITIVITY ANALYSIS AND INPUT SCREENING

Sensitivity analysis provides a rigorous framework to support exploration of the relationship between the model input quantities and the model output quantities. Input screening is a simplified form of sensitivity analysis that allows the user to identify the most important input quantities and potentially to reformulate the model so that only the most important input quantities are treated as uncertain.

The input screening process has several benefits as a preparatory step for uncertainty evaluation. The main benefit is a reduction in complexity of the problem, but the results of the process can also lead to a clearer understanding of the links between the model input quantities and output quantities, particularly for complicated models that use “black box” software. Some uncertainty evaluation methods become

difficult to use for a large number of input quantities, either due to instability or high computational cost, so input screening can lead to a broader range of possible methods being available.

Input screening involves evaluation of the model and thus has an associated computational cost. In some cases this cost will be recouped by the ability to use a simpler model after the input quantities have been screened. If the screening suggests that all the model input quantities are necessary, it may still be possible to reuse the model evaluations made for input screening as part of the subsequent uncertainty evaluation process, for instance as part of a random sample or as training points for a surrogate model.

5.1.1 Input screening

One of the most conceptually simple methods for input screening is the full factorial design. An n-level full factorial design selects n values for each of the model inputs and evaluates the model using every possible combination of these values of the inputs. For a model with N inputs, a full factorial design requires n^N model evaluations. It is common to set $n = 2$ and to refer to the higher value of each input quantity as the positive value and the lower as the negative value. A two-level design assumes the output quantity behaves approximately linearly in response to variations in the input quantities. It is common to test this assumption by carrying out an extra model evaluation at a centre point, for instance using the mean value of each of the input quantities. All of the subsequent material in this section, unless otherwise stated, discusses two-level designs, but similar analysis techniques can be applied to the results of designs with more levels.

The full factorial design can be used to generate a model of the form

$$Y = \beta + \sum_{i=1}^{N} \beta_i X_i + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \beta_{ij} X_i X_j + \cdots + \beta_{123\ldots N} X_1 X_2 \ldots X_N$$

for each output quantity Y, where the $\beta_i, i = 1, 2, \ldots N$, are called the main effects associated with the input quantities (i.e. the extent to which an input quantity affects the output quantity under the assumption of approximate linearity), and the $\beta_{ij}\ i = 1, 2, \ldots N-1,\ j = i+1, i+2, \ldots N$, are the effects of the two-input interactions.

To get a quantitative assessment of the main effects associated with each input quantity or of an interaction, the mean response to an input quantity or interaction at its positive level minus the mean response at the negative level is calculated. For a main effect, if we order the output quantity values y_i such that $y_1, y_2, \ldots, y_p$, where $p = 2^{N-1}$, were obtained from the full model using the high (positive) value of X_j , and $y_{p+1}, y_{p+2}, \ldots, y_{2p}$ were obtained using the low value of X_j, then the main effect associated with X_j is calculated from

$$E_j = \frac{1}{p}\left(\sum_{i=1}^{p} y_i - \sum_{i=p+1}^{2p} y_i\right)$$

The interaction effect associated with the interaction between X_j and X_i is calculated by ordering the model outputs by the value of the interaction, so that $y_1, y_2, \ldots, y_p$ were obtained from the full model when X_j and X_i are either both high or both low and $y_{p+1}, y_{p+2}, \ldots, y_{2p}$ were obtained when one of X_j and X_i is high and the other is low, and calculating the quantity E_{ij} using the expression for E_j above. Similar interaction effects can be obtained for higher-order interactions in the same way.

In order to evaluate if these effects are significant an estimate of the standard error is required, determined by

$$s^E = \frac{s}{\sqrt{2^N}}$$

where s is the standard deviation of the 2^N responses. Any effect greater in absolute value than the standard error is deemed to be significant. It should be noted that it is possible for an input quantity to be significant in its interactions but not as a main variable. As an example, consider the function $X_1X_2 - X_1X_3$ and the choices $x_1^p = x_2^p = x_3^p = 1,\ x_1^m = x_2^m = x_3^m = -1$. The calculated main effect for all of the quantities X_i is zero, but it is obvious from the function that all of the input quantities are significant. For further details of input screening of this example, please see Section 2.3 of [11].

For a computationally expensive problem, a full factorial design may be too time-consuming to perform. In such cases, the number of model evaluations can be reduced by using a fractional factorial design, which requires 2^{N-m} model evaluations where m is the level of reduction. A good general introduction to fractional factorial designs is given on the NIST website [12] or in "Sensitivity Analysis" by Saltelli et al [13].

Fractional factorial designs are created by choosing a subset of the input quantities, defining a set of model evaluations based on every combination of high and low values of those quantities, associating each of the remaining input quantities with an interaction of some combination of the input quantities in the chosen subset, and assigning high or low values for the remaining input quantities according to the value of the interaction.

The drawback of a fractional design is the confounding effect: it becomes impossible to separate the contributions of some combinations of effects and interactions, and in particular the input quantities not in the chosen subset are confounded with interactions of the input quantities that are in the subset. Careful design can help reduce the impact of confounding by ensuring as much as possible that main effects and low level interactions do not confound with each other. Generally, it is likely that physical insight can help to identify the interactions that are most important and hence should not be confounded.

5.1.2 Morris designs and Sobol' indices

Morris designs are input screening designs. The principle is to define a base point and to successively perturb each of the input quantities by a known amount in order to estimate the effect of the varied input quantity. In order to evaluate potential interactions, this process is repeated *R* times, for example see Figure 5 for two input quantities with three repetitions. If an input quantity is involved in an interaction, then its effect will vary according to the value taken by other input quantities so the effects at the three points shown in Figure 5 would be different.

For each run of the design, the effect for the i^{th} input quantity evaluated at the j^{th} repetition is denoted as E_{ij} where

$$E_{ij} = \frac{F\left(x_1^{(j)}, x_2^{(j)}, \ldots, x_i^{(j)} + \Delta x_i^{(j)}, \ldots x_N^{(j)}\right) - F\left(\mathbf{x}^{(j)}\right)}{\Delta x_i^{(j)}}$$

where $\boldsymbol{x}_j = \left(x_1^{(j)}, x_2^{(j)}, \ldots, x_N^{(j)}\right)$ is the j^{th} base point and $\Delta x_i^{(j)}$ is the size of the perturbation of the i^{th} variable. The mean effect for the i^{th} input quantity and the standard deviation are then computed according to

$$m_{Ei} = \frac{1}{R}\sum_{j=1}^{R} E_{ij}$$

$$s_{Ei} = \sqrt{\frac{1}{R-1}\sum_{j=1}^{R}\left(E_{ij} - m_{Ei}\right)^2}$$

Input quantities with a high mean effect m_{Ei} in comparison with the mean error as defined above have a significant influence on the output quantity and those with a high effect standard deviation are likely to be involved in a significant interaction and should not be neglected.

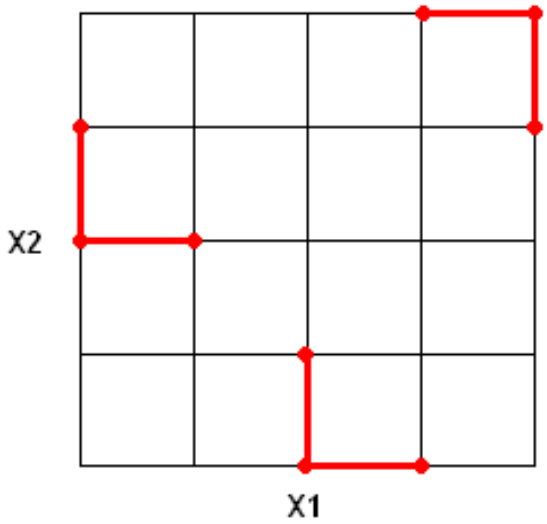

Figure 5: Variation of two input quantities according to a Morris design, for two input quantities and *R* = 3 repetitions.

The Sobol' method is based on the estimation of conditional variances. The total variance decomposition theorem yields that the variance of the output quantity may be written as

$$V(Y) = V(E[Y|X_i]) + E(V[Y|X_i])$$

where $V(Y)$ denotes the variance of the quantity Y and $E(Y)$ is the expectation of Y, the first term $V_i = V(E[Y|X_i])$ denotes the part of the variance of Y that is due to the variations of the input quantity X_i while the second term denotes the part of the variance of Y that is due to variations of all the input quantities apart from X_i. The first and second order sensitivity indices are then defined, respectively, as

$$S_i = \frac{V_i}{V(Y)}$$

$$S_{i,j} = \frac{V\left(E\left[Y|X_i, X_j\right]\right) - V_i - V_j}{V(Y)}$$

The effect of a given input quantity should be accounted for whether it comes from a first, second, or even higher order effect. To include higher order effects, some methods use the total sensitivity index T_i for the input quantity X_i, given by

$$T_i = S_i + \sum_{j \neq i} S_{i,j} + \cdots + S_{1,2,\ldots,N}$$

Sobol' estimation of the sensitivity indices relies on two samples of size M_{sample}, $x_{i,j}, i = 1,\ldots,N, j = 1,\ldots,K$, and $x'_{i,j}$, $i = 1,\ldots,N, j = 1,\ldots,K$, that are mixed together in a deterministic way in order to

obtain an estimate of the sensitivity index. For the first order index associated with X_i, the model is evaluated for all of the $\boldsymbol{x}_j$ and for an altered version of the $\boldsymbol{x}'_j$ where the value of $x_{i,j}$ replaces the value of $x'_{i,j}$ for all j. The first order sensitivity index is estimated by the quantity

$$S_i^{Sobol} = \frac{\widehat{D}_i}{\widehat{D}}$$

where

$$\widehat{D}_i = \frac{1}{M_{sample}} \sum_{k=1}^{M_{sample}} F\left(x_{1,k}, \dots, x_{N,k}\right) F\left(x'_{1,k}, x'_{2,k}, \dots, x'_{i-1,k}, x_{i,k}, x'_{i+1,k}, \dots x'_{N,k}\right) - \hat{F}_0^2$$

is the estimator of the variance of the conditional expectation of Y given X_i,

$$\widehat{D} = \frac{1}{M_{sample}} \sum_{k=1}^{M_{sample}} F^2\left(x_{1,k}, \dots, x_{N,k}\right) - \hat{F}_0^2$$

is the estimator of the total variance of Y, and

$$\hat{F}_0 = \frac{1}{M_{sample}} \sum_{k=1}^{M_{sample}} F\left(x_{1,k}, \dots, x_{N,k}\right)$$

is the empirical mean of y.

$\widehat{D}_{ij}$ may be defined in a similar manner with both $x_{i,k}$ and $x_{j,k}$ replacing $x'_{i,k}$and $x'_{j,k}$ in $\boldsymbol{x}'_k$, leading to estimates of the second order indices (or effect of the interaction between two input quantities) as

$$S_{i,j}^{Sobol} = \frac{\widehat{D}_{ij} - \widehat{D}_i - \widehat{D}_j}{\widehat{D}}$$

Higher order indices can be obtained in the same manner. As these indices are often null, it may be more efficient to compute the total order sensitivity indices to establish whether any significant contributions exist among higher order terms. The estimation of the total sensitivity indices relies on the estimation of the variance of the output due to the variations of all input quantities other than X_i. Sobol' proposes then the corresponding estimate

$$T_i^{Sobol} = 1 - \frac{\widehat{D}_{-i}}{\widehat{D}}$$

where

$$\widehat{D}_{-i} = \frac{1}{M_{sample}} \sum_{k=1}^{M_{sample}} F\left(x_{1,k}, \dots, x_{N,k}\right) F\left(x_{1,k}, \dots, x_{i-1,k}, x'_{i,k}, x_{i+1,k}, \dots, x_{N,k}\right) - \hat{F}_0^2$$

The estimation of Sobol' indices is computationally expensive. For this reason, best practice is to evaluate only the most significant effects, holding the input quantities with negligible influence at their best estimate, in particular for computationally expensive systems. In order to improve the convergence of these sensitivity estimates, smart sampling methods (such as Latin hypercube sampling) can also be used to generate the samples $\boldsymbol{x}_k$ and $\boldsymbol{x}'_k$.

5.2 METHOD CHOICE

Universal rules on choosing a method for uncertainty evaluation are difficult to develop. Each case must be judged separately. This chapter provides a list of factors to be considered when making a choice, discusses the aspects of a problem that can limit the choice of method, and some ways of removing these limitations. The main factors discussed here are:

- Numbers of input and output quantities;
- Computational and mathematical complexity and software availability;
- User knowledge of the model and the input quantities;
- Correlation between the input quantities;
- Existence of historical model evaluations and the need for sample size adaptivity.

It is assumed throughout this section that, as mentioned in the introduction, the user has a known upper bound K on the number of model evaluations that can feasibly be made. The following section mentions methods that have yet to be described. The reader is referred to the relevant subsection of the report for further details.

5.2.1 Factors to consider

It should be noted that the information required by different methods is not necessarily mutually exclusive. In particular, the points that are generated if a sampling method is chosen can often be used as training points for constructing a surrogate model. In some cases the fitting of a surrogate model to sampled results can identify regions where more samples would improve the estimated uncertainties. However, it is not necessarily the case that the surrogate model will give a more accurate estimate of the uncertainty than the direct analysis of the sample results.

One of the most important factors affecting method choice is the number of input quantities. The average distance between a fixed number of randomly chosen points increases as the dimensionality of the space increases, so the sampling points will be distributed more sparsely as the number of input quantities increases. This sparsity can mean that important areas of the input space might not be sampled.

The number of input quantities also strongly affects the performance of surrogate models. Construction of many surrogate models requires the evaluation of parameters for the surrogate model and the number of parameters usually increases at least proportionally to the number of input quantities. The unique determination of a set of P parameters requires at least P model evaluations, so some methods may be ruled out if $P > K$. Parameter determination also adds computational cost to the method, and some parameter determination methods suffer from instability as the number of input quantities increases. These factors may also rule out some methods for models with a large number of input quantities. In some cases, the number of input quantities can be reduced by using input screening to identify any insignificant inputs.

Some methods may not be suitable for models with multiple output quantities. As an example, consider a model with one input X_1 with a rectangular distribution on $[0, 1]$ and two outputs $Y_1 = 1/(0.001 + X_1)$ and $Y_2 = (10X_1)^3$, so that Y_1 is sensitive to small values of X_1 and Y_2 is sensitive to large values of X_1. The different areas of sensitivity may make it difficult to use importance sampling or stratified sampling for both output quantities at once.

Computational complexity and software availability can limit the choice of method. The determination of parameter estimates in surrogate models can add computational complexity. For instance, the parameter evaluation process for Gaussian process models (see section 5.4) involves the solution of a complicated nonlinear inverse model requiring the use of optimisation routines. This computational complexity may make a method too difficult to use, and in particular if the user does not have access to

suitable optimisation routines then the method may not be usable. Similarly, the complexity of the mathematics that underpins the polynomial chaos method can deter potential users. In some cases the complexity can be avoided by collaborating with an appropriate expert, or by using a trusted software package (various automated uncertainty evaluation packages are available) to generate samples and process the results.

User knowledge about the input quantities and the nature of the model can strongly affect the choice of method. A simple example is that if the user knows that the model exhibits strongly nonlinear or discontinuous behaviour, then a quadratic response surface, which assumes a degree of smoothness of the model, is unlikely to give reliable results. Most methods assume that all parts of the input space are equally important. If the user has knowledge that the model output quantities are particularly sensitive to a specific region of the input space then a method such as importance sampling or stratified sampling can be used to ensure that this knowledge is used to direct the sampling.

A lack of knowledge can make use of a particular method challenging. Some methods require the user to make decisions that will strongly affect the results but that involve choices where the best choice is not obvious. Examples include the best choice of subdivision of the input space for stratified sampling and the best choice of proposal density for importance sampling. In such cases it is useful to test the sensitivity of the method to the choice being made, but this approach may not be possible due to the constraints of computational expense.

The nature of the joint distribution associated with the input quantities affects the choice of method. Several methods, in particular polynomial chaos and Latin hypercube sampling, are difficult to use for correlated input quantities. Polynomial chaos also works best when the independent input quantities have distributions that can be related to a family of orthogonal polynomials, so if the distributions do not have a known associated polynomial family then the method will exhibit poor convergence.

Sometimes historical model evaluations may be available that could be used in addition to new model evaluations, but some methods cannot easily include extra data. Similarly if the estimate of the number of model evaluations that is feasible increases (for instance, if a more powerful computer becomes available), some methods struggle to alter the sample size adaptively. In particular, Latin hypercube sampling cannot easily adjust its sample size without imposing correlation between the input quantities, and the use of sparse grid sampling to evaluate the integrals required for the polynomial chaos method means that the model is evaluated at specific values of the input quantities which may not be consistent with historical values.

5.3 SAMPLING METHODS

A sampling method consists of a method for generating samples $\boldsymbol{x}_k$, $k = 1, 2, \ldots, K$, from the space of input variables (the input space), and a method for post-processing the resulting output values $y_k = F(\boldsymbol{x}_k)$ to obtain estimates of some properties of the desired distribution of Y. The simplest possible sampling method is random sampling, also known as Monte Carlo sampling, as described in Section 4 above.

There are many applications, within metrology and beyond, that require models that take upwards of several minutes to run. The computational expense of such models, which is the prime motivation for this section, renders random sampling computationally intractable. However, other sampling methods exist that produce better repeatability for small sample sizes.

Random sampling does not guarantee that a given region of the input space will include a sample point, and it does not exclude the possibility of one or more points being very close together. These features mean that it is not an efficient sampling method because some areas of the input space that may be related to critical values of the output quantities may not be sampled, whilst other less critical areas are sampled repeatedly. For small sample sizes, the estimates of the joint distribution of the output variables

are strongly affected by every sampled point, and so since the samples are not guaranteed to cover evenly the whole of the input space, estimates of the joint distribution of the output variables obtained using different random samples of the same small size K can exhibit a large sample-to-sample variance.

Stratified sampling is a technique that has lower sample-to-sample variance than random sampling and was developed in an effort to address some of the shortcomings of random sampling. Some references appear to use the names “stratified sampling” and “importance sampling” interchangeably. They are considered as separate here, as importance sampling uses a distinctly different approach to sample selection from that used in stratified sampling. Importance sampling is not discussed here as it is unlikely to be of use for the problem of interest.

Stratified sampling divides the complete input space into a number of distinct non-overlapping regions S_i, $i = 1, 2, \ldots, n_S$, and takes I_i, $i = 1, 2, \ldots, n_S$, samples within the i^{th} region, such that

$$\sum_{i=1}^{n_S} I_i = K$$

The regions need not be of equal probability, and it is useful to define the probability of a sample occurring within the i^{th} region as p_i.

The estimates of the means of the output variables are determined from a weighted sum of the results of the model evaluated at each sample point, with the weights being equal to p_i/I_i. The covariances for pairs of such estimates can be obtained from a similar weighted sum.

The main strength of the method is that it allows use of expert knowledge about the underlying model to generate samples that will create the most useful information about the output quantities and to ensure that any regions of particular importance are included in the sampling, whilst maintaining a proper probabilistic consideration of the input distributions. Even when the model is a black box, it is possible that the user will know that some regions of the input space give almost constant values of the output variables and that other regions produce more variation in the output variables and so should be sampled more densely.

The chief drawback of the method is that creating a set of non-overlapping regions of known probability that completely fill the input space efficiently is not simple, particularly for correlated input variables and for problems with a large number of input variables. Another drawback is that if there are multiple output variables of interest then the best choice subdivisions for different output variables could easily be incompatible, meaning that a unique best choice subdivision may not be possible.

Latin hypercube sampling (LHS) was first described by McKay [14]. Latin hypercube sampling is an extension of stratified sampling to ensure that every sub-region of every individual input variable is sampled, and uses a simple method to generate samples spread across the full range of values.

The basic form of the sampling method requires that all input variables be independent. The full range of each input variable is divided into K regions of equal probability, and a sample of each input variable is taken from each region. If an input quantity is not defined on a finite region (e.g., if the quantity has a Gaussian distribution), then one or two of the sub-regions may be infinite. The K samples of each of the N input variables are then randomly arranged into K vectors of length N, with each vector containing exactly one sample of each input variable and each sample being used in exactly one vector. The K vectors are used to generate K vectors of output variable values, and these values are post-processed using the same techniques as are used to post-process random samples.

To generate a Latin hypercube sample of size K for N independent input quantities X_j with cumulative distribution functions $g_j(X_j)$, take the following steps:

- Generate $N \times K$ random samples of a random variable that has a rectangular distribution on the interval $[0,1]$ and group them as N vectors of length K, $\xi_j^{(i)}$, $i = 1,2,\ldots,K$, $j = 1,2,\ldots,N$.
- Generate N different permutations of the numbers $\{1,2,\ldots,K\}$, where $\pi_j^{(i)}, i = 1,2,\ldots,K, j = 1,2,\ldots,N$, is the value in the i^{th} position of the j^{th} permutation. In the absence of a random permutation generator, the easiest way of generating a single such permutation is
  - Generate K random samples $v_1, v_2, \ldots, v_K$ of a random variable V that has a rectangular distribution on the interval $[0,1]$.
  - Sort these numbers in increasing order, and use the resulting rearrangement of the sample indices as the random permutation.
  - For instance, if $K = 4$ and the random samples are v_1 = 0.384, v_2 = 0.027, v_3 = 0.725, and v_4 = 0.168 then the rearranged order is $v_2 < v_4 < v_1 < v_3$, so the permutation is 2, 4, 1, 3.
- Calculate the quantities $u_j^{(i)} = (\pi_j^{(i)} - \xi_j^{(i)})/K, i = 1,2,\ldots K, j = 1,2,\ldots,N$. These quantities are samples of K random variables, each uniformly distributed on one of the K sub-regions of $[0,1]$, arranged in an order determined by the random permutation.
- Construct the vectors of input quantities by setting $x_j^{(i)} = g_j^{-1}(u_j^{(i)})$, $i = 1,2,\ldots,K, j = 1,2,\ldots,N$, and setting the i^{th} vector $x^{(i)}$ to consist of the values $\left(x_1^{(i)}, x_2^{(i)}, \ldots, x_N^{(i)}\right)^{\mathrm{T}}$. Then each vector contains exactly one value of each input quantity, each sub-region of the range of each input quantity is sampled exactly once, and the vectors are assembled randomly.
- Evaluate the model function $\mathbf{F}(\mathbf{X})$ at each of the sampled values to obtain K sets of output quantities, so that $\mathbf{y}^{(1)} = \mathbf{F}\left(\mathbf{x}^{(1)}\right), \mathbf{y}^{(2)} = \mathbf{F}\left(\mathbf{x}^{(2)}\right), \ldots, \mathbf{y}^{(K)} = \mathbf{F}\left(\mathbf{x}^{(\mathrm{K})}\right)$.

LHS combines the space-filling properties and control of stratified sampling with the ease of sample generation of random sampling. In the original paper describing the method [14], it is shown that LHS leads to unbiased estimates for the moments of the output distribution, and that if the function **F** is monotonic in each of its inputs, then the variance of those estimates is lower for LHS than it is for random sampling. Some papers suggest that the variance is better than that of random sampling for a wider range of models. Extensions to LHS that optimize the sampling of points, handle correlation, inequality constraints on the inputs, and that enable sensitivity analysis have been proposed.

The main disadvantages of LHS are the difficulty of treating correlated inputs, and sample size inflexibility. As with stratified sampling, correlated input variables provide a challenge for Latin hypercube sampling. Iman and Conover [15] developed a method to induce a user-defined rank correlation matrix in a set of uncorrelated multivariate samples. The approach assumes that imposing a rank correlation matrix is equivalent to imposing a sample correlation matrix, which is meaningful in most modelling situations but may not be perfect in some cases. The dependence of the hypercubes used in LHS on the sample size K makes changing the sample size difficult, so if there are doubts about the adequacy of the sample size, the trial will have to be rerun from the start. Theoretically it is possible to subdivide each of the K intervals for each input value and sample in the new regions, but this approach would not lead to truly random vectors of input values (since new values would only be input with other new values) and would have a potentially unwanted correlation structure.

5.4 SURROGATE MODELS

A surrogate model is a simplified model that captures the main features of the full model for a certain range of input quantities. Models are generally created by evaluating the full scale model at some set of training points and fitting the surrogate model to these evaluations.

Most surrogate modelling methods permit an almost free choice of training points, but the choice of training points can strongly affect the quality of the approximation. The training points should span the space of likely input values. Various schemes are available for choosing these values, amongst the most commonly used being full factorial and fractional factorial designs (as mentioned in section 5.2) and various optimized forms of Latin hypercube design (as mentioned in section 5.3). Values need not be chosen randomly (Hammersley sampling is a popular deterministic method), and (as with sampling methods, see section 5.3) the training points should be most dense in regions where the output quantities are sensitive to the values of the input quantities.

Response surface methodology (RSM) is one of the most popular forms of surrogate model, possibly because the methodology is easy to implement. The approach approximates the full function as a sum of polynomial terms. It is common to use at most a second-order polynomial so that the surrogate G(**X**; β) is given by

$$G(\mathbf{X};\beta) = \beta_0 + \sum_{i=1}^{N} \beta_i X_i + \sum_{i=1}^{N} \sum_{j \geq i}^{N} \beta_{ij} X_i X_j$$

where β are a set of parameters to be determined by minimising the sum of squares of the differences between the surrogate model results and the full model evaluated at the training points.

RSM is a well-established method and its implementation has been used for many applications. The evaluation of parameters is generally quick once the training values have been obtained. RSM is best suited for fewer than 10 input random variables and is able to tackle weakly nonlinear problems. RSM is less efficient for strongly nonlinear or discontinuous problems. In principle, higher order polynomials can be adopted, but it needs a large number of training points to accurately capture the function behaviour, reducing the benefits of using RSM. Alternatively the input quantity space could be divided into subsets and a local response surface could be fitted over each subset.

Kriging was developed as a geostatistical estimator for single realizations of random fields that infers the value of the field at unobserved locations from samples. It is a specific application of a broader class of surrogate models known as Gaussian process emulators.

Gaussian process emulation in its simplest form is optimal interpolation based on regression against values of the surrounding training points. The predicted value of the emulator for a new set of input variable values is a weighted sum of the values of the training points. The weights in the sum determining the mean and variance of the function at a new set of input variable values are determined by a covariance function that depends on the distances between the new set of variable values and the training input values. The full mathematical details of the method will not be given here; please see section 5.4 of [11] or [16] for a more complete discussion.

One of the main benefits of the method is that it produces not only a smooth interpolating function for the model results at the training points, but also an uncertainty estimate at each point where the surrogate model is evaluated. This estimate allows the user to identify regions where further model evaluations would lead to an improved surrogate. An example is shown in figure 6, which shows the true curve and the training points in red, the evaluations of the surrogate model in black, and the bounds of a 95% confidence interval in blue.

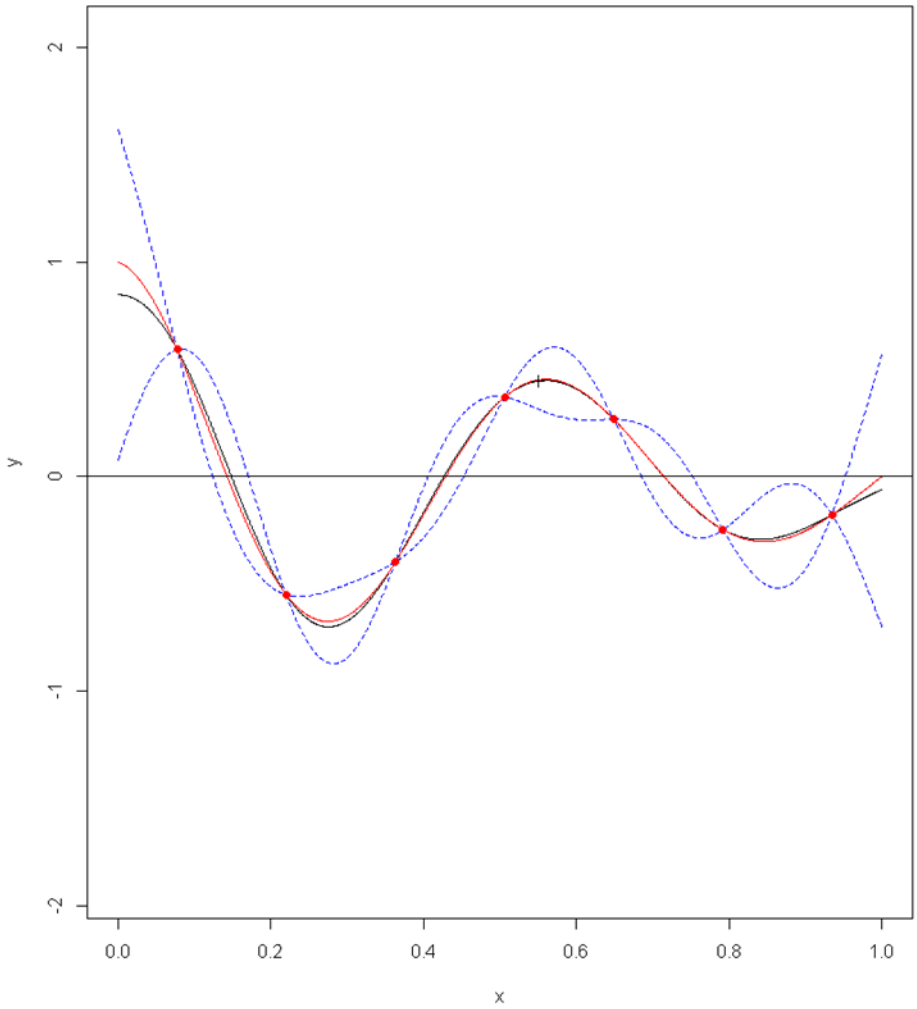

Figure 6: Output of a Gaussian process surrogate model, including confidence interval.

Gaussian process emulation is an extremely flexible method that imposes minimal assumptions on behaviour. Further flexibility is given by the wide range of covariance functions available. The functions can be tailored to fit particular features of a training data set. The technique can be applied to problems with many input quantities (examples with up to 50 input quantities appear in the literature). The performance is good and comparable to second order polynomial regression. The main drawback of the method is that the mathematics involved is quite demanding in places, and in particular the methods for determining best fit parameters can be challenging.

Polynomial chaos is a surrogate model that captures the statistical behaviour of its outputs by constructing a weighted sum of polynomials in its uncertain inputs. The weights are defined as integrals over the space of input quantities, with the integrands being a product of the model function and one of the polynomials. The integrals are evaluated using numerical quadrature, requiring model evaluations. This approach has very rapid convergence properties (in many cases, low order polynomials give a good approximation of the model output), meaning that the methods can outperform most sampling methods. The mean and the variance of the distribution of the output quantity can be obtained directly from the weights of the polynomial and hence are very cheap to compute. For more detail on the background theory, see section 4.5 of [11].

The method is restricted to models with independent input quantities. In some cases involving correlation, the correlation can be eliminated because it is caused by a well-understood physical effect, and the model should treat this effect as an independent random variable. For instance, many material properties are correlated due to their dependence on temperature. If the temperature dependence is

characterised and temperature is treated as a random input quantity, the correlation between input quantities can be removed.

One drawback of the method is the curse of dimensionality, meaning that the number of model evaluations required increases rapidly as the number of input quantities increases. Additionally, convergence becomes less good for cases when there is not a direct link between the distribution of interest and a family of polynomials. These cases are typically addressed by identifying the most similar distribution that does have an associated family of polynomials, and using that family in the expansion.

6 RECOMMENDATIONS FOR THE APPLICATION TO THE MEASUREMENT OF BRAIN PERFUSION USING A CFD-MRI SIMULATION

Figure 1 shows the flow chart of the CFD-MRI simulation that GSP have created. In order to evaluate the measurement uncertainty associated with the estimate of cerebral blood flow (CBF), the first step is to identify the measurement model. It is clear that this problem can be seen as a multi-stage measurement model (as described in Section 4.4) and therefore needs to be broken up into sub-models. The flow chart in Figure 1 show that these sub-models are: 1) geometry; 2) CFD model; 3) particle tracking; 4) tracer kinetic modelling for image generation; and 5) general kinetic modelling for CBF calculation. It is important to identify the measurands for each sub-model, and the input and output quantities for each stage. As stated in Section 4, in the first stage of a multi-stage measurement model, it is necessary to provide information about all the input quantities. In subsequent stages, the input quantities constitute some or all of the output quantities from previous stages plus, possibly, further input quantities.

Given the computationally expensive nature of this problem, it is highly probable that a Monte Carlo calculation would be unfeasible for the multi-stage measurement model, and therefore it is recommended that the advice given in Section 5 is followed. The identification of the numbers of input and outputs quantities will be helpful to carry out the relevant sensitivity analysis and screening processes, potentially leading to a redefinition of the model (following Section 5.1), as well as to help infer the choice of model (as described in Section 5.2) and then using a sampling method (as those described in Section 5.3) or a surrogate model (described in Section 5.4). The Latin hypercube approach is generally robust and comparatively easy to implement, and may therefore provide a good place to start. A surrogate modelling method (that uses the same Latin hypercube sample) may be also worth considering for comparison.

NPL will be providing GSP with further support on the construction of their uncertainty budget through a second A4I project running from August 2018 to June 2019.

ACKNOWLEDGMENTS

This work has been funded through the Analysis for Innovators programme from Innovate UK. Thanks to Louise Wright and Peter Harris for their contributions to this report. Thanks to Trevor Esward for his extensive contributions to the formulation and delivery of this project.

BIBLIOGRAPHY

[1] https://www.goldstandardphantoms.com/innovate-uk-analysis-innovators. [Online]. [Accessed August 2018].

[2] Joint Committee for Guides in Metrology, “JCGM 100: Evaluation of Measurement Data - Guide to the Expression of Uncertainty in Measurement,” 2008.

[3] Joint Committee for Guides in Metrology, “JCGM 101: Evaluation of measurement data – Supplement 1 to the Guide to the expression of uncertainty in measurement - Propagation of distributions using a Monte Carlo method,” 2008.

[4] Joint Committee for Guides in Metrology, “JCGM 102: Evaluation of measurement data – Supplement 2 to the Guide to the expression of uncertainty in measurement - Extension to any number of output quantities,” 2011.

[5] Joint Committee for Guides in Metrology, “JCGM 200: International Vocabulary of Metrology - Basic and General Concepts and Associated Terms,” 2012.

[6] https://www.bipm.org/en/publications/guides/. [Online]. [Accessed August 2018].

[7] M. G. Cox and P. M. Harris, “SSfM Software Support for Metrology, Best Practice Guide No. 6, Uncertainty Evaluation,” NPL, 2010.

[8] M. G. Cox, P. M. Harris and I. M. Smith, “Software specifications for uncertainty evaluation,” NPL, 2010.

[9] http://www.npl.co.uk/science-technology/mathematics-modelling-and-simulation/mathematics-and-modelling-for-metrology/software-for-measurement-uncertainty-evaluation. [Online]. [Accessed August 2018].

[10] M. G. Cox and B. R. L. Siebert, “The use of a Monte Carlo method for evaluating uncertainty and expanded uncertainty,” *Metrologia*, p. s178, 2006.

[11] FORCE (DK) and LNE (Fr) and NPL (UK) and PTB (DE) , “Best practice guide to uncertainty evaluation for computationally expensive models,” EURAMET, 2015.

[12] http://www.itl.nist.gov/div898/handbook/pri/section3/pri3341.htm, March 2015. [Online].

[13] A. Saltelli, K. Chan and E. M. Scott (eds.), Sensitivity Analysis, first edition, John Wiley & Sons, ISBN 0-471-99892-3, 2001.

[14] 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. 21, no. 2, 1979.

[15] R. L. Iman and W. J. A. Conover, “A distribution-free approach to inducing rank correlation among input variables,” *Communications in Statistics - Simulation and Computation*, vol. 11, no. 3, pp. 311-334, 1982.

[16] C. E. Rasmussen and C. K. I. Williams, Gaussian process for machine learning, Massachussetts: MIT press, Cambridge, 2006.