NPL
National Physical Laboratory

REPORT

NPL Report MS 7

Software specifications for uncertainty evaluation

M G Cox, P M Harris and I M Smith

March 2010

National Physical Laboratory, Hampton Road, Teddington, Middlesex, United Kingdom, TW11 0LW
Switchboard 020 8977 3222 NPL Helpline 020 8943 6880 Fax 020 8614 0446 E-mail enquiry@npl.co.uk www.npl.co.uk

NPL Report MS 7

Software specifications for uncertainty evaluation

M G Cox, P M Harris and I M Smith
Mathematics and Scientific Computing Group

March 2010

ABSTRACT

This document provides specifications of software units for the evaluation of measurement uncertainty. It is intended to align with and support established guides and extend their functionality in a consistent manner. It is also intended to complement the best-practice guide to uncertainty evaluation that has been produced as part of the National Measurement System *Software Support for Metrology* (SS*f*M) programme supported by the UK's Department for Business, Innovation and Skills. The target audience is those who in their work wish to use software to assist in the evaluation of uncertainty.

This document is a revised edition of previous reports, accounting for revision of the SS*f*M best-practice guide to uncertainty evaluation and the preparation of Supplements to the 'Guide to the expression of uncertainty in measurement' (GUM).

NPL Report MS 7

ISSN 1754–2960

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

We gratefully acknowledge the financial support of the UK Department for Business, Innovation and Skills (National Measurement Office)

Approved on behalf of the Managing Director, NPL by Jonathan Williams,
Knowledge Leader for the Optical Technologies and Scientific Computing Division

Contents

List of Tables

List of Figures

1 Introduction

The purpose of this document is to provide specifications for software that is relevant to uncertainty evaluation and associated statistical analyses. The specifications relate to software *units* that are useful in this area, rather than packages or systems for uncertainty evaluation. The user or supplier that is implementing software for uncertainty evaluation would need to consider software units of the types covered here. The document is intended to align with and support established guides [3, 4, 5, 18, 49] and extend their functionality in a consistent manner. It is also intended to complement the best-practice guide [14] to uncertainty evaluation that has been produced as part of the National Measurement System *Software Support for Metrology* (SS*f*M) programme supported by the UK's Department for Business, Innovation and Skills. The target audience is those who in their work wish to use software to assist in the evaluation of uncertainty. This document is a revised edition of previous reports [11, 12, 13], accounting for revision of the SS*f*M best-practice guide to uncertainty evaluation and the preparation of Supplements [2] to the 'Guide to the expression of uncertainty in measurement' (GUM) [3].

The software specifications are not intended to be mandatory. They typify constituent parts of uncertainty evaluation. In particular, they indicate the input and output parameters of the software units and the purpose of each unit, viz., a statement of the computational aim of the unit. The units are presented in the context of the more complete calculations in which they would be used.

The specifications generally have a minimal number of input and output parameters. Computational control parameters, such as those relating to the convergence criteria of iterative techniques, are indicated only broadly. Diagnostic parameters, such as those that indicate failure or degree of success of the computation, are not included.

The scope of the specifications is those that relate to:

1. The *GUM uncertainty framework* as summarized in Clause 8 of the 'Guide to the expression of uncertainty in measurement' (GUM) [3];[1]
2. More general calculations, consistent with the GUM, for the *propagation of distributions* based on the use of a *Monte Carlo method* [4, 14];
3. The validation of (the use of) the GUM uncertainty framework using a Monte Carlo method [4, 14].

It is emphasized that the primary concern is the *specification* of relevant software. Algorithms and software *per se* are not the main consideration: the provision of such material

[1] In this document the term 'GUM uncertainty framework' is used to describe the procedure that includes (a) the application of the law of propagation of uncertainty in terms of a first-order approximation to the measurement model to obtain an estimate of the output quantity and the associated standard uncertainty, and (b) the assumptions of the central limit theorem to obtain a coverage interval.

is the responsibility of the user or supplier that is implementing software for uncertainty evaluation. However, two departures from this stance are made:

1. Because of the importance of being able to generate random numbers from a rectangular distribution that (a) have sound statistical properties in their own right, (b) can be used as a basis for the many generators that make use of a rectangular random number generator for their function, and (c) in certain circumstances can be regenerated exactly, a specific rectangular random number generator is recommended (Section 5.3). In addition, generators for Gaussian, t–, multivariate Gaussian and multivariate t–distributions are regarded as so fundamental that algorithmic statements for them are included (also in Section 5.3).

2. Some numerical considerations accompany the specifications, especially related to the numerical stability of the underlying algorithms. Such considerations are regarded as an important adjunct to the specification: users need to be able to rely on software written in accordance with the specifications to perform reliably in a numerical way as well as in a functional manner.

The fundamental relationship between the input quantities and the output quantity is the *measurement model*. If there is a single (scalar) output quantity Y, the measurement model takes the general form

$$h(Y, \boldsymbol{X}) = h(Y, X_1, \ldots, X_N) = 0,$$

in which $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$ denotes the N input quantities. There may be more than one output quantity, viz., $\boldsymbol{Y} = (Y_1, \ldots, Y_m)^\top$. In this case the measurement model is

$$\boldsymbol{h}(\boldsymbol{Y}, \boldsymbol{X}) = \boldsymbol{0},$$

where $\boldsymbol{h}(\boldsymbol{Y}, \boldsymbol{X}) = (h_1(\boldsymbol{Y}, \boldsymbol{X}), \ldots, h_m(\boldsymbol{Y}, \boldsymbol{X}))^\top$, a vector of measurement models.

If the measurement model can be expressed in the equivalent form

$$Y = f(\boldsymbol{X}) = f(X_1, \ldots, X_N),$$

for a scalar output quantity, or

$$\boldsymbol{Y} = \boldsymbol{f}(\boldsymbol{X}) = \boldsymbol{f}(X_1, \ldots, X_N),$$

for a vector output quantity, the function f or $\boldsymbol{f}$ is the *measurement function*. The measurement function can be a mathematical formula, a step-by-step calculation procedure, computer software or other prescription, which yields a (unique) value y or $\boldsymbol{y}$ of the output quantity given values $\boldsymbol{x} = (x_1, \ldots, x_N)^\top$ of the input quantities.

The specifications provided relate to a classification of measurement model types, there being $2 \times 2 \times 2 = 8$ model types in all, based on whether

- there is one or more than one output quantity,

- it is possible or not to write the measurement model as a measurement function, and
- the quantities within the measurement model are real or complex.

The coverage of the software specifications is divided according to whether they relate to

- the *formulation* or *calculation* stages of the process of uncertainty evaluation (Section 1.1),
- the procedure followed is in accordance with the GUM uncertainty framework [3, Clause 8], a Monte Carlo method [4, 14] or the validation of the GUM uncertainty framework using a Monte Carlo method [4, 14], and
- the concern is with measurement models with a univariate (scalar) or a multivariate (vector) output quantity.

Where a vector of values, e.g., the estimates $\boldsymbol{x} = (x_1, \ldots, x_N)^\top$ of the N input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$ is needed, it is used in the specifications here as a *column vector* or a *row vector*, as appropriate, for consistency with calculations expressed in linear algebra terms [24]. It is not necessary that the physical means of data storage adopted in software implementations accords with these representations.

The bulk of this document is concerned with coverage intervals corresponding to a coverage probability of $95\,\%$ and is couched this way. It is generally extensible to other coverage probabilities.

The report is organized as follows. In the remainder of this section the above stages of uncertainty evaluation are discussed. In Section 2 the formulation stage and its outputs are described in terms of the information required for implementing the GUM uncertainty framework and a Monte Carlo method. Section 3 covers the numerical evaluation of the measurement model according to the eight types indicated earlier. In Sections 4, 5 and 6 specifications of software units are provided for the three aspects of uncertainty evaluation considered: the GUM uncertainty framework, a Monte Carlo method and the validation of the GUM uncertainty framework using a Monte Carlo method. Conclusions are given in Section 7.

1.1 The stages of uncertainty evaluation

The software specifications are divided according to whether they relate to the *formulation* or *calculation* stages of the process of uncertainty evaluation.

In the formulation stage a measurement model is derived and the input quantities in the model are quantified. Each input quantity is characterized by a probability distribution in the form of a probability density function (PDF) or distribution function (DF). According

to the GUM, distributions are obtained from an analysis of a series of indication values [3, Clauses 2.3.2, 3.3.5] or are based on scientific judgement using all the relevant information available [3, Clauses 2.3.3, 3.3.5]. A distribution may also be defined by a previous uncertainty evaluation as part of a multi-stage uncertainty evaluation process [14]. The distribution may be available analytically in a recognized form (rectangular, Gaussian, etc.) or as an approximation obtained from a previous application of a Monte Carlo method, for example. A means for determining such an approximation is given in Section 5.4.4.

In the calculation stage the distributions are propagated through the measurement model to obtain the distribution for the output quantity. This distribution is used to obtain the expectation of the output quantity, taken as an estimate of the output quantity, the standard deviation of the quantity, taken as the standard uncertainty associated with the estimate, and a coverage interval corresponding to a specified coverage probability. The GUM uncertainty framework and a Monte Carlo method both provide approaches to undertaking the calculation stage of the process of uncertainty evaluation.

For an input quantity that is independent of the other input quantities, the GUM uncertainty framework requires for its operation only three parameters that summarize the information about the quantity represented by its distribution:

- An estimate of the input quantity;
- The standard uncertainty associated with the estimate;
- The corresponding degrees of freedom.

If there are dependencies, the procedure will also require the covariances associated with estimates of the mutually dependent quantities.[2]

Software can support the formulation stage by providing estimates of location and dispersion and degrees of freedom from which a probability distribution can be constructed after making appropriate assumptions. In the case of a sufficiently large number of repeated indication values of a set of the input quantities, the data can be used to define a covariance matrix.[3] The specifications below relate to the (arithmetic) mean as a measure of location, and its associated standard deviation as a measure of dispersion. In addition, the determination of a covariance matrix is covered.

The output of the formulation stage is required to be in a form to act as inputs either to the GUM uncertainty framework or to a Monte Carlo procedure. For the GUM uncertainty framework, the inputs are the outputs of the formulation stage identified above. For a Monte

[2] The degree of mutual dependence associated with the estimates x_i and x_j of the input quantities X_i and X_j is sometimes characterized by the *correlation coefficient*. The covariance $\mathrm{cov}(x_i, x_j)$ and correlation coefficient $r(x_i, x_j)$ are related by $\mathrm{cov}(x_i, x_j) = r(x_i, x_j)u(x_i)u(x_j)$, where $u(x_i)$ and $u(x_j)$ are, respectively, the standard uncertainties associated with the estimates x_i and x_j.

[3] A covariance matrix is also known as a variance-covariance or uncertainty matrix [4].

Carlo method, the inputs are probability distributions or, where appropriate, joint distributions (where there are mutual dependencies) for the input quantities.

The basis of the GUM uncertainty framework is to 'propagate the uncertainties' associated with estimates of the input quantities through the measurement model to provide the uncertainties associated with estimates of the output quantities. For this purpose it is required to:

- Evaluate the measurement model at the estimates of the input quantities to obtain estimates of the output quantities;
- Determine sensitivity coefficients;
- Propagate the covariance matrix associated with the estimates of the input quantities through (a linearization of) the measurement model to yield the covariance matrix associated with estimates of the output quantities;
- Apply the Welch-Satterthwaite formula for effective degrees of freedom;
- Determine percentage points of a Gaussian or a t–distribution in order to provide a coverage interval for the output quantity.

These calculations would apply, as appropriate, for each of the measurement model types, and software can assist with the calculations.

For a Monte Carlo procedure it is necessary to 'propagate (discrete representations of) the distributions' for the input quantities through the measurement model. For this purpose it is required to:

- Obtain values for the input quantities by making random draws from the probability distributions used to characterize the quantities;
- Evaluate the measurement model for the values for the input quantities to obtain a set of values for the output quantities;
- Use the set of values for the output quantities to form a discrete representation of the distribution for the output quantities;
- Determine from the discrete representation estimates of the output quantities and the associated covariance matrix;
- Determine from the discrete representation a coverage interval for the output quantity.

Again, these calculations would apply, as appropriate, for each of the measurement model types, and software can assist with their implementation.

The remaining software specifications are intended to support a recommended procedure for validating the GUM uncertainty framework using a Monte Carlo method [4, 14]. This procedure constitutes operating both the GUM uncertainty framework and a Monte Carlo method for an uncertainty evaluation of concern, and carrying out an appropriate comparison of the results obtained.

1.2 Document history

This document is intended to align with and help support established guides [3, 18, 48, 49] on uncertainty evaluation and extend their functionality in a consistent manner. It is complementary to the best-practice guide [14] on uncertainty evaluation, produced as part of the UK's *Software Support for Metrology* (SS*f*M) programme. The first edition of this best-practice guide was published in March 2001, having been developed during the first SS*f*M programme covering the period April 1998 to March 2001. During that period Working Group 1, 'Expression of Uncertainty in Measurement', of the Joint Committee for Guides in Metrology (JCGM) started work, following its first meeting in March 2000, on the first Supplement [4] to the 'Guide to the expression of uncertainty in measurement' (GUM) [3] concerned with numerical methods for the propagation of distributions [2]. Material from the evolving best-practice guide was used in various parts of the Supplement and subsequently refined appropriately for consistency with the published Supplement.

This revised report, produced during the fourth SS*f*M programme, April 2007 to March 2010, takes account of the revision of the best-practice guide (now in its fourth edition) and the preparation of the Supplement. In particular, material from the drafts of the Supplement prepared during the second and third programmes that had an origin in the first edition of the best-practice guide has been re-used. This revised report also takes account of the work of Working Group 1 of the JCGM to prepare Supplement 2 to the GUM on using the GUM uncertainty framework and a Monte Carlo method as approaches to uncertainty evaluation for measurement models with a general number of output quantities. At the time of publishing this report, a draft of Supplement 2 is available to the member organizations of the JCGM for review by those organizations.

2 Formulation

In the formulation stage of uncertainty evaluation, it is necessary to characterize the input quantities in the measurement model by probability density functions (PDFs). In the GUM uncertainty framework only the expectations and standard deviations of the quantities characterized by these PDFs, and covariances where appropriate, are used. For a Monte Carlo method the PDFs themselves are used. The PDFs that characterize the input quantities depend on the information that is available about the quantities. Two types of information are considered.

If a set of q indication values is available, obtained independently, the average of these values is regarded as an instance of a quantity X_i with unknown expectation and standard deviation. A statistical analysis of these values is undertaken to determine an estimate x_i of X_i together with the associated standard uncertainty $u_i = u(x_i)$ and corresponding degrees of freedom ν_i. These are the parameters used in the GUM uncertainty framework (Section 4). To apply a Monte Carlo method the parameters are used to characterize X_i by a PDF. The analysis of repeated indication values to characterize an input quantity by a PDF is described in Section 2.1. In the GUM the use of this type of information is referred to as 'a Type A evaluation of uncertainty'.

If repeated indication values are not available, the input quantity X_i is characterized by a PDF that is constructed on the basis of any relevant information concerning the quantity. Such information may be, for example, prior knowledge, experience or the result of a previous uncertainty evaluation. The PDF is used directly in a Monte Carlo method (Section 5). To apply the GUM uncertainty framework parameters x_i, u_i and ν_i are determined from the PDF. Results of the determination for a number of common PDFs are given in Section 2.2. In the GUM the use of this type of information is referred to as 'a Type B evaluation of uncertainty'.

A variant of these analyses applies if some or all of the input quantities are mutually dependent. For the GUM uncertainty framework a covariance matrix associated with the estimates of the relevant input quantities is evaluated. For a Monte Carlo method these input quantities would be characterized by a joint (multivariate) PDF.

In practice both types of information will be available, each applying to a subset of the input quantities. For the GUM uncertainty framework a covariance matrix would be constructed from those covariance matrices arising from the two types of analysis. For a Monte Carlo method a joint PDF would be constructed from those PDFs arising from the two types of analysis.

2.1 Formulation based on analysing repeated indication values

The statistical analysis of repeated indication values can be approached in two parts.

1. Given a set of indication values, obtained independently, of a quantity with unknown expectation X_i, determine an estimate x_i of X_i together with the standard uncertainty u_i associated with that estimate and the corresponding degrees of freedom ν_i (Section 2.1.1).

2. Given a set of indication values of a pair of quantities with unknown expectations X_i and X_j, determine the covariance associated with estimates x_i and x_j of X_i and X_j (Section 2.1.2).

These calculations are the basis for determining a covariance matrix associated with the estimates of a number of input quantities (Section 2.1.3).

2.1.1 Mean and its associated standard deviation

Suppose $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})$ are q indication values, obtained independently, of a quantity with unknown expectation X_i. The estimate x_i of X_i and the associated standard uncertainty u_i are determined, respectively, as $\bar{x}_i$, the arithmetic mean of the indication values, and s_i, the standard deviation associated with $\bar{x}_i$. The corresponding degrees of freedom ν_i is $q - 1$. Table 1 defines the mean of a set of indication values, the standard deviation associated with this mean and the degrees of freedom, and specifies the input and output parameters associated with their determination. This is the information required for the GUM uncertainty framework (Section 4).

Given the estimate $\bar{x}_i$ and the associated standard uncertainty s_i so obtained, X_i is characterized by a PDF as follows [14]:

- If the distribution underlying the indication values is unknown, characterize X by the Gaussian distribution $\mathrm{N}(\bar{x}_i, s_i^2)$;
- If the distribution underlying the indication values is known to be Gaussian, characterize X by the t–distribution $t_{\nu_i}(\bar{x}_i, s_i^2)$ with $\nu_i = q - 1$ degrees of freedom.

This is the information required for a Monte Carlo method (Section 5). Information on how to generate pseudo-random numbers from these distributions for the purpose of a Monte Carlo method is given in Sections 5.3.2 and 5.3.3, respectively.

2.1.2 Covariance associated with two means

Suppose $(x_{i,k}, x_{j,k})^\top$, $k = 1, \ldots, q$, are q pairs of indication values, each pair obtained independently of the remaining pairs, of quantities with unknown expectations X_i and X_j. An estimate x_i of X_i, together with the associated standard uncertainty u_i, may be determined as the mean $\bar{x}_i$ and the standard deviation s_i associated with the mean for the set of indication values $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})$ as in Section 2.1.1, and similarly for X_j. The covariance of X_i and X_j is taken as the covariance $u_{i,j} = u(\bar{x}_i, \bar{x}_j)$ associated with the means $\bar{x}_i$ and $\bar{x}_j$. Table 2 defines this covariance in terms of repeated indication values, and specifies the input and output parameters associated with its determination.

2.1.3 Covariance matrix for input quantities

Suppose $(x_{1,k}, x_{2,k}, \ldots, x_{N,k})^\top$, $k = 1, \ldots, q$, are q N-tuples of indication values of quantities with unknown expectations X_i, $i = 1, \ldots, N$, each N-tuple obtained independently

Input parameters	
q	Number of indication values
$\boldsymbol{x}_i$	Indication values $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})$, obtained independently
Output parameters	
$\bar{x}_i$	Mean, defined by $$\bar{x}_i = \frac{1}{q}\sum_{k=1}^{q} x_{i,k}$$
s_i	Standard deviation associated with the mean, defined by $$s_i^2 = \frac{1}{q(q-1)}\sum_{k=1}^{q}(x_{i,k} - \bar{x}_i)^2$$
ν_i	Degrees of freedom, defined by $$\nu_i = q - 1$$
Numerical analysis	
	The above formula for s_i should be used rather than the mathematically equivalent formula $$s_i^2 = \frac{1}{q-1}\left(\frac{1}{q}\sum_{k=1}^{q} x_{i,k}^2 - \bar{x}_i^2\right).$$ For cases in which s_i is very much smaller than $\lvert\bar{x}_i\rvert$ (in which case the $x_{i,k},\ k = 1, \ldots, q$, have a number of leading digits in common) the latter formula suffers from subtractive cancellation (involving a mean square less a squared mean). The cancellation effects can be so severe that the resulting value of s_i may have too few correct significant figures for the uncertainty evaluation to be valid [10]

Table 1: Mean, associated standard deviation and corresponding degrees of freedom from repeated indication values.

Input parameters	
q	Number of indication values
$\boldsymbol{x}_i$	Indication values $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})$
$\boldsymbol{x}_j$	Indication values $(x_{j,1}, x_{j,2}, \ldots, x_{j,q})$ paired with those in $\boldsymbol{x}_i$. Each pair $(x_{i,k}, x_{j,k})^\top$ is obtained independently of the remaining pairs
Output parameter	
$u_{i,j}$	Covariance associated with the estimates for the ith and jth input quantities, defined by $$u_{i,j} = \frac{1}{q(q-1)} \sum_{k=1}^{q} (x_{i,k} - \bar{x}_i)(x_{j,k} - \bar{x}_j),$$ where $\bar{x}_i$ and $\bar{x}_j$ are the means of the indication values $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, respectively
Numerical analysis	
	The above formula for $u(x_i, x_j)$ should be used rather than the mathematically equivalent formula $$u_{i,j} = \frac{1}{q-1} \left(\frac{1}{q} \sum_{k=1}^{q} x_{i,k} x_{j,k} - \bar{x}_i \bar{x}_j \right).$$ The latter formula can suffer from subtractive cancellation as in the standard deviation calculation (Table 1) and the resulting value of $u(x_i, x_j)$ may have too few correct figures for the uncertainty evaluation to be valid

Table 2: Covariance associated with two means from repeated indication values.

of the remaining N-tuples, and that these are assembled into the $N \times q$ matrix $\boldsymbol{\Phi}$.[4] Let $\boldsymbol{\Phi}'$ be obtained from $\boldsymbol{\Phi}$ by correcting for the means, i.e., the mean $\bar{x}_i$ of the elements in the ith row is subtracted from all elements $x_{i,j},\ j = 1, \ldots, q$, in that row. Then, the covariance matrix $\boldsymbol{U}_{\boldsymbol{x}}$ of covariances $u_{i,j}$ associated with the means $\bar{\boldsymbol{x}} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_N)^\top$ is given by

$$\boldsymbol{U}_{\boldsymbol{x}} = \frac{1}{q(q-1)} \boldsymbol{\Phi}'(\boldsymbol{\Phi}')^\top.$$

Table 3 defines the covariance matrix associated with the means in terms of repeated indication values, and specifies the input and output parameters associated with its determination. The estimates $\boldsymbol{x} = \bar{\boldsymbol{x}}$ for the input quantities $\boldsymbol{X}$ and the associated covariance matrix $\boldsymbol{U}_{\boldsymbol{x}}$ is the information required for the GUM uncertainty framework (Section 4).

Input parameters	
N	Number of input quantities
q	Number of indication values
$\boldsymbol{\Phi}$	$N \times q$ matrix containing the indication values $x_{i,j}$, where $x_{i,j}$ is the jth indication value corresponding to the ith input quantity. Each N-tuple $(x_{1,k}, x_{2,k}, \ldots, x_{N,k})^\top$ is obtained independently of the remaining N-tuples
Output parameter	
$\boldsymbol{U}_{\boldsymbol{x}}$	Covariance matrix of dimension $N \times N$ associated with the estimates of the input quantities $\boldsymbol{X}$, defined by $\frac{1}{q(q-1)} \boldsymbol{\Phi}'(\boldsymbol{\Phi}')^\top$, where $\boldsymbol{\Phi}'$ is $\boldsymbol{\Phi}$ corrected for the means

Table 3: Covariance matrix associated with the means from repeated indication values.

Given estimates $\bar{\boldsymbol{x}}$ and the associated covariance matrix $\boldsymbol{U}_{\boldsymbol{x}}$, $\boldsymbol{X}$ is characterized by a joint PDF as follows:

- If the distribution underlying the indication values is unknown, characterize $\boldsymbol{X}$ by the multivariate Gaussian distribution $\mathrm{N}(\bar{\boldsymbol{x}}, \boldsymbol{U}_{\boldsymbol{x}})$;
- If the distribution underlying the indication values is known to be Gaussian, characterize $\boldsymbol{X}$ by the multivariate t–distribution $t_\nu(\bar{\boldsymbol{x}}, \boldsymbol{S}_{\boldsymbol{x}})$ with $\nu = q - N$ degrees of freedom,[5] where

$$\boldsymbol{S}_{\boldsymbol{x}} = \left(\frac{q-1}{q-N}\right) \boldsymbol{U}_{\boldsymbol{x}}.$$

[4] The symbol $\boldsymbol{\Phi}$ is (reluctantly) used to denote the matrix of indication values $x_{i,j}$, since X is used to denote a scalar input quantity and $\boldsymbol{X}$ a vector input quantity.

[5] q must be strictly greater than N for this characterization to apply.

This is the information required for a Monte Carlo method (Section 5). Information on how to generate pseudo-random numbers from these distributions for the purpose of a Monte Carlo method is given in Sections 5.3.7 and 5.3.8, respectively.

In practice there may be 'simultaneous' indication values $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})$ for *some* of the X_i, e.g, for $i = 2$, 4 and 5. In this case the above covariance considerations would apply to this 'group'. Any other such groups would be handled similarly, and the complete covariance matrix of dimension $N \times N$ constructed from these groups, together with the variances associated with estimates of those input quantities that are mutually independent.

2.2 Formulation based on other available information

If knowledge of an input quantity is based on non-statistical information, the quantity would be characterized by a PDF that depended on the nature of the information. For a Monte Carlo method this PDF is used directly. The information that leads to the characterization of a quantity by some common distributions is indicated below (Sections 2.2.1 to 2.2.4). Table 4 gives the estimate x_i, the associated standard uncertainty u_i and the corresponding degrees of freedom ν_i used by the GUM uncertainty framework based on the available information.

Distribution		x_i	u_i	ν_i
Rectangular	$\mathrm{R}(a, b)$	$\dfrac{b+a}{2}$	$\dfrac{b-a}{2\sqrt{3}}$	∞
Gaussian	$\mathrm{N}(\mu, \sigma^2)$	μ	σ	∞
Curvilinear trapezoid	$\mathrm{CTrap}(a, b, d)$	$\dfrac{b+a}{2}$	$\dfrac{b-a}{2\sqrt{3}}$	$\dfrac{1}{2}\left[\dfrac{b-a}{2d}\right]^2$
U-shaped	$\mathrm{U}(a, b)$	$\dfrac{b+a}{2}$	$\dfrac{b-a}{2\sqrt{2}}$	∞

Table 4: Estimate, associated standard uncertainty and corresponding degrees of freedom for an input quantity characterized by some common probability distributions.

For example, suppose the only available information regarding an input quantity X_i is a

lower limit a and an upper limit b with $a < b$. Then, X_i is characterized by a rectangular distribution with limits a and b (Section 2.2.1). Information on how to generate pseudo-random numbers from this distribution for the purpose of a Monte Carlo method is given in Section 5.3.1. For this distribution

$$x_i = \frac{b+a}{2}, \quad u_i = \frac{b-a}{2\sqrt{3}}, \quad \nu_i = \infty,$$

and this is the information required for the GUM uncertainty framework (Table 4). For a quantity X_i characterized by a different PDF, Table 4 would be used in a similar way.

2.2.1 Rectangular distribution

If the only available information regarding a quantity X is a lower limit a and an upper limit b with $a < b$, then, according to the principle of maximum entropy, X would be characterized by the rectangular distribution $\mathrm{R}(a, b)$ over the interval $[a, b]$.

The PDF for X is

$$g_X(\xi) = \begin{cases} 0, & \xi < a, \\ 1/(b-a), & a \le \xi \le b, \\ 0, & b < \xi. \end{cases}$$

Information on how to generate pseudo-random numbers from this distribution for the purpose of a Monte Carlo method is given in Section 5.3.1.

2.2.2 Gaussian distribution

If a best estimate μ and associated standard uncertainty σ are the only information regarding a quantity X, then, according to the principle of maximum entropy, X would be characterized by the Gaussian probability distribution $\mathrm{N}(\mu, \sigma^2)$.

The PDF for X is

$$g_X(\xi) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(\xi-\mu)^2}{2\sigma^2}\right].$$

Information on how to generate pseudo-random numbers from this distribution for the purpose of a Monte Carlo method is given in Section 5.3.2.

2.2.3 Curvilinear trapezoid distribution

A quantity X is known to lie between limits A and B with $A < B$, where the midpoint $(A + B)/2$ of the interval defined by these limits is fixed and the length $B - A$ of the interval is not known exactly. A is known to lie in the interval $a \pm d$ and B in $b \pm d$, where

a, b and d, with $d > 0$ and $a + d < b - d$, are specified. If no other information is available concerning X, A and B, the principle of maximum entropy can be applied to characterize X by the curvilinear trapezoid CTrap(a, b, d).

The PDF for X is

$$g_X(\xi) = \frac{1}{4d} \begin{cases} 0, & \xi < a - d, \\ \ln\left[(w + d) / (h - \xi)\right], & a - d \le \xi \le a + d, \\ \ln\left[(w + d) / (w - d)\right], & a + d < \xi < b - d, \\ \ln\left[(w + d) / (\xi - h)\right], & b - d \le \xi \le b + d, \\ 0, & b + d < \xi, \end{cases}$$

where $w = (b - a)/2$ and $h = (a + b)/2$. Information on how to generate pseudo-random numbers from this distribution for the purpose of a Monte Carlo method is given in Section 5.3.4.

2.2.4 U-shaped distribution

If a quantity X is known to cycle sinusoidally, with unknown phase Φ, between specified limits a and b, with $a < b$, then, according to the principle of manimum entropy, Φ would be characterized by the rectangular distribution $\mathrm{R}(0, 2\pi)$, and X by the U-shaped distribution $U(a, b)$.

The PDF for X is

$$g_X(\xi) = \begin{cases} 0, & \xi < a, \\ 1/[\pi\sqrt{w^2 - (\xi - h)^2}], & a \le \xi \le b, \\ 0, & b < \xi. \end{cases}$$

where $w = (b - a)/2$ and $h = (a + b)/2$. Information on how to generate pseudo-random numbers from this distribution for the purpose of a Monte Carlo method is given in Section 5.3.5.

3 Calculation of a value of the output quantity

The classification of measurement models in Section 1 covers eight types of model. In principle, any measurement model will naturally fall into one, and only one, of these categories. Some measurement models can be converted from one type to another. Whether doing so is desirable depends on circumstances. For instance, it may not be numerically stable to do so. A complex measurement model can always be converted into a real measurement model, by replacing each complex quantity by two real quantities, its real and imaginary

parts. Again, doing so is not necessarily desirable for purposes of calculating a value of the output quantity.

Consider the calculation of the value y or $\boldsymbol{y}$ for the *measurement functions* $Y = f(\boldsymbol{X})$ (univariate) and $\boldsymbol{Y} = \boldsymbol{f}(\boldsymbol{X})$ (multivariate) given the value $\boldsymbol{x}$ of $\boldsymbol{X}$. This calculation requires the evaluation of the function $f(\boldsymbol{x})$ or $\boldsymbol{f}(\boldsymbol{x})$. Tables 5 and 6 specify the calculations for these types of measurement model, and the input and output parameters required.

Consider the calculation of the value y or $\boldsymbol{y}$ for the *measurement models* $h(Y, \boldsymbol{X}) = 0$ (univariate) and $\boldsymbol{h}(\boldsymbol{Y}, \boldsymbol{X}) = \boldsymbol{0}$ (multivariate) given the value $\boldsymbol{x}$ of $\boldsymbol{X}$. This calculation requires the solution of the equation $h(y, \boldsymbol{x}) = 0$ or system of equations $\boldsymbol{h}(\boldsymbol{y}, \boldsymbol{x}) = \boldsymbol{0}$. For a univariate measurement model, it is necessary to solve a single equation. For a multivariate measurement model, a system of equations is to be solved. Tables 7 and 8 specify the calculations for these types of measurement model, and the input and output parameters required.

The calculation of a value of the output quantity for *complex* measurement models need be no more complicated than for the categories of real measurement models considered above. Many software packages and languages provide a complex type for complex quantities together with functions for performing complex arithmetic. An alternative to using such facilities is to store explicitly each complex quantity in terms of its real and imaginary parts, and to undertake all numerical operations in terms of these two (real) parts.

An issue that requires consideration as part of any implementation for these categories of complex measurement models is the way the '(standard) uncertainty' associated with an estimate of a complex quantity is stored. If X_i is complex with real and imaginary parts X_i^{R} and X_i^{I}, the 'squared (standard) uncertainty' associated with an estimate x_i of X_i is described by the covariance matrix

$$\boldsymbol{U}_i = \begin{bmatrix} \mathrm{cov}(x_i^{\mathrm{R}}) & \mathrm{cov}(x_i^{\mathrm{R}}, x_i^{\mathrm{I}}) \\ \mathrm{cov}(x_i^{\mathrm{R}}, x_i^{\mathrm{I}}) & \mathrm{cov}(x_i^{\mathrm{I}}) \end{bmatrix}$$

of dimension 2×2, where $\mathrm{cov}(x_i^{\mathrm{R}}, x_i^{\mathrm{R}}) = u^2(x_i^{\mathrm{R}})$ and $\mathrm{cov}(x_i^{\mathrm{I}}, x_i^{\mathrm{I}}) = u^2(x_i^{\mathrm{I}})$ are, respectively, the variances associated with the estimates x_i^{R} and x_i^{I} of the real and imaginary parts, and $\mathrm{cov}(x_i^{\mathrm{R}}, x_i^{\mathrm{I}})$ is the covariance associated with these estimates. Furthermore, the covariance matrix $\boldsymbol{U}_{\boldsymbol{x}}$ for the complete set of input quantities $X_i,\ i = 1, \ldots, N$, is a matrix of dimension $2N \times 2N$. Consequently, although the quantities themselves may be regarded as complex, it is necessary to store the corresponding uncertainty information using real matrices, and operate on them using real arithmetic.

Input parameters	
N	Number of input quantities
f	Function specifying the measurement function $Y = f(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
Output parameter	
y	Value obtained by evaluating the function $$y = f(\boldsymbol{x})$$

Table 5: Value of the output quantity in a univariate, real measurement function.

Input parameters	
N	Number of input quantities
m	Number of output quantities
$\boldsymbol{f}$	Function with m components specifying the measurement function $\boldsymbol{Y} = \boldsymbol{f}(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
Output parameter	
$\boldsymbol{y}$	Column vector of values $(y_1, \ldots, y_m)^\top$ obtained by evaluating the function $$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$$

Table 6: Value of the output quantity in a multivariate, real measurement function.

Input parameters	
N	Number of input quantities
h	Function specifying the measurement model $h(Y, \boldsymbol{X}) = 0$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$ and the output quantity Y
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
$\boldsymbol{t}$	Computational control parameters such as lower and upper bounds for y and technical parameters relating to the equation-solving software
Output parameter	
y	Value obtained by solving the equation $h(y, \boldsymbol{x}) = 0$
Numerical analysis	
	A zero-finding algorithm [16, 23], such as the bisection algorithm in cases where suitable lower and upper bounds are known for y, can be used to solve the equation

Table 7: Value of the output quantity in a univariate, real measurement model.

4 GUM uncertainty framework

4.1 Procedure

For the application of the GUM uncertainty framework, the outputs of the formulation stage are (Section 2):

- Estimates $\boldsymbol{x} = (x_1, \ldots, x_N)^\top$ of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$;
- Standard uncertainties $\boldsymbol{u} = (u_1, \ldots, u_N)^\top$ associated with these estimates;
- Corresponding degrees of freedom $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_N)^\top$;
- Where appropriate, covariances associated with estimates of the input quantities that are mutually dependent.

This information is conveniently represented by $\boldsymbol{x}$, $\boldsymbol{U_x}$ and $\boldsymbol{\nu}$, where $\boldsymbol{U_x}$ is a covariance matrix that holds the variances (squared standard uncertainties) associated with the estimates $\boldsymbol{x}$ and the covariances associated with these estimates. $\boldsymbol{U_x}$ is a matrix of dimension $N \times N$, whose (i, j)th element contains the covariance $\mathrm{cov}(x_i, x_j)$ associated with x_i and x_j, with $\mathrm{cov}(x_i, x_i) = u_i^2$.

Input parameters	
N	Number of input quantities
m	Number of output quantities
$\boldsymbol{h}$	Function with m components specifying the measurement model $\boldsymbol{h}(\boldsymbol{Y}, \boldsymbol{X}) = \boldsymbol{0}$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$ and the output quantities $\boldsymbol{Y} = (Y_1, \ldots, Y_m)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
$\boldsymbol{t}$	Computational control parameters such as an initial approximation to $\boldsymbol{y}$ and technical parameters relating to the equation-solving software
Output parameter	
$\boldsymbol{y}$	Column vector of values $(y_1, \ldots, y_m)^\top$ obtained by solving the equations $\boldsymbol{h}(\boldsymbol{y}, \boldsymbol{x}) = \boldsymbol{0}$
Numerical analysis	
	An iterative algorithm such as Newton's method [23], starting from a suitable approximation to $\boldsymbol{y}$, can be used to solve the system of equations

Table 8: Value of the output quantity in a multivariate, real measurement model.

$\boldsymbol{x}$, $\boldsymbol{U}_{\boldsymbol{x}}$ and $\boldsymbol{\nu}$, together with the measurement model and the required coverage probability p (e.g., 0.95), constitute the inputs to the calculation stage of the GUM uncertainty framework.

For a *univariate measurement function*, the procedure is as follows:

1. Calculate the estimate of the output quantity by evaluating the measurement function at the estimates of the input quantities.[6] See Section 3.

2. Form the partial derivatives of first order of the measurement function with respect to the input quantities, and calculate the sensitivity coefficients by evaluating these partial derivatives at the estimates of the input quantities.[7] See Section 4.2.

3. Calculate the standard uncertainty associated with the estimate of the output quantity by combining the standard uncertainties associated with the estimates of the input quantities, the covariances associated with these estimates and the sensitivity coefficients. See Section 4.3.

4. When the input quantities $\boldsymbol{X}$ are mutually independent, use the Welch-Satterthwaite formula to calculate $\nu_{\rm eff}$, the effective degrees of freedom attached to the standard uncertainty, from the standard uncertainties associated with the estimates of the input quantities, the corresponding degrees of freedom, the sensitivity coefficients and the standard uncertainty associated with the estimate of the output quantity. The GUM uncertainty framework does not state how $\nu_{\rm eff}$ is to be calculated when the input quantities are correlated. See Section 4.4.1.

5. Calculate the coverage factor corresponding to $\nu_{\rm eff}$ and the required coverage probability p as a percentage point of the (standard) Gaussian distribution ($\nu_{\rm eff} = \infty$) or a t–distribution ($\nu_{\rm eff} < \infty$). Hence, calculate the expanded uncertainty, and thus an interval containing the output quantity with the stipulated coverage probability, by forming the product of this coverage factor and the standard uncertainty associated with the estimate of the output quantity. See Sections 4.4.2 and 4.4.3.

The computational flow of the calculation stage for the GUM uncertainty framework, indicating the inputs and the outputs, viz., an estimate of the output quantity, the standard uncertainty associated with this estimate and a coverage interval, is given in Figure 1. This figure applies in the case of a univariate, real measurement function with input quantities that are mutually independent. Other measurement model types would give diagrams that constitute a variant of Figure 1.

For *multivariate* measurement models, Steps 1, 2 and 3 would be performed as above (Sections 3, 4.2 and 4.3). The extension of steps 4 and 5 to the evaluation of coverage *regions* for

[6] For a general measurement model, the measurement model is solved for the estimate of the output quantity given estimates of the input quantities: see Section 3.

[7] For a general measurement model, the partial derivative of first order of the measurement model with respect to the output quantity is also required to determine the sensitivity coefficients: see Section 4.2.

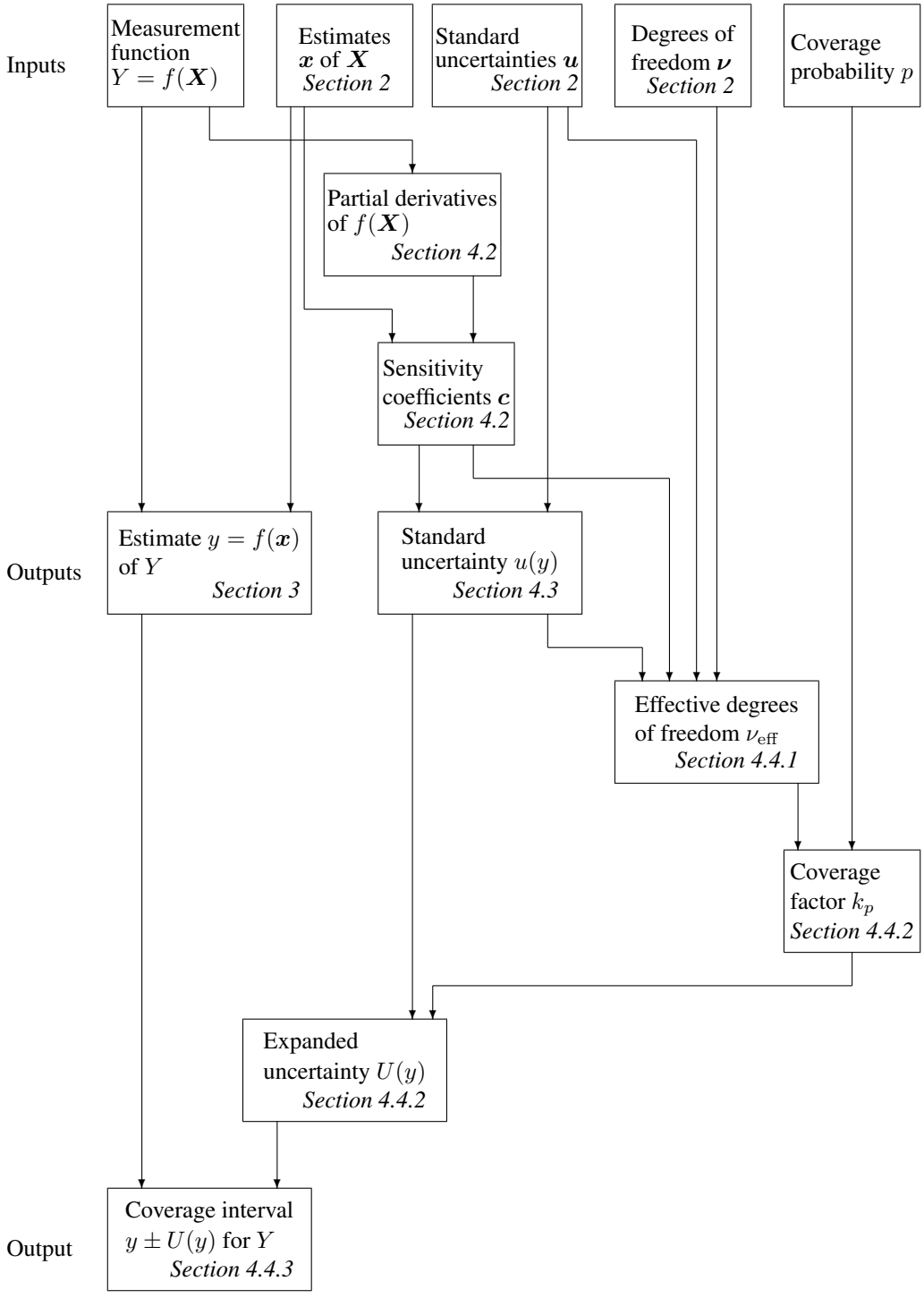

Figure 1: Uncertainty evaluation using the GUM uncertainty framework for a univariate, real measurement function with mutually independent input quantities.

multivariate output quantities is not straightforward. Provision of coverage regions is limited to those taking the form of ellipses (for the case of two output quantities) and ellipsoids (for the case of a general number of output quantities): see Section 4.5.

Implementation of the above procedure would be achieved in terms of software 'units' as described in Sections 3, 4.2, 4.3, and 4.4 or 4.5. For example, for a univariate, real measurement function, these units are specified in Tables 5, 9, 13, 14, 15 and 16.

4.2 Sensitivity coefficients

The procedure for the GUM uncertainty framework (Section 4.1) covers the univariate, real measurement function $Y = f(\boldsymbol{X})$. The *sensitivity coefficients* used by that procedure are denoted here by the (row) vector[8] $\boldsymbol{C} = (c_1, \ldots, c_N)$, whose jth element c_j is the partial derivative $\partial f/\partial X_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$. Table 9 specifies the evaluation of sensitivity coefficients for this category of measurement model, and indicates the input and output parameters necessary for their determination.

Table 10 specifies the counterpart for a multivariate, real measurement function $\boldsymbol{Y} = \boldsymbol{f}(\boldsymbol{X})$. In this case $\boldsymbol{C}$ takes the form of a matrix of sensitivity coefficients, whose (i, j)th element is the partial derivative of first order of the ith output quantity with respect to the jth input quantity evaluated at $\boldsymbol{X} = \boldsymbol{x}$.

Input parameters	
N	Number of input quantities
f	Function specifying the measurement function $Y = f(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
Output parameter	
$\boldsymbol{C}$	$1 \times N$ vector of sensitivity coefficients, whose jth element is the partial derivative $\partial f/\partial X_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$

Table 9: Sensitivity coefficients for a univariate, real measurement function.

The general univariate, real measurement model is $h(Y, \boldsymbol{X}) = 0$. The sensitivity coefficients $\boldsymbol{C} = (c_1, \ldots, c_N)$ are determined from the partial derivatives of first order of h with respect to both the input quantities $\boldsymbol{X}$ and the output quantity Y. The jth sensitivity coefficient c_j is given by

$$-\left(\frac{\partial h}{\partial Y}\right)^{-1}\left(\frac{\partial h}{\partial X_j}\right),$$

[8] The symbol $\boldsymbol{C}$ rather than the 'more natural' $\boldsymbol{c}$ for this vector is used to denote this set of coefficients. The reason for this choice is that for multivariate measurement models $\boldsymbol{C}$ is used to hold an array (matrix) of sensitivity coefficients, and that it is appropriate to use the same symbol throughout.

Input parameters	
N	Number of input quantities
m	Number of output quantities
$\boldsymbol{f}$	Function specifying the measurement function $\boldsymbol{Y} = \boldsymbol{f}(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
Output parameter	
$\boldsymbol{C}$	$m \times N$ matrix of sensitivity coefficients, whose (i, j)th element is the partial derivative $\partial f_i / \partial X_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$

Table 10: Sensitivity coefficients for a multivariate, real measurement function.

where the partial derivatives of first order are evaluated at $\boldsymbol{X} = \boldsymbol{x}$ and $Y = y$, with y satisfying $h(y, \boldsymbol{x}) = 0$. Table 11 specifies the evaluation of sensitivity coefficients for this category of measurement model, and indicates the input and output parameters necessary for their determination.

The general multivariate, real measurement model is $\boldsymbol{h}(\boldsymbol{Y}, \boldsymbol{X}) = \boldsymbol{0}$. The matrix $\boldsymbol{C}$ of sensitivity coefficients is determined as the solution to the linear system of equations

$$\boldsymbol{H_y C} = -\boldsymbol{H_x},$$

where the matrices $\boldsymbol{H_x}$ and $\boldsymbol{H_y}$ contain, respectively, the partial derivatives $\partial h_i / \partial X_j$ and $\partial h_i / \partial Y_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$ and $\boldsymbol{Y} = \boldsymbol{y}$, with $\boldsymbol{y}$ satisfying $\boldsymbol{h}(\boldsymbol{y}, \boldsymbol{x}) = \boldsymbol{0}$. Table 12 specifies the evaluation of sensitivity coefficients for this category of measurement model, and indicates the input and output parameters necessary for their determination.

Counterparts would apply for complex measurement models. It is necessary to form partial derivatives of first order of the real and imaginary parts of the (components of the) measurement model with respect to the real and imaginary parts of the input quantities (and the output quantities for general measurement models).

The sensitivity coefficients can be formed [8]:

1. Manually, by the algebraic differentiation of, e.g., $f(\boldsymbol{X})$ with respect to each component X_i, $i = 1, \ldots, N$, of $\boldsymbol{X}$, followed by setting $\boldsymbol{X} = \boldsymbol{x}$;

2. As 1, except by the use of a computer package for algebraic differentiation or a symbolic-algebra package that provides this capability;

3. By the use of finite-difference formulae, including that based on the complex-step method;

Input parameters	
N	Number of input quantities
h	Function specifying the measurement model $h(Y, \boldsymbol{X}) = 0$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$ and the output quantity Y
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
y	Estimate of the output quantity Y that satisfies $h(y, \boldsymbol{x}) = 0$
Output parameter	
$\boldsymbol{C}$	$1 \times N$ vector of sensitivity coefficients, whose jth element is $$-\left(\frac{\partial h}{\partial Y}\right)^{-1}\left(\frac{\partial h}{\partial X_j}\right),$$ where the partial derivatives of first order are evaluated at $\boldsymbol{X} = \boldsymbol{x}$ and $Y = y$

Table 11: Sensitivity coefficients for a univariate, real measurement model.

Input parameters	
N	Number of input quantities
m	Number of output quantities
$\boldsymbol{h}$	Function specifying the measurement model $\boldsymbol{h}(\boldsymbol{Y}, \boldsymbol{X}) = \boldsymbol{0}$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$ and the output quantities $\boldsymbol{Y} = (Y_1, \ldots, Y_m)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
$\boldsymbol{y}$	Column vector $(y_1, \ldots, y_m)^\top$ of estimates of the output quantities $\boldsymbol{Y}$ that satisfy $\boldsymbol{h}(\boldsymbol{y}, \boldsymbol{x}) = \boldsymbol{0}$
Output parameter	
$\boldsymbol{C}$	$m \times N$ matrix of sensitivity coefficients that solves $$\boldsymbol{H_y C} = -\boldsymbol{H_x},$$ where the matrices $\boldsymbol{H_x}$ and $\boldsymbol{H_y}$ contain, respectively, the partial derivatives $\partial h_i / \partial X_j$ and $\partial h_i / \partial Y_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$ and $\boldsymbol{Y} = \boldsymbol{y}$
Numerical analysis	
	The above system of equations should be solved using a numerically stable procedure, such as Gaussian elimination with a pivoting strategy [24]

Table 12: Sensitivity coefficients for a multivariate, real measurement model.

4. By the use of program differentiation techniques, including forward automatic differentiation and reverse automatic differentiation (which are examples of operator overloading), and source to source transformation.

The manner in which the partial derivatives of first order required in forming the sensitivity coefficients $\boldsymbol{C}$ in Tables 9 to 12 are obtained requires careful consideration. These derivatives and hence the sensitivity coefficients can be difficult to determine by hand for measurement models that are complicated. The above options of using a symbolic-algebra package, finite-difference formulae or program differentiation techniques can be attractive in such circumstances. There are learning overheads associated with the use of a symbolic-algebra package and program differentiation techniques: their use can be justified if the user needs to address a sufficient number of complicated measurement models. Finite-difference formulae may provide inadequate accuracy if used inappropriately. These alternatives to the manual determination of sensitivity coefficients are addressed in Appendices A, B and C.

4.3 Uncertainty associated with the estimate of the output quantity

For a univariate, real measurement function, the standard uncertainty $u(y)$ associated with the estimate y of Y is obtained from the formula [3]

$$u^2(y) = \sum_{i=1}^{N} \sum_{j=1}^{N} c_i c_j \mathrm{cov}(x_i, x_j),$$

where c_i is the sensitivity coefficient for the ith input quantity, and $\mathrm{cov}(x_i, x_j)$ the covariance associated with the estimates x_i and x_j, with $\mathrm{cov}(x_i, x_i) = u_i^2$, the variance (squared standard uncertainty) associated with the ith estimate. A compact way of representing the above expression is

$$\boldsymbol{U_y} = \boldsymbol{C}\boldsymbol{U_x}\boldsymbol{C}^\top, \tag{1}$$

where $\boldsymbol{U_y} = u^2(y)$, $\boldsymbol{C} = (c_1, \ldots, c_N)$, and $\boldsymbol{U_x}$ is the covariance matrix of dimension $N \times N$ associated with the estimates $\boldsymbol{x}$ of the input quantities $\boldsymbol{X}$.

Using this notation, formula (1) for $\boldsymbol{U_y}$ applies for *all* categories of measurement model. For univariate, real measurement models, $\boldsymbol{U_y}$ is the variance associated with the estimate y of the output quantity Y; for other measurement models, it is the covariance matrix associated with the estimates of the output quantities. Table 13 specifies the evaluation of the uncertainty associated with the estimate of the output quantity, and indicates the input and output parameters necessary for its determination.

Input parameters	
N	Number of input quantities
m	Number of output quantities
$\boldsymbol{U_x}$	$N \times N$ covariance matrix associated with the estimates $\boldsymbol{x}$ of the input quantities $\boldsymbol{X}$
$\boldsymbol{C}$	$m \times N$ matrix of sensitivity coefficients
Output parameter	
$\boldsymbol{U_y}$	Covariance matrix of dimension $m \times m$ associated with the estimate y (or $\boldsymbol{y}$) of the output quantity Y (or $\boldsymbol{Y}$) obtained by evaluating $$\boldsymbol{U_y} = \boldsymbol{C}\boldsymbol{U_x}\boldsymbol{C}^\top$$

Table 13: Variance (squared standard uncertainty) associated with the estimate y of Y for a univariate ($m = 1$) measurement model or the covariance matrix associated with the estimate $\boldsymbol{y}$ of $\boldsymbol{Y}$ for a multivariate ($m > 1$) measurement model.

4.4 Coverage interval for a univariate, real output quantity

4.4.1 Effective degrees of freedom

Table 14 specifies the Welch-Satterthwaite formula for evaluating the effective degrees of freedom ν_{eff} for a univariate, real output quantity Y when the input quantities $\boldsymbol{X}$ are mutually independent. The GUM uncertainty framework does not state how ν_{eff} is to be calculated when the input quantities are correlated.

4.4.2 Expanded uncertainty

Table 15 specifies the calculation of expanded uncertainty $U(y)$ associated with the estimate y of a univariate, real output quantity Y when the input quantities $\boldsymbol{X}$ are mutually independent. The expanded uncertainty is evaluated as the product of the standard uncertainty $u(y)$ (Section 4.3) and a coverage factor k_p that depends on the required coverage probability p and the effective degrees of freedom ν_{eff} (Section 4.4.1).

The calculation of expanded uncertainty depends on knowledge of the distribution that characterizes the output quantity Y. In the GUM uncertainty framework, a t–distribution with ν_{eff} degrees of freedom is assigned to the random variable

$$T = \frac{Y - y}{u(y)}.$$

Input parameters	
N	Number of input quantities
$\boldsymbol{u}$	Column vector $(u_1, \ldots, u_N)^\top \equiv (u(x_1), \ldots, u(x_N))^\top$ of standard uncertainties associated with the estimates $\boldsymbol{x} = (x_1, \ldots, x_N)^\top$ of the input quantities $\boldsymbol{X}$
$\boldsymbol{C}$	$1 \times N$ (row) vector of sensitivity coefficients, whose jth element is the partial derivative $\partial f / \partial X_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$
$\boldsymbol{\nu}$	Column vector $(\nu_1, \ldots, \nu_N)^\top$ of degrees of freedom. If the estimate x_i of X_i is taken as the mean of a set of q repeated indication values, ν_i is taken as $q - 1$. If a rectangular distribution with accurately known endpoints is assigned to X_i, ν_i is taken as infinite (∞). (Since ∞ cannot be represented as such as an input parameter, a convention may be adopted. For instance such a value can be 'coded' as 0 (zero), and the procedure would be designed to interpret this value, which cannot occur otherwise, as infinite.) These are two important cases. There are other possibilities [3] for attaching a degrees of freedom to the standard uncertainties associated with estimates of the input quantities. Each case is treated on its own merits
$u(y)$	Standard uncertainty associated with the estimate y of the output quantity
Output parameter	
ν_{eff}	Effective degrees of freedom determined from the Welch-Satterthwaite formula $$\frac{u^4(y)}{\nu_{\text{eff}}} = \sum_{i=1}^{N} \frac{c_i^4 u^4(x_i)}{\nu_i}$$

Table 14: Effective degrees of freedom according to the Welch-Satterthwaite formula.

It follows that k_p is the percentage point $t_p(\nu_{\text{eff}})$ of the t–distribution such that the probability that $|t|$ is no greater than $t_p(\nu_{\text{eff}})$ is equal to p, i.e.,

$$p = \Pr\left[|t| \leq t_p(\nu_{\text{eff}})\right],$$

where t has a t–distribution with ν_{eff} degrees of freedom. Values of k_p for various choices of coverage probability p and degrees of freedom ν_{eff} may be obtained from statistical tables and implemented as a 'look-up' table. Many mathematical and statistical software libraries provide an implementation and can be used. For $\nu_{\text{eff}} \geq 473$, $k_p = 1.96$, correct to two decimal places, the corresponding value for the standard Gaussian distribution $\mathrm{N}(0, 1)$.

Input parameters	
$u(y)$	Standard uncertainty associated with the estimate y of the output quantity
ν_{eff}	Effective degrees of freedom determined from the Welch-Satterthwaite formula
p	Coverage probability (typically 0.95)
Output parameter	
$U(y)$	Expanded uncertainty, defined by $$U(y) = k_p u(y),$$ where k_p is a coverage factor, depending on the stipulated coverage probability p, that is obtained from tables of percentage points of the Gaussian distribution ($\nu_{\text{eff}} = \infty$) or a t–distribution ($\nu_{\text{eff}} < \infty$)

Table 15: Expanded uncertainty for a univariate, real measurement model.

4.4.3 Coverage interval

Table 16 specifies the calculation of the coverage interval for a univariate, real output quantity Y when the input quantities $\boldsymbol{X}$ are mutually independent. The interval is centred at the estimate y of the output quantity (Section 3) with a semi-width equal to the expanded uncertainty associated with the estimate (Section 4.4.2). The coverage interval can be expected to contain $100p\,\%$ of the values that can reasonably be attributed to the output quantity.

4.5 Coverage region for a multivariate output quantity

For a multivariate quantity $\boldsymbol{Y}$, the counterpart of a coverage interval is a coverage region in m-dimensions that contains $\boldsymbol{Y}$ with specified coverage probability p. If the estimate $\boldsymbol{y}$ and associated covariance matrix $\boldsymbol{U_y}$ constitute the only available information about $\boldsymbol{Y}$, the

Input parameters	
y	Estimate of the output quantity
$U(y)$	Expanded uncertainty corresponding to required coverage probability p
Output parameter	
$y_{\text{low}}, y_{\text{high}}$	Endpoints of a coverage interval, defined by $$y_{\text{low}} = y - U(y), \quad y_{\text{high}} = y + U(y)$$

Table 16: Coverage interval for a univariate, real measurement model.

joint PDF characterizing $\boldsymbol{Y}$ is the multivariate Gaussian $\mathrm{N}(\boldsymbol{y}, \boldsymbol{U}_{\boldsymbol{y}})$ [4, Clause 6.4.8]. Then, the boundary of the coverage region for $\boldsymbol{Y}$ of smallest (hyper-)volume is the ellipsoid

$$(\boldsymbol{\eta} - \boldsymbol{y})^{\top} \boldsymbol{U}_{\boldsymbol{y}}{}^{-1} (\boldsymbol{\eta} - \boldsymbol{y}) = k_{p,m}^2$$

centred at $\boldsymbol{y}$, with $k_{p,m}^2$ given by an upper percentage point of the chi-squared distribution with m degrees of freedom and satisfying

$$p = \Pr(\chi_m^2 \leq k_{p,m}^2),$$

where χ_m^2 has a chi-squared distribution with m degrees of freedom [34]. Values of $k_{p,m}^2$ for various choices of coverage probability p and number m of output quantities may be obtained from statistical tables and implemented as a 'look-up' table. Many mathematical and statistical software libraries provide an implementation and can be used.

5 Monte Carlo method

5.1 Procedure

For the application of a Monte Carlo method, the outputs of the formulation stage are the PDFs[9] $\boldsymbol{g}(\boldsymbol{\xi}) = (g_1(\xi_1), \ldots, g_n(\xi_N))^{\top}$ for the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^{\top}$ (Section 2).[10] The PDFs, together with the measurement model and the required coverage probability p (e.g., 0.95), constitute the inputs to the calculation stage of the Monte Carlo procedure.

For a *univariate measurement function*, the procedure is as follows:

[9] A joint (multivariate) PDF for (a subset of) the input quantities is also possible (Section 2.1.3).

[10] The following notation is used: X_i to denote the ith input quantity, x_i an estimate of X_i, and ξ_i a (general) value of X_i. Hence, the PDF for X_i is written as a function of ξ_i. Similarly, Y, y and η are used for a (univariate) output quantity.

1. Select the number M of Monte Carlo trials to be made. See Section 5.2.

2. Generate M vectors by making random draws from the PDFs for the (set of N) input quantities. See Section 5.3.

3. For each vector, evaluate the measurement function to give the corresponding value of the output quantity. See Section 5.4.1.

4. Calculate the estimate of the output quantity and the associated standard uncertainty as the (arithmetic) mean and standard deviation of the values of the output quantity. Optionally, use the values to form an approximation to the PDF for the output quantity. See Sections 5.4.2 and 5.4.3.

5. Sort the values of the output quantity into non-decreasing order, and use the sorted values to provide a discrete representation of the distribution function for the output quantity. Optionally, use the discrete representation to form a (continuous) approximation to the distribution function for the output quantity. See Sections 5.4.4 and 5.4.5.

6. Use the discrete representation of the distribution function to calculate a coverage interval for the output quantity for the required coverage probability p. See Section 5.4.6.

Figure 2 shows the procedure diagrammatically.

The modifications to the procedure illustrated in Figure 2 for other types of univariate measurement model are straightforward. Multivariate measurement models are considered in Section 5.5.

Section 5.6 describes how a Monte Carlo method may be used to undertake a sensitivity analysis for the measurement model with respect to each input quantity, yielding 'non-linear' sensitivity coefficients that are the counterpart of (linear) sensitivity coefficients necessary for the implementation of the GUM uncertainty framework.

Section 5.7 indicates a basic implementation of an adaptive Monte Carlo procedure that removes the need to make an *a priori* choice of the number of Monte Carlo trials.

5.2 The number of Monte Carlo trials

A value of M, the number of Monte Carlo trials to be made, needs to be selected. It can be chosen *a priori*, in which case there will be no direct control over the degree of approximation delivered by the Monte Carlo procedure. The reason is that the number needed to provide a prescribed degree of approximation will depend on the 'shape' of the PDF for the output quantity and the coverage probability required. Also, the calculations are stochastic in nature, being based on making random draws from the PDFs for the input

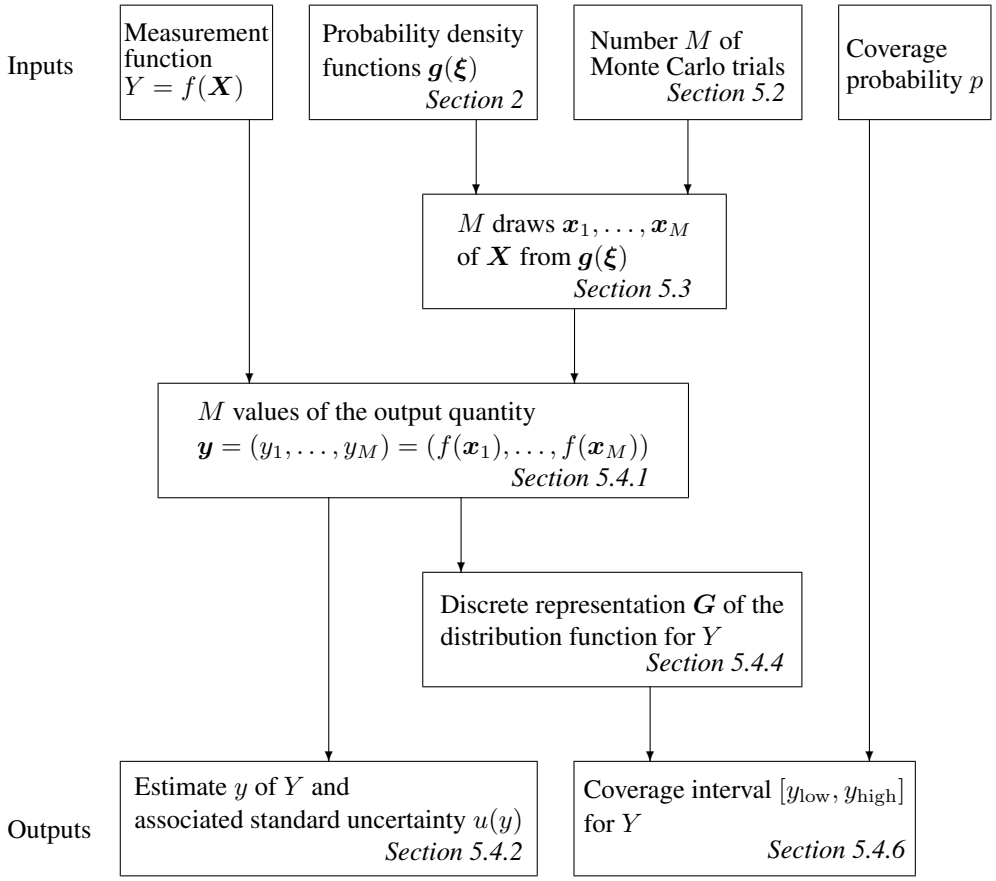

Figure 2: Uncertainty evaluation using a Monte Carlo method for a univariate, real measurement function.

quantities. However, a value of $M = 10^6$ can often be expected to deliver a $95\,\%$ coverage interval, having a length with a degree of approximation of one or two significant decimal digits, for the output quantity.

Because there is no guarantee that this or any specific number will suffice, it is recommended to use a process that selects M adaptively, i.e., as the trials progress. A property of such a process is that it takes a number of trials that is economically consistent with the achievement of the required degree of approximation [1, 4, 14, 51]. A basic implementation of an adaptive Monte Carlo procedure is described in Section 5.7.

5.3 Making random draws from the probability density functions

In an implementation of the Monte Carlo procedure M vectors $\boldsymbol{x}_r$, $r = 1, \ldots, M$, are randomly drawn from the PDFs for the input quantities $\boldsymbol{X}$. Random draws are made from a joint (multivariate) distribution when appropriate (Section 2.1.3).

Recommendations concerning the manner in which random draws should be made are given here for the commonest distributions, viz., the rectangular, the Gaussian, the t–distribution, the curvilinear trapezoid, the U-shaped, the multivariate Gaussian and the multivariate t. It is possible to prepare software for making random draws from almost any distribution, and indeed to develop a general framework for doing so (Section 5.3.6).

Tests of randomness of the numbers produced by a generator are indicated.

5.3.1 Rectangular distribution

The ability to generate pseudo-random numbers from a rectangular distribution is fundamental in its own right, and also as the basis for generating numbers from any distribution (Section 5.3.6) using an appropriate algorithm or formula. In the latter regard, the quality of the numbers generated from a non-rectangular distribution depends on that of the numbers generated from the rectangular distribution and on the properties of the algorithm employed. The quality of the non-rectangular generator can therefore be expected to be related to that of the rectangular generator. A good rectangular generator and a good algorithm can be expected to provide a good non-rectangular generator. A poor rectangular generator and a good or bad algorithm can be expected to provide a poor non-rectangular generator. It is thus especially important that the underlying rectangular generator is sound (cf. [32]). Unless the user is sure of the pedigree of a rectangular generator it should not be used until adequate testing has been carried out. Invalid results can otherwise be obtained. Some of the ‘tests for randomness’ that should be undertaken are indicated below. A recommended rectangular pseudo-random number generator, that has been shown to perform well in these tests and that is straightforward to implement, is given in this section.

Table 17 defines relevant aspects of the functioning of a procedure for generating rectan-

gular pseudo-random numbers in the interval $(0,\ 1)$, specifying the input, input-output and output parameters associated with their determination.

Input parameter	
q	Number of rectangular pseudo-random numbers to be generated
Input-output parameter	
$\boldsymbol{t}$	Column vector of parameters required as input quantities and that may be changed as part of the computation. The subsequent values of these quantities are not usually of immediate concern to the user. The parameters are needed to help control the process by which the pseudo-random numbers are produced. The parameters may be realized as global variables and thus not explitly appear as parameters of the procedure. One or more of these parameters may be a seed, used to initiate the sequence of random numbers produced by successive calls of the procedure. By setting the seed(s) to values previously used, the same sequence of random numbers can be produced. Doing so is important as part of software regression testing, used to verify the consistency of results produced using the software with those from previous versions
Output parameter	
$\boldsymbol{r}$	q random draws from a rectangular distribution between zero and one

Table 17: Pseudo-random number generation.

A random draw from the rectangular distribution on the interval $(a,\ b)$ can be formed from $a + (b - a)r$, where r is a random draw from the rectangular distribution on the interval $(0,\ 1)$.

A review [39] has been carried out on the use of random numbers in solving problems using a Monte Carlo procedure. It draws conclusions concerning, in particular, the best methods to employ for generating rectangular pseudo-random numbers. The so-called 'combination generators' are recommended and are reported as being favoured by experts as fulfilling the requirements of possessing the longest periods[11] and passing a set of statistical tests for randomness.[12]

A combination generator simultaneously uses more than one generator. Each such generator is typically a member of the class of congruential generators or the class of shift register generators, both of which are widely discussed in the literature [22, 30, 40, 42].

[11] A pseudo-random number generator provides a sequence of numbers. The period of the sequence is the number of consecutive values in the sequence before they are repeated.

[12] The tests include the so-called *standard tests* [30], viz., the χ^2 test, the Kolmogorov-Smirnov test, the frequency test, the serial test, the gap test, the poker test, the coupon collector's test and the more stringent *Die Hard tests* [35], that include the overlapping M-tuple test, the overlapping permutation test, the parking lot and lattice test and the birthday-spacing test.

The KISS[13] generator [36] is a combination of a congruential generator and two shift register generators. A version in the C programming language is available [41, p42] and in Fortran [36].

The test suite TestU01 [31] may be used to carry out an extensive test of the statistical properties of any generator submitted to it. The suite is very detailed, with many individual tests, including the so-called 'Big Crush'. Several generators that pass the suite of tests are listed by Wichmann and Hill [50]. An enhanced Wichmann-Hill generator also passes the test and is recommended [4].

Table 18 defines the enhanced Wichmann-Hill generator for generating rectangular pseudo-random numbers in the interval $(0,\ 1)$.

5.3.2 Gaussian distribution

The procedure in Table 19 provides a straightforwardly implementable approach [9] to generate random draws from the standard Gaussian distribution $\mathrm{N}(0,\ 1)$ using the Box-Muller transform.

A random draw from the Gaussian distribution $\mathrm{N}(\mu, \sigma^2)$ can be formed from $\mu + \sigma z$, where z is a random draw from the standard Gaussian distribution $\mathrm{N}(0, 1)$.

5.3.3 t–distribution

The procedure in Table 20 provides an approach [29], [41, p63] to generate random draws from the t–distribution t_ν with ν degrees of freedom, that is also straightforward to implement.

A random draw from the t–distribution $t_\nu(\mu, \sigma^2)$ with shift parameter μ and scale parameter σ can be formed from $\mu + \sigma t$, where t is a random draw from the t–distribution t_ν.

5.3.4 Curvilinear trapezoid distribution

A random draw from the curvilinear trapezoid distribution $\mathrm{CTrap}(a, b, d)$ can be formed from

$$a_s + (b_s - a_s)r_2,$$

where $a_s = (a - d) + 2dr_1$, $b_s = (a + b) - a_s$ and r_1 and r_2 are random draws made independently from the rectangular distribution R(0, 1).

[13] Keep It Simple, Stupid!

Input parameter	
None	
Input-output parameters	
ix, iy, iz, it	Integer parameters required as input parameters and that are changed by the procedure. Set to integers between 1 and 2 147 483 647 before the first call. Do not disturb between calls. Subsequent values of these parameters are not usually of concern to the user. The parameters provide the basis by which the pseudo-random numbers are generated. They may be realized as global variables and thus not appear explicitly as parameters of the procedure
Output parameter	
r	A random draw from a rectangular distribution between zero and one
Computation	

1. Form $ix = 11\,600 \times (ix \bmod 185\,127) - 10\,379 \times \lfloor ix \div 185\,127 \rfloor$;
2. Form $iy = 47\,003 \times (iy \bmod 45\,688) - 10\,479 \times \lfloor iy \div 45\,688 \rfloor$;
3. Form $iz = 23\,000 \times (iz \bmod 93\,368) - 19\,423 \times \lfloor iz \div 93\,368 \rfloor$;
4. Form $it = 33\,000 \times (it \bmod 65\,075) - 8\,123 \times \lfloor it \div 65\,075 \rfloor$;
5. If $ix < 0$, form $ix = ix + 2\,147\,483\,579$;
6. If $iy < 0$, form $iy = iy + 2\,147\,483\,543$;
7. If $iz < 0$, form $iz = iz + 2\,147\,483\,423$;
8. If $it < 0$, form $it = it + 2\,147\,483\,123$;
9. Form $w = wx + wy + wz + wt$, where
   - 9.1 $wx = ix/2\,147\,483\,579.0$,
   - 9.2 $wy = iy/2\,147\,483\,543.0$,
   - 9.3 $wz = iz/2\,147\,483\,423.0$,
   - 9.4 $wt = it/2\,147\,483\,123.0$;
10. Form $r = w - \lfloor w \rfloor$.

Table 18: Enhanced Wichmann-Hill pseudo-random number generator for the rectangular distribution. $ix \bmod n$ denotes the modulus (or remainder) after division by n, and $\lfloor w \rfloor$ denotes the integer part of w.

Input parameter	
None	
Output parameters	
z_1, z_2	Two random draws from the standard Gaussian distribution $\mathrm{N}(0,\ 1)$
Computation	

1. Generate random draws r_1 and r_2 independently from the rectangular distribution between zero and one
2. Form $z_1 = \sqrt{-2 \log r_1} \cos 2\pi r_2$ and $z_2 = \sqrt{-2 \log r_1} \sin 2\pi r_2$

Table 19: Box-Muller pseudo-random number generator for the Gaussian distribution.

Input parameter	
ν	Degrees of freedom
Output parameter	
t	A random draw from a t–distribution with ν degrees of freedom
Computation	

1. Generate random draws r_1 and r_2 independently from the rectangular distribution between zero and one
2. If $r_1 < 1/2$, form $t = 1/(4r_1 - 1)$ and $v = r_2/t^2$; otherwise set $t = 4r_1 - 3$ and $v = r_2$
3. If $v < 1 - |t|/2$ or $v < (1 + t^2/\nu)^{-(\nu+1)/2}$, accept t as a random draw from the t–distribution; otherwise repeat from Step 1

Table 20: Pseudo-random number generator for the t–distribution.

5.3.5 U-shaped distribution

A random draw from the U-shaped distribution $\mathrm{U}(a, b)$ can be formed from

$$(a + b)/2 + (b - a)/2 \sin(2\pi r),$$

where r is a random draw from the rectangular distribution R(0, 1).

5.3.6 General univariate distributions

Sections 5.3.1 to 5.3.5 presented procedures for making random draws from the PDFs relating to common (univariate) distributions, viz., rectangular, Gaussian, t–distribution, curvilinear trapezoid and U-shaped, respectively. Consideration is given here to the task of making random draws from a general (univariate) distribution defined by its distribution function $G(\xi)$.

A random draw z from this distribution is obtained as follows:

1. Generate a random draw ψ from a rectangular distribution between zero and one (Section 5.3.1)
2. Find the value z satisfying $G(z) = \psi$.

The 'inversion' step (in 2 above) of forming $z = G^{-1}(\psi)$ may be possible analytically or, otherwise, performed numerically. In the latter case, z is evaluated by solving the equation

$$G(z) - \psi = 0.$$

Upper and lower bounds for z are generally easily found, in which case a 'bracketing' algorithm such as bisection can be used to determine z [16, 23].

5.3.7 Multivariate Gaussian distribution

The most important multivariate distribution is the multivariate Gaussian distribution. A vector $\boldsymbol{\mu}$ of dimension $n \times 1$ of expectations and a covariance matrix $\boldsymbol{V}$ of dimension $n \times n$ are the parameters of the n-dimensional Gaussian distribution $\mathrm{N}(\boldsymbol{\mu}, \boldsymbol{V})$. Random draws can be made from this distribution [43, 47] using the procedure in Table 21.

Figure 3 shows three examples of $1\,000$ points generated from bivariate Gaussian distributions using the MULTNORM generator [43]. In all three cases the distributions characterize a quantity with expectation $\boldsymbol{\mu} = (2, 3)^\top$. In the top graph, the quantity has covariance matrix

$$\boldsymbol{V} = \begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 2.0 \end{bmatrix},$$

Input parameters	
n	Dimension of the multivariate Gaussian distribution
$\boldsymbol{\mu}$	$n \times 1$ vector of expectations
$\boldsymbol{V}$	$n \times n$ covariance matrix
q	Number of random draws from the multivariate Gaussian distribution

Output parameter	
$\boldsymbol{\Phi}$	$n \times q$ matrix, the jth column of which is a random draw from the multivariate Gaussian distribution

Computation

1. Form the Cholesky factor $\boldsymbol{R}$ of $\boldsymbol{V}$, i.e., the upper triangular matrix satisfying $\boldsymbol{V} = \boldsymbol{R}^\top \boldsymbol{R}$. (To generate q pseudo-random numbers it is necessary to perform this matrix factorization only once.)

2. Generate q random draws from the n-dimensional standard Gaussian distribution $\mathrm{N}(0,1) \times \cdots \times \mathrm{N}(0,1)$. Doing so simply means generating an $n \times q$ array $\boldsymbol{Z}$ of random draws from the standard Gaussian distribution

3. Provide the required random draws (the Cholesky factor acts as a transformation from the uncorrelated standardized space to that required):
$$\boldsymbol{\Phi} = \boldsymbol{\mu}\mathbf{1}^\top + \boldsymbol{R}^\top \boldsymbol{Z},$$
where $\mathbf{1}$ denotes a $q \times 1$ vector of ones

Table 21: Pseudo-random number generator for the multivariate Gaussian distribution.

i.e., the components of the quantity are independent and have the same standard deviation, and the cloud of points resembles a disk with centre at the expectation $\boldsymbol{\mu}$. In the middle graph, the quantity has covariance matrix

$$\boldsymbol{V} = \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 4.0 \end{bmatrix},$$

i.e., the components are independent and have different standard deviations, and the cloud of points resembles an ellipse with major and minor axes parallel to the co-ordinate axes. In the bottom graph, the quantity has covariance matrix

$$\boldsymbol{V} = \begin{bmatrix} 2.0 & 1.9 \\ 1.9 & 2.0 \end{bmatrix},$$

i.e., the components are correlated, and the cloud of points resembles an ellipse with axes oriented to the co-ordinate axes by an angle that is determined by the covariance of the components.

Similar generators are available elsewhere [17].

5.3.8 Multivariate t–distribution

Another important multivariate distribution is the multivariate t–distribution (Section 2.1.3). A vector $\boldsymbol{\mu}$ of dimension $n \times 1$ of expectations, a scale matrix $\boldsymbol{S}$ of dimension $n \times n$ and a degrees of freedom ν are the parameters of the n-dimensional t–distribution $t_\nu(\boldsymbol{\mu}, \boldsymbol{S})$. Random draws can be made from this distribution using the procedure given in Table 22. The procedure relies on making random draws from a chi-squared distribution with ν degrees of freedom (see, e.g., [19]).

5.4 Monte Carlo procedure for univariate measurement models

5.4.1 Calculation of the values of the output quantity

Denote by $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M$ the M random draws from the PDFs for the N input quantities, where the rth draw $\boldsymbol{x}_r$ contains values $x_{1,r}, \ldots, x_{N,r}$, with $x_{i,r}$ a draw from the PDF for X_i. For a univariate, real measurement function, the corresponding values of the output quantity are obtained by evaluating the measurement function:

$$y_r = f(\boldsymbol{x}_r),\ r = 1, \ldots, M.$$

For a univariate, real measurement model, they are determined by solving the measurement model:

$$h(y_r, \boldsymbol{x}_r) = 0,\ r = 1, \ldots, M.$$

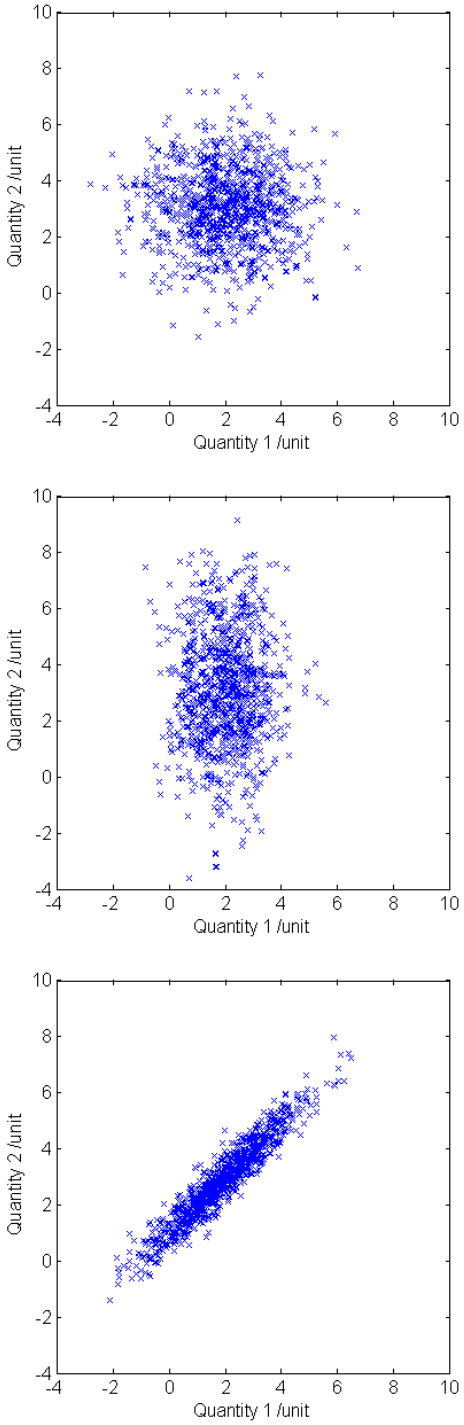

Figure 3: Points drawn from bivariate Gaussian distributions used to characterize quantities that are independent with the same standard deviation (top), independent having different standard deviations (middle), and correlated (bottom).

Input parameters	
n	Dimension of the multivariate t–distribution
$\boldsymbol{\mu}$	$n \times 1$ vector of expectations
$\boldsymbol{S}$	$n \times n$ scale matrix
ν	Degrees of freedom
q	Number of random draws from the multivariate t–distribution

Output parameter	
$\boldsymbol{T}$	$n \times q$ matrix, the jth column of which is a random draw from the multivariate t–distribution

Computation

1. Form the Cholesky factor $\boldsymbol{R}$ of $\boldsymbol{S}$, i.e., the upper triangular matrix satisfying $\boldsymbol{S} = \boldsymbol{R}^\top \boldsymbol{R}$. (To generate q pseudo-random numbers it is necessary to perform this matrix factorization only once.)

2. Generate q random draws from the n-dimensional standard Gaussian distribution $\mathrm{N}(0,1) \times \cdots \times \mathrm{N}(0,1)$. Doing so simply means generating an $n \times q$ array $\boldsymbol{Z}$ of random draws from the standard Gaussian distribution

3. Generate q random draws $w_1, \ldots, w_q$ from the chi-squared distribution with ν degrees of freedom

4. Provide the required random draws:

$$\boldsymbol{T} = \boldsymbol{\mu}\mathbf{1}^\top + \boldsymbol{R}^\top \boldsymbol{Z}\boldsymbol{D},$$

where $\mathbf{1}$ denotes a $q \times 1$ vector of ones and $\boldsymbol{D}$ a diagonal matrix of dimension $q \times q$ with diagonal elements $\sqrt{\nu/w_1}, \ldots, \sqrt{\nu/w_q}$

Table 22: Pseudo-random number generator for the multivariate t–distribution.

The values $y_1, \ldots, y_M$ are used in the evaluation of the estimate y of Y and the associated standard uncertainty $u(y)$ (Section 5.4.2), and as the basis for calculating an approximation to the PDF for Y (Section 5.4.3). The values are also used to provide a discrete representation of the distribution function for Y (Section 5.4.4) in terms of which are obtained a (continuous) approximation to the distribution function (Section 5.4.5) and a coverage interval for Y (Section 5.4.6). The case of a univariate measurement model is covered below. Multivariate measurement models are considered in Section 5.5.

An 'updating' procedure for forming y, $u(y)$ and approximations to the PDF and distribution function for the output quantity, which avoids the need to store and sort the complete set of values of the output quantity, is described in Appendix E.

Section 3 provides advice on calculating a value of the output quantity given values for the input quantities. Note that in the Monte Carlo procedure a value of the output quantity is calculated for each random draw of the input quantities and hence for values that may be distanced by 'several standard deviations' from the estimates of the input quantities. This is in contrast to the GUM uncertainty framework in which a value of the output quantity is calculated only for the estimates of the input quantities and, if finite difference approximations are used [3, Clause 5.1.3], also for points perturbed from these estimates by $\pm$ one standard deviation for each quantity in turn. For this reason some issues may arise regarding the numerical procedure used to calculate a value of the output quantity, e.g., ensuring its convergence (where iterative schemes are used) and numerical stability. The user should ensure that, where appropriate, the numerical methods used to calculate a value of the output quantity are valid for a sufficiently large region centred on the estimates of the input quantities.

5.4.2 Estimate of the output quantity and the associated standard uncertainty

The average of the values y_r, $r = 1, \ldots, M$, of the output quantity is taken as the estimate y of the output quantity, and the standard deviation of the values is taken as the standard uncertainty $u(y)$ associated with y:

$$y = \frac{1}{M} \sum_{r=1}^{M} y_r, \tag{2}$$

and

$$u^2(y) = \frac{1}{M-1} \sum_{r=1}^{M} (y_r - y)^2. \tag{3}$$

Table 23 specifies the evaluation of the estimate y of Y and the associated standard uncertainty $u(y)$ using formulae (2) and (3), respectively, and indicates the input and output parameters necessary for their determination.

The evaluations of y and $u(y)$ (Table 23) require the summation of M numbers with M large (typically of the order of 10^5 or 10^6: Section 5.2). A procedure for undertaking these

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{y}$	Values $(y_1, \ldots, y_M)$ of the output quantity corresponding to the M random draws $\boldsymbol{x}_r$ from the PDFs for the N input quantities, i.e., $y_r = f(\boldsymbol{x}_r)$ or $h(y_r, \boldsymbol{x}_r) = 0$
Output parameters	
y	Estimate of the output quantity: the arithmetic mean of the values of the output quantity, defined by $$y = \frac{1}{M}\sum_{r=1}^{M} y_r$$
$u(y)$	Standard uncertainty: the standard deviation of the values of the output quantity, defined by $$u^2(y) = \frac{1}{M-1}\sum_{r=1}^{M}(y_r - y)^2$$
Numerical analysis	
	The above formula for $u(y)$ should be used rather than the mathematically equivalent formula $$u^2(y) = \frac{M}{M-1}\left(\frac{1}{M}\sum_{r=1}^{M} y_r^2 - y^2\right).$$ For cases in which $u(y)$ is very much smaller than $\lvert y\rvert$ (in which case the $y_r,\ r = 1, \ldots, M$, have a number of leading digits in common) the latter formula suffers from subtractive cancellation (involving a mean square less a squared mean). The cancellation effects can be so severe that the resulting value of $u(y)$ may have too few correct significant figures for the uncertainty evaluation to be valid [10]

Table 23: Estimate of the output quantity and the associated standard uncertainty for a univariate output quantity.

summations, designed to reduce the effect of rounding errors associated with the floating-point operations (of which there are many), is described in Appendix D. An 'updating' procedure for evaluating the estimate of the output quantity and the associated standard uncertainty that avoids the need to store the complete set of values of the output quantity is described in Appendix E.

The value of y so obtained yields the smallest mean squared deviation over all possible choices of the estimate of the output quantity. However, the value will not in general agree with that corresponding to the estimates of the input quantities [3, Clause 4.1.4]. Agreement (in a practical sense) will be achieved for a large value of M when the measurement model is linear in the input quantities. Whether this general lack of agreement is important depends on the application. The value of y, even in the limit as $M \to \infty$, is not in general equal to the value of the output quantity for the expectation values of the input quantities, unless the measurement model is linear [3, Clause 4.1.4].

5.4.3 Approximation to the probability density function

An approximation $\widetilde{g}(\eta)$ to the PDF $g(\eta)$ for the output quantity can be formed from the values $y_r,\ r = 1, \ldots, M$. These values, when assembled into a histogram with suitable bin widths, form a frequency distribution that, when normalized to have unit area, constitutes $\widetilde{g}(\eta)$. Calculations are not generally carried out in terms of this histogram, the resolution of which depends on the choice of bin size. The histogram can, however, be useful as an aid to understanding the nature of the PDF, e.g., the extent of its asymmetry.

One way of obtaining $\widetilde{g}(\eta)$ is as follows. Let $[\eta_0, \eta_b]$ be an interval of values of the output quantity partitioned into b subintervals $\{B_k : k = 1, \ldots, b\}$, where

$$B_k = \begin{cases} [\eta_{k-1}, \eta_k), & k = 1, \ldots, b-1, \\ [\eta_{k-1}, \eta_k], & k = b, \end{cases}$$

and

$$\eta_0 \leq \min\{y_r : r = 1, \ldots, M\}, \qquad \max\{y_r : r = 1, \ldots, M\} \leq \eta_b,$$

i.e., each of the values y_r of the output quantity lies in exactly one of the intervals B_k.[14] Define,[15] for $k = 1, \ldots, b$,

$$M_k = \text{card}(\{y_r \in B_k : r = 1, \ldots, M\}), \qquad h_k = \frac{M_k}{\eta_k - \eta_{k-1}}, \qquad g_k = \frac{h_k}{M}.$$

M_k is the number of values y_r in the kth bin. h_k is the height of the kth bar corresponding to the bin B_k in a histogram of the values y_r. The height is chosen so that the area of the bar is proportional to the number of values of the output quantity contained in the bin. g_k is a probability density obtained by scaling the heights of the bars so that the total area

[14] Often the bins will be chosen to have equal width, $\delta\eta$ say, where $\delta\eta = \eta_k - \eta_{k-1},\ k = 1, \ldots, b$.

[15] card(A) is used to denote the *cardinality* of the set A, i.e., the number of elements in the set.

of the bars is unity. The scaled histogram defined by the bins B_k and probability densities $g_k,\ k = 1, \ldots, b$, defines an approximation to the PDF for the output quantity as the piecewise-constant function

$$\widetilde{g}(\eta) = g_k, \qquad \eta \in B_k, \qquad k = 1, \ldots, b.$$

Table 24 specifies the evaluation of $\widetilde{g}(\eta)$, and indicates the input and output parameters necessary for its determination.

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{y}$	Values $(y_1, \ldots, y_M)$ of the output quantity corresponding to the M random draws $\boldsymbol{x}_r$ from the PDFs for the N input quantities, i.e., $y_r = f(\boldsymbol{x}_r)$ or $h(y_r, \boldsymbol{x}_r) = 0$
b	Number of bins in the approximation to the PDF for the output quantity
$\boldsymbol{\eta}$	Values $(\eta_0, \ldots, \eta_b)$, with $\eta_0 \leq \min\{y_r : r = 1, \ldots, M\}$ and $\max\{y_r : r = 1, \ldots, M\} \leq \eta_b$, that define bins $B_k,\ k = 1, \ldots, b$, where $$B_k = \begin{cases} [\eta_{k-1}, \eta_k) & k = 1, \ldots, b-1, \\ [\eta_{k-1}, \eta_k], & k = b. \end{cases}$$
Output parameter	
$\boldsymbol{g}$	Probability densities $(g_1, \ldots, g_b)$, defined by $$g_k = \frac{h_k}{M}, \qquad k = 1, \ldots, b,$$ where $$h_k = \frac{M_k}{\eta_k - \eta_{k-1}}$$ and $$M_k = \text{card}(\{y_r \in B_k : r = 1, \ldots, M\}).$$ The scaled histogram defined by bins B_k and probability densities g_k defines an approximation to the PDF for the output quantity

Table 24: Approximation to the probability density function for a univariate output quantity.

5.4.4 Discrete representation of the distribution function

A discrete representation $\boldsymbol{G}$ of the distribution function for the output quantity is obtained by sorting the values $y_r, r = 1, \ldots, M$, provided in Section 5.4.1 into non-decreasing order.

Denoting the sorted values by $y_{(r)}$, $r = 1, \ldots, M$, the discrete representation is given by $\boldsymbol{G} = (y_{(1)}, \ldots, y_{(M)})$. Table 25 specifies the evaluation of a discrete representation of the distribution function for the output quantity, and indicates the input and output parameters necessary for its determination.

The discrete representation is used as the basis for calculating a coverage interval for the output quantity (Section 5.4.6). It is also used as the basis for obtaining a (continuous) approximation to the distribution function for the output quantity (Section 5.4.5) that may be used, for example, to obtain random draws from the distribution for the output quantity (in the manner described in Section 5.3.6).

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{y}$	Values $(y_1, \ldots, y_M)$ of the output quantity corresponding to the M random draws $\boldsymbol{x}_r$ from the PDFs for the N input quantities, i.e., $y_r = f(\boldsymbol{x}_r)$ or $h(y_r, \boldsymbol{x}_r) = 0$
Output parameter	
$\boldsymbol{G}$	Discrete representation of the distribution function for the output quantity, where $\boldsymbol{G} = (y_{(1)}, \ldots, y_{(M)})$, the values of the output quantity sorted into non-decreasing order
Numerical analysis	
	It is recommended that a sorting algorithm that takes a number of operations proportional to $M \log M$ be used (e.g., [44]). A naive algorithm would take a time proportional to M^2 and may make the computation time unacceptable

Table 25: Discrete representation of the distribution function for a univariate output quantity.

5.4.5 Approximation to the distribution function

An approximation $\widetilde{G}(\eta)$ to the distribution function $G(\eta)$ for the output quantity is obtained as follows. Assign uniformly spaced cumulative probabilities $p_r = (r - 1/2)/M$, $r = 1, \ldots, M$, to the ordered values $y_{(r)}$ in the discrete representation $\boldsymbol{G}$ of the distribution function for the output quantity.[16] Form $\widetilde{G}(\eta)$ as the piecewise-linear function joining the M points $(y_{(r)}, p_r),\ r = 1, \ldots, M$:

$$\widetilde{G}(\eta) = \frac{r - 1/2}{M} + \frac{\eta - y_{(r)}}{M(y_{(r+1)} - y_{(r)})}, \qquad y_{(r)} \leq \eta \leq y_{(r+1)}, \tag{4}$$

[16]The values p_r, $r = 1, \ldots, M$, are the midpoints of M contiguous probability intervals of width $1/M$ between zero and one.

for $r = 1, \ldots, M-1$. Table 26 specifies the evaluation of an approximation to the distribution function for the output quantity, and indicates the input and output parameters necessary for its determination.

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{G}$	Discrete representation of the distribution function for the output quantity, where $\boldsymbol{G} = (y_{(1)}, \ldots, y_{(M)})$, the values of the output quantity sorted into non-decreasing order
Output parameter	
$\boldsymbol{p}$	Probabilities $(p_1, \ldots, p_M)$, defined by $$p_r = (r - 1/2)/M.$$ The function $\widetilde{G}(\eta)$ determined as the piecewise-linear function joining the points $(y_{(r)}, p_r),\ r = 1, \ldots, M$, provides an approximation to the distribution function for the output quantity. ($\widetilde{G}(\eta)$ is defined only for values of η corresponding to values of probability p in the interval $M/2 \leq p \leq 1 - M/2$. Indeed, it should not be used near the endpoints of this interval, because it is less reliable there.)

Table 26: Approximation to the distribution function for a univariate output quantity.

Formulae (2) and (3) for the estimate of the output quantity and the associated standard uncertainty do not in general provide values that are identical to the expectation and standard deviation of the quantity characterized by the distribution function $\widetilde{G}(\eta)$. The latter values are given by

$$\widetilde{y} = \frac{1}{M} {\sum_{r=1}^{M}}'' y_{(r)} \tag{5}$$

and

$$u^2(\widetilde{y}) = \frac{1}{M} \left[{\sum_{r=1}^{M}}'' (y_{(r)} - \widetilde{y})^2 - \frac{1}{6} \sum_{r=1}^{M-1} (y_{(r+1)} - y_{(r)})^2 \right], \tag{6}$$

where the double prime on the summation in expression (5) and on the first summation in expression (6) indicates that the first and the last terms are to be taken with weight one half. However, for a sufficiently large value of M, the values obtained using expressions (2) and (3) are generally indistinguishable for practical purposes from those given by expressions (5) and (6).

5.4.6 Coverage interval

Let α denote any value between zero and $1-p$, where p is the required coverage probability (e.g., 0.95). The endpoints of a $100p\,\%$ coverage interval for the value of the output quantity are $G^{-1}(\alpha)$ and $G^{-1}(p+\alpha)$, i.e., the α– and $(p+\alpha)$–quantiles of $G(\eta)$. Here, the β–quantile is the value of η for which $G(\eta) = \beta$.

The choice $\alpha = 0.025$ gives the coverage interval defined by the 0.025- and 0.975-quantiles. This choice provides a $95\,\%$ coverage interval that is probabilistically symmetric. The probability is $2.5\,\%$ that the value of the output quantity is smaller than the left-hand endpoint of the interval and $2.5\,\%$ that it is larger than the right-hand endpoint. If $g(\eta)$ is symmetric about its expectation, this coverage interval is symmetric about the estimate y of the output quantity, and the left-hand and right-hand endpoints of the coverage interval are equidistant from y.

A value of α different from 0.025 would generally be appropriate were the PDF asymmetric. Usually the shortest coverage interval is required, because it corresponds to the best possible location of the output quantity for a specified probability. It is given by the value of α satisfying $g(G^{-1}(\alpha)) = g(G^{-1}(p+\alpha))$, if $g(\eta)$ is single-peaked, and in general by the value of α such that $G^{-1}(p+\alpha) - G^{-1}(\alpha)$ is a minimum. If $g(\eta)$ is symmetric, the shortest coverage interval is given by taking $\alpha = (1-p)/2$, corresponding to the probabilistically symmetric interval.

The endpoints of a coverage interval can be obtained from the discrete representation of the distribution function for the output quantity (Section 5.4.4) as follows. Let $q = pM$, if pM is an integer, or the integer part of $pM + 1/2$, otherwise. Then, $[y_{\rm low}, y_{\rm high}] = [y_{(r)}, y_{(r+q)}]$ for any $r = 1, \ldots, M-q$, is a $100p\,\%$ coverage interval. The probabilistically symmetric $100p\,\%$ coverage interval is given by $r = (M-q)/2$ if $(M-q)/2$ is an integer, or the integer part of $(M-q+1)/2$, otherwise. The shortest $100p\,\%$ coverage interval is given by determining $r = r^*$ such that, $y_{(r^*+q)} - y_{(r^*)}$ is the minimum of $y_{(r+q)} - y_{(r)}$ for $r = 1, \ldots, M-q$. Table 27 specifies the determination of the shortest coverage interval from a discrete representation of the distribution function for the output quantity, and indicates the input and output parameters necessary for their determination.

The endpoints of a coverage interval can also be obtained from the approximation $\widetilde{G}(\eta)$ to $G(\eta)$ obtained in Section 5.4.5 or Appendix E. For a sufficiently large value of M, the coverage interval obtained using the discrete representation $\boldsymbol{G}$ of $G(\eta)$ can be expected to be indistinguishable for practical purposes from those obtained using the approximation $\widetilde{G}(\eta)$. To find the left-hand endpoint $y_{\rm low}$ such that $\alpha = \widetilde{G}(y_{\rm low})$, identify the index r for which the points $(y_{(r)}, p_r)$ and $(y_{(r+1)}, p_{r+1})$ satisfy

$$p_r \leq \alpha < p_{r+1}.$$

Then, by inverse linear interpolation,

$$y_{\rm low} = y_{(r)} + \left(y_{(r+1)} - y_{(r)}\right) \frac{\alpha - p_r}{p_{r+1} - p_r}.$$

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{G}$	Discrete representation of the distribution function for the output quantity, where $\boldsymbol{G} = (y_{(1)}, \ldots, y_{(M)})$, the values of the output quantity sorted into non-decreasing order
p	Coverage probability (e.g., 0.95)
Output parameters	
$y_{\rm low}, y_{\rm high}$	Endpoints of the shortest $100p\,\%$ coverage interval for the output quantity, given by $[y_{(r^*)}, y_{(r^*+q)}]$, where $q = pM$, if pM is an integer, or the integer part of $pM + 1/2$, otherwise, and r^*, an integer between 1 and $M - q$, is chosen so that for $r = 1, \ldots, M - q$, $$y_{(r^*+q)} - y_{(r^*)} \leq y_{(r+q)} - y_{(r)}$$

Table 27: Shortest coverage interval obtained from a discrete representation of the distribution function for a univariate output quantity.

Similarly, the upper endpoint $y_{\rm high}$ is calculated from

$$y_{\rm high} = y_{(s)} + \left(y_{(s+1)} - y_{(s)}\right) \frac{p + \alpha - p_s}{p_{s+1} - p_s},$$

where the index s is identified to satisfy

$$p_s \leq p + \alpha < p_{s+1}.$$

The shortest coverage interval can generally be obtained computationally from $\widetilde{G}(\eta)$ by determining α such that $\widetilde{G}^{-1}(p + \alpha) - \widetilde{G}^{-1}(\alpha)$ is a minimum. A straightforward approach to determining the minimum is to evaluate $\widetilde{G}^{-1}(p + \alpha) - \widetilde{G}^{-1}(\alpha)$ for a sufficient number of choices $\{\alpha_k\}$ of α between zero and $1 - p$, and to choose that value α_ℓ from the set $\{\alpha_k\}$ yielding the minimum value from the set $\{\widetilde{G}^{-1}(p + \alpha_k) - \widetilde{G}^{-1}(\alpha_k)\}$.

5.5 Monte Carlo procedure for multivariate measurement models

5.5.1 Calculation of the values of the output quantities

Denote by $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M$ the M random draws from the PDFs for the N input quantities. For a multivariate, real measurement function, the corresponding values $\boldsymbol{y}_r$ of the output quantity, which are vectors of dimension $m \times 1$, are obtained by evaluating the measurement function:

$$\boldsymbol{y}_r = \boldsymbol{f}(\boldsymbol{x}_r),\ r = 1, \ldots, M.$$

For a multivariate, real measurement model, they are determined by solving the measurement model:

$$\boldsymbol{h}(\boldsymbol{y}_r, \boldsymbol{x}_r) = \boldsymbol{0},\ r = 1, \ldots, M.$$

The values $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M$ are used in the evaluation of the estimate $\boldsymbol{y}$ of $\boldsymbol{Y}$ and the associated covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$ (Section 5.5.2), and as the basis for calculating an approximation to the PDF for $\boldsymbol{Y}$ (Section 5.5.3) and a coverage region for $\boldsymbol{Y}$ (Section 5.5.4).

An 'updating' procedure for forming $\boldsymbol{y}$, $\boldsymbol{U}_{\boldsymbol{y}}$, an approximation to the PDF for the output quantities, and a coverage region, which avoids the need to store and sort the complete set of values of the output quantities, is described in Appendix E.

5.5.2 Estimate of the output quantity and the associated covariance matrix

The average of the values $\boldsymbol{y}_r, r = 1, \ldots, M$, is taken as the estimate $\boldsymbol{y}$ of $\boldsymbol{Y}$:

$$\boldsymbol{y} = \frac{1}{M} \sum_{r=1}^{M} \boldsymbol{y}_r.$$

To evaluate the uncertainty associated with this estimate, assemble the vectors $\boldsymbol{y}_r$, into an $m \times M$ matrix:[17]

$$\boldsymbol{\Psi} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M).$$

From this matrix the covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$ associated with $\boldsymbol{y}$ is calculated from

$$\boldsymbol{U}_{\boldsymbol{y}} = \frac{1}{M-1} \boldsymbol{\Psi}'(\boldsymbol{\Psi}')^{\top},$$

where $\boldsymbol{\Psi}'$ is $\boldsymbol{\Psi}$ corrected for the means over all M trials, i.e., the mean of the elements in the jth row is subtracted from all elements in that row. Table 28 specifies the evaluations of the estimates $\boldsymbol{y}$ and the associated covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$ in terms of the vectors $\boldsymbol{y}_r$, and indicates the input and output parameters necessary for its determination.

This covariance matrix contains (generally a more reliable estimate of) the information that would be delivered by a linear analysis such as the GUM uncertainty framework. (In fact, it provides more than the GUM uncertainty framework, since that procedure does not in general cover multivariate measurement models.) The matrix $\boldsymbol{\Psi}$ provides much richer information, however, in the following sense. Any column of $\boldsymbol{\Psi}$ corresponds to the values of the output quantities for one choice (random draw) of the input quantities. *Any* (scalar) derived quantity can be determined from this single set of values. This quantity can be calculated for all columns, the resulting row vector of dimension $1 \times M$ being used to provide a discrete representation of the distribution function for that quantity (as in Section 5.4.4). In particular, the discrete representation can be used to provide a coverage interval for the

[17] The symbol $\boldsymbol{\Psi}$ is (reluctantly) used to denote the matrix of y-vectors, since Y is used to denote a scalar output quantity and $\boldsymbol{Y}$ a vector output quantity.

Input parameters	
m	Number of output quantities
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{\Psi}$	$m \times M$ matrix containing the values $\boldsymbol{y}_r$ of the output quantities, i.e., $$\boldsymbol{\Psi} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M).$$
Output parameter	
$\boldsymbol{y}$	Estimates of the output quantities $\boldsymbol{Y}$: the arithmetic mean of the values of the output quantities, defined by $$\boldsymbol{y} = \frac{1}{M} \sum_{r=1}^{M} \boldsymbol{y}_r.$$
$\boldsymbol{U}_{\boldsymbol{y}}$	Covariance matrix associated with estimates $\boldsymbol{y}$ of the output quantities $\boldsymbol{Y}$, defined by $$\frac{1}{M-1} \boldsymbol{\Psi}'(\boldsymbol{\Psi}')^{\top},$$ where $\boldsymbol{\Psi}'$ is $\boldsymbol{\Psi}$ corrected for the means

Table 28: Estimates of the output quantities and the associated covariance matrix.

derived quantity (as in Section 5.4.6). Another quantity could be so introduced and the two row vectors used to compute any statistics required (mean, median, etc.) and the pair of vectors used to estimate correlation effects. Thus, the matrix $\boldsymbol{\Psi}$ is a very valuable array, being the basis of almost unlimited statistical information about $\boldsymbol{Y}$.

5.5.3 Approximation to the probability density function

The construction of an approximation $\widetilde{g}(\boldsymbol{\eta})$ to the PDF $g(\boldsymbol{\eta})$ for a bivariate output quantity, i.e., for $m = 2$, is described. The construction is a natural extension of that described in Section 5.4.3 for a univariate output quantity. Its generalization for output quantities with $m > 2$ is straightforward.

The values $\boldsymbol{y}_r,\ r = 1, \ldots, M$, when assembled into a (two-dimensional) histogram with suitable rectangular bins, form a frequency distribution that, when normalized to have unit volume, constitutes $\widetilde{g}(\boldsymbol{\eta})$. Calculations are not generally carried out in terms of this histogram, the resolution of which depends on the choice of bin size. However, visualization of the histogram, for example as a surface or contour plot, can be useful as an aid to understanding the nature of the PDF.

One way of obtaining $\widetilde{g}(\boldsymbol{\eta})$ is as follows. For $j = 1, 2$, let $[\eta_{j,0}, \eta_{j,b_j}]$ be an interval of values of the jth output quantity partitioned into b_j subintervals $\{B_{k_j} : k_j = 1, \ldots, b_j\}$, where

$$B_{k_j} = \begin{cases} [\eta_{j,k_j-1}, \eta_{k_j}), & k_j = 1, \ldots, b_j - 1, \\ [\eta_{j,k_j-1}, \eta_{k_j}], & k_j = b_j, \end{cases}$$

and

$$\eta_{j,0} \leq \min\{y_{j,r} : r = 1, \ldots, M\}, \qquad \max\{y_{j,r} : r = 1, \ldots, M\} \leq \eta_{b_j},$$

i.e., each of the values $y_{j,r}$ of the jth output quantity lies in exactly one of the intervals B_{k_j}.[18]

Let B_{k_1,k_2} denote the rectangular bin

$$B_{k_1,k_2} = B_{k_1} \times B_{k_2},$$

composed of the points $\boldsymbol{\eta} = (\eta_1, \eta_2)^\top$ such that $\eta_1 \in B_{k_1}$ and $\eta_2 \in B_{k_2}$. Define, for $k_1 = 1, \ldots, b_1$, and $k_2 = 1, \ldots, b_2$,

$$\begin{aligned} M_{k_1,k_2} &= \text{card}(\{\boldsymbol{y}_r \in B_{k_1,k_2} : r = 1, \ldots, M\}), \\ h_{k_1,k_2} &= \frac{M_{k_1,k_2}}{(\eta_{1,k_1} - \eta_{1,k_1-1})\,(\eta_{2,k_2} - \eta_{2,k_2-1})}, \\ g_{k_1,k_2} &= \frac{h_{k_1,k_2}}{M}. \end{aligned}$$

[18] Often the bins will be chosen to have equal width, $\delta\eta_j$ say, where $\delta\eta_j = \eta_{j,k_j} - \eta_{j,k_j-1},\ k_j = 1, \ldots, b_j$.

M_{k_1,k_2} is the number of values $\boldsymbol{y}_r$ in the rectangular bin B_{k_1,k_2}. h_{k_1,k_2} is the height of the cuboid corresponding to the rectangular bin B_{k_1,k_2} in a histogram of the values $\boldsymbol{y}_r$. The height is chosen so that the volume of the cuboid is proportional to the number of values of the output quantity contained in the rectangular bin. g_{k_1,k_2} is a probability density obtained by scaling the heights of the cuboids so that the total volume of the cuboids is unity. The scaled histogram defined by the rectangular bins B_{k_1,k_2} and probability densities $g_{k_1,k_2},\ k_1 = 1,\ldots,b_1,\ k_2 = 1,\ldots,b_2$, defines an approximation to the PDF for the output quantity as the piecewise constant function

$$\widetilde{g}(\boldsymbol{\eta}) = g_{k_1,k_2}, \qquad \boldsymbol{\eta} \in B_{k_1,k_2}, \qquad k_1 = 1,\ldots,b_1, \qquad k_2 = 1,\ldots,b_2.$$

Table 29 specifies the evaluation of $\widetilde{g}(\boldsymbol{\eta})$, and indicates the input and output parameters necessary for its determination.

5.5.4 Coverage region

The provision of coverage regions for multivariate output quantities is not straightforward, because the operation of sorting multivariate data is generally not well-defined. Even in the univariate case, the choice of coverage interval is not unique, and there is far greater freedom of choice in the multivariate case, where any domain containing $95\,\%$ of the distribution of possible values constitues a $95\,\%$ coverage region.

The construction of coverage regions of a particular form, viz., an ellipsoid in m-dimensions, is described. For linear or linearized problems, the covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$ associated with the estimates $\boldsymbol{y}$ of $\boldsymbol{Y}$ defines a one-standard-deviation ellipsoid centred on $\boldsymbol{y}$. Ellipsoids concentric with this one contain various fractions of the distribution of values of $\boldsymbol{Y}$. For a given coverage probability p, the size of the ellipsoid can be found, using the theory of multidimensional Gaussian distributions, that contains $100p\,\%$ of the possible values of $\boldsymbol{Y}$ (see Section 4.5). Such an ellipsoid can be constructed from $\boldsymbol{U}_{\boldsymbol{y}}$, but its size would depend on the Gaussian assumption and not on the actual distribution of $\boldsymbol{Y}$. An ellipsoid is required that contains $100p\,\%$ of the actual distribution. Since the values $\boldsymbol{y}_r$ can be expected to reflect faithfully the distribution of $\boldsymbol{Y}$, a coverage region is constructed as the ellipsoid that (just) contains $100p\,\%$ of these $\boldsymbol{y}_r$. The steps in the construction are as follows.

Form the Cholesky factor $\boldsymbol{L}$ of the covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$, i.e., the lower-triangular matrix $\boldsymbol{L}$ that satisfies

$$\boldsymbol{U}_{\boldsymbol{y}} = \boldsymbol{L}\boldsymbol{L}^{\top}.$$

Transform the points $\boldsymbol{y}_r$ to give

$$\dot{\boldsymbol{y}}_r = \boldsymbol{L}^{-1}\left(\boldsymbol{y}_r - \boldsymbol{y}\right), \qquad r = 1,\ldots,M.$$

Sort the transformed points $\dot{\boldsymbol{y}}_r$ according to increasing value of d_r, where

$$d_r^2 = \dot{\boldsymbol{y}}_r^{\top}\dot{\boldsymbol{y}}_r = \sum_{j=1}^{m} \dot{y}_{j,r}^2, \qquad r = 1,\ldots,M.$$

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{\Psi}$	$2 \times M$ matrix containing the values $\boldsymbol{y}_r$ of the bivariate output quantity, i.e., $$\boldsymbol{\Psi} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M),$$ where $\boldsymbol{y}_r = (y_{1,r}, y_{2,r})^\top$
b_1, b_2	For $j = 1, 2$, number of intervals for the jth output quantity in the approximation to the PDF for the bivariate output quantity
$\boldsymbol{\eta}_1, \boldsymbol{\eta}_2$	For $j = 1, 2$, values $\boldsymbol{\eta}_j = (\eta_{j,0}, \ldots, \eta_{j,b_j})$, with $$\eta_{j,0} \leq \min\{y_{j,r} : r = 1, \ldots, M\},$$ and $$\max\{y_{j,r} : r = 1, \ldots, M\} \leq \eta_{j,b_j},$$ that define intervals $B_{k_j},\ k_j = 1, \ldots, b_j$, where $$B_{k_j} = \begin{cases} [\eta_{j,k_j-1}, \eta_{j,k_j}) & k_j = 1, \ldots, b_j - 1, \\ [\eta_{j,k_j-1}, \eta_{j,k_j}], & k_j = b_j. \end{cases}$$

Output parameter	
$\boldsymbol{g}$	Probability densities g_{k_1,k_2}, defined by $$g_{k_1,k_2} = \frac{h_{k_1,k_2}}{M}, \qquad k_1 = 1, \ldots, b_1, \qquad k_2 = 1, \ldots, b_2,$$ where $$h_{k_1,k_2} = \frac{M_{k_1,k_2}}{(\eta_{1,k_1} - \eta_{1,k_1-1})\,(\eta_{2,k_2} - \eta_{2,k_2-1})}$$ and $$M_{k_1,k_2} = \text{card}(\{\boldsymbol{y}_r \in B_{k_1,k_2} : r = 1, \ldots, M\}).$$ The scaled histogram defined by rectangular bins $B_{k_1,k_2} = B_{k_1} \times B_{k_2}$ and probability densities g_{k_1,k_2} defines an approximation to the PDF for the bivariate output quantity

Table 29: Approximation to the probability density function for a bivariate output quantity.

Using the sorted $\dot{\boldsymbol{y}}_r$, determine the coverage factor k_p such that a fraction p of the $\dot{\boldsymbol{y}}_r$ satisfies $d_r < k_p$, i.e., $k_p = d_{(r^*)}$ where r^* is the integer part of pM and $d_{(r)},\ r = 1, \ldots, M$, are the values d_r sorted into non-decreasing order. The boundary of a $100p\,\%$ coverage region for $\boldsymbol{Y}$ is then defined by the ellipsoid

$$(\boldsymbol{\eta} - \boldsymbol{y})^{\top} \boldsymbol{U}_{\boldsymbol{y}}{}^{-1} (\boldsymbol{\eta} - \boldsymbol{y}) = k_p^2.$$

5.6 Sensitivity analysis

In the application of a Monte Carlo method there is no immediate counterpart of a sensitivity coefficient (Section 4.2) since the Monte Carlo procedure operates in terms of the actual non-linear measurement model rather than a linearized counterpart. Recall that with a linear measurement model the sensitivity coefficients 'reproduce' linear effects, and for a non-linear model the sensitivity coefficients provide first-order information. Therefore, those practitioners accustomed to the GUM uncertainty framework may find the absence of sensitivity coefficients disconcerting.

It is possible and straightforward, however, to adapt the Monte Carlo procedure such that it provides information that in a sense constitutes a non-linear counterpart of a sensitivity coefficient. Consider holding all input quantities but one, say X_k, at their estimates. In this setting the measurement model effectively becomes one having a single input quantity, viz., X_k. Draw values randomly from the PDF for this input quantity, taking the standard deviation $u_k(y)$ of the corresponding values of the output quantity as an approximation to the component of the standard uncertainty $u(y)$ corresponding to X_k.

A 'non-linear' sensitivity coefficient $\widetilde{c}_k$ is defined by

$$\widetilde{c}_k = \frac{u_k(y)}{u(x_k)}.$$

It will be equal to the magnitude $|c_k|$ of the 'linear' sensitivity coefficient c_k when the measurement model is linear in X_k, and be close to its value when the non-linearity with respect to X_k is negligible. When $\widetilde{c}_k$ is appreciably different from $|c_k|$ the non-linearity effect may noticeably influence the standard uncertainty $u(y)$. Thus, the deviation of $\widetilde{c}_k$ from $|c_k|$ can be used as an approximate measure of the influence of measurement model non-linearity with regards to X_k alone.

The sensitivity coefficients so obtained are not generally to be taken in conjunction with the standard uncertainties associated with the estimates of the input quantities as the only contributions to the standard uncertainty associated with the estimate of the output quantity. There will be further contributions arising from any interaction (i.e., non-additive) terms in the measurement model.

Table 30 specifies the calculation of a non-linear sensitivity coefficient relating to the input quantity X_k using the Monte Carlo procedure.

Input parameters	
M	Number of random draws (equal to the number of Monte Carlo trials)
$\boldsymbol{y}^k$	Values $(y_1^k, \ldots, y_M^k)$ of the output quantity corresponding to the M random draws $\boldsymbol{x}_r^k$ obtained by holding all input quantities but one, X_k, at their estimates and making random draws from the PDF $g_k(\xi_k)$ for X_k
$u(x_k)$	Standard uncertainty associated with the estimate x_k of X_k
Output parameters	
$u_k(y)$	Component of the standard uncertainty $u(y)$: the standard deviation of the values $(y_1^k, \ldots, y_M^k)$, defined by $$u_k^2(y) = \frac{1}{M-1}\sum_{r=1}^{M}(y_r^k - y^k)^2$$ where $$y^k = \frac{1}{M}\sum_{r=1}^{M} y_r^k.$$
$\widetilde{c}_k$	Non-linear sensitivity coefficient, defined by $$\widetilde{c}_k = \frac{u_k(y)}{u(x_k)}.$$

Table 30: Sensitivity analysis using a Monte Carlo procedure.

For a multivariate output quantity $\boldsymbol{Y}$, the analysis described above can be straightforwardly extended. All input quantities but one, say X_k, are held at their estimates. Draw values randomly from the PDF for this input quantity and determine the estimates of the output quantities and the associated covariance matrix. Calculate the square roots $u_k(y_j),\ j = 1, \ldots, m$, of the diagonal elements of the covariance matrix. The standard deviation $u_k(y_j)$ is taken as an approximation to the component of the standard uncertainty $u(y_j)$ associated with the estimate y_j of Y_j corresponding to X_k. The 'non-linear' sensitivity coefficients $\widetilde{c}_{j,k}$ are defined by

$$\widetilde{c}_{j,k} = \frac{u_k(y_j)}{u(x_k)}, \qquad j = 1, \ldots, m, \qquad k = 1, \ldots, N.$$

$\widetilde{c}_{j,k}$ will be equal to the magnitude $|c_{j,k}|$ in the case that the measurement model is linear in X_k, where $c_{j,k}$ is the partial derivative of first order of Y_j with respect to X_k evaluated at $X_i = x_i,\ i = 1, \ldots, N$.

5.7 Adaptive Monte Carlo procedure

A basic implementation of an adaptive Monte Carlo procedure is described as follows. It is based on carrying out an increasing number of Monte Carlo trials until the various parameters of interest have stabilized in a statistical sense. The parameters of interest include the estimate y of the output quantity Y, the standard uncertainty $u(y)$ associated with y, and the endpoints $y_{\rm low}$ and $y_{\rm high}$ of a $95\,\%$ coverage interval for Y.[19] A parameter is deemed to have stabilized if twice the standard deviation associated with the estimate of the parameter is less than a numerical tolerance used to assess the 'degree of approximation' required in the standard uncertainty $u(y)$, and corresponds to that given by expressing $u(y)$ to what is regarded as a meaningful number of siginficant decimal digits (see Section 6 and Table 31).

A practical approach consists of carrying out a sequence of Monte Carlo calculations, each containing a relatively small number, say $M_{\rm adap} = 10^4$, trials.[20] For each Monte Carlo calculation in the sequence, y, $u(y)$ and the endpoints of a $95\,\%$ coverage interval are formed from the results obtained, as in Sections 5.4.2 and 5.4.6. Denote by $y^{(h)}$, $u(y^{(h)})$, $y^{(h)}_{\rm low}$ and $y^{(h)}_{\rm high}$ the values of y, $u(y)$ and the left- and right-hand endpoints of the $95\,\%$ coverage interval for the hth member of the sequence.

After the hth Monte Carlo calculation (apart from the first) in the sequence, the arithmetic mean of the values $y^{(1)}, \ldots, y^{(h)}$ and the standard deviation s_y associated with this arithmetic mean are formed. The counterparts of these statistics for y are determined

[19] When a coverage interval is not required, the parameters of interest would include only the estimate y and the associated standard uncertainty $u(y)$.

[20] It is recommended that each sequence of calculations be performed using the same random number generator (albeit using a different part of the generated sequence) or two or more independent generators. This is because the statistical properties of a random number generator are defined 'within' sequences and not 'between' sequences (as, for example, two sequences obtained from the same number generator may contain a common sub-sequence even if initialized with different seeds).

for $u(y)$, y_{low} and y_{high}. If the largest of $2s_y$, $2s_{u(y)}$, $2s_{y_{\mathrm{low}}}$ and $2s_{y_{\mathrm{high}}}$ does not exceed the numerical tolerance used to assess the degree of approximation required in $u(y)$, the overall computation is regarded as having stabilized. The results from the total number of Monte Carlo trials taken are then used to provide the estimate of the output quantity, the associated standard uncertainty and the coverage interval for the output quantity.

For a multivariate output quantity, the procedure described above can be straightforwardly extended. It is (also) based on carrying out an increasing number of Monte Carlo trials until the various parameters of interest have stabilized in a statistical sense. The parameters of interest can include[21]

- the estimates $y_j,\ j = 1, \ldots, m$, of the output quantities,
- the standard uncertainties $u(y_j),\ j = 1, \ldots, m$, associated with the estimates,
- the correlation coefficients $r(y_l, y_j),\ l = 1, \ldots, m,\ j = 1, \ldots, m$, associated with pairs of estimates,
- the 'size' k_p of the $95\,\%$ covering ellipsoid, and
- any other quantity determined from the values of the output quantity.

A parameter is (again) deemed to have stabilized if twice the standard deviation associated with the estimate of the parameter is less than a numerical tolerance used to assess the degree of approximation required in the parameter. For y_j and $u(y_j)$, the numerical tolerance is calculated in terms of a number of significant decimal digits regarded as meaningful in the value of $u(y_j)$. For $r(y_l, y_j)$, it is calculated in terms of a number of significant decimal digits regarded as meaningful in the value unity.[22] For k_p, it is calculated in terms of a number of significant decimal digits regarded as meaningful in the value of k_p.

A sequence of Monte Carlo calculations is carried out until the parameters of interest have stabilized. The results from the total number of Monte Carlo trials taken are then used to provide estimates of the output quantities, the associated covariance matrix and a coverage region for the output quantity.

6 Validation of the GUM uncertainty framework

The GUM uncertainty framework has some limitations [3, 14]. Although the procedure can be expected to work well in many circumstances, it is generally difficult to quantify the

[21] When a coverage region is not required, the parameters of interest would include only the estimates y_j, associated standard uncertainties $u(y_j)$ and correlation coefficients $r(y_l, y_j)$. The standard uncertainties and correlation coefficients together define the covariance matrix $\boldsymbol{U_y}$ associated with the estimates of the output quantities.

[22] The motivation for this choice is that the correlation coefficients $r(y_l, y_j)$ are normalized (and dimensionless) quantities.

effects of the approximations involved, viz., linearization, the Welch-Satterthwaite formula for the effective degrees of freedom and the assumption that the output quantity is Gaussian (i.e., that the Central Limit Theorem is applicable). Indeed, the degree of difficulty of doing so would typically be considerably greater than that required to apply a Monte Carlo method. Therefore, since these circumstances cannot readily be tested, any cases of doubt should be validated. To this end, since a Monte Carlo method is a more general approach to uncertainty evaluation, it is recommended that both the GUM uncertainty framework and the Monte Carlo approach are applied and the results compared. If the comparison is favourable, the GUM uncertainty framework can be used on this occasion and for sufficiently similar problems in the future. Otherwise, consideration can be given to using a Monte Carlo method instead.

Specifically, for a univariate measurement model, it is recommended that the two steps below and the following comparison process are carried out.

1. Apply the GUM uncertainty framework to yield a $95\,\%$ coverage interval $y \pm U(y)$ for the output quantity.

2. Apply a Monte Carlo method to yield the standard uncertainty $u(y)$ associated with an estimate of the output quantity and the endpoints y_{low} and y_{high} of a $95\,\%$ coverage interval for the output quantity.

A comparison procedure is based on the following objective: determine whether the coverage intervals obtained by the GUM uncertainty framework and a Monte Carlo method agree to a stipulated degree of approximation.[23] This degree of approximation is assessed in terms of the endpoints of the coverage intervals and corresponds to that given by expressing the standard uncertainty $u(y)$ to what is regarded as a meaningful number of *significant decimal digits*.

The procedure is as follows:

1. Let n_{ndig} denote the number of significant decimal digits regarded as meaningful in the numerical value of $u(y)$. Usually, $n_{\mathrm{ndig}} = 1$ or $n_{\mathrm{ndig}} = 2$. Express the value of $u(y)$ in the form $a \times 10^r$, where a is an n_{ndig}-digit integer and r an integer. The numerical tolerance for assessing the degree of approximation is

$$\delta = 0.5 \times 10^r.$$

   Table 31 specifies the calculation of the numerical tolerance for assessing the degree of approximation of a parameter in an adaptive application of a Monte Carlo method and for validating a parameter from the GUM uncertainty framework.

[23] When a coverage interval is not required, the comparison can be undertaken in terms of the estimates and associated standard uncertainties obtained by the GUM uncertainty framework and a Monte Carlo method.

2. Compare the coverage intervals obtained by the GUM uncertainty framework and a Monte Carlo method to determine whether the required number of correct digits in the coverage interval provided by the GUM uncertainty framework has been obtained. Specifically, determine the values

$$|y - U(y) - y_{\text{low}}|$$

and

$$|y + U(y) - y_{\text{high}}|,$$

viz., the absolute differences of the respective endpoints of the two coverage intervals. Then, if both these values are no larger than δ the comparison is favourable and the GUM uncertainty framework has been validated in this instance.

Example of calculating the numerical tolerance δ. The estimate of the output quantity for a nominally 100 g standard of mass [3, Clause 7.2.2] is $y = 100.021\,47\,\text{g}$. The standard uncertainty $u(y) = 0.000\,35\,\text{g}$. Thus, $n_{\text{ndig}} = 2$ and $u(y)$ is expressed as $35 \times 10^{-5}\,\text{g}$, and so $a = 35$ and $r = -5$. Take $\delta = 0.5 \times 10^{-5}\,\text{g} = 0.000\,005\,\text{g}$.

Input parameters	
n_{dig}	Number of significant decimal digits regarded as meaningful in the numerical value of $u(y)$
z	Numerical value of a parameter, e.g., the standard uncertainty $u(y)$ associated with the estimate y of the output quantity
Output parameter	
δ	Numerical tolerance for assessing the degree of approximation of a parameter in an adaptive application of a Monte Carlo method and for validating a parameter from the GUM uncertainty framework, given by $\delta = 0.5 \times 10^{r},$ where z is expressed in the form $a \times 10^{r}$ in which a is an n_{ndig}-digit integer and r an integer

Table 31: Numerical tolerance for assessing the degree of approximation of a parameter in an adaptive application of a Monte Carlo method and for validating a parameter from the GUM uncertainty framework.

For a multivariate output quantity, the procedure described above can be straightforwardly extended. The objective is to determine whether the parameters of interest obtained by the GUM uncertainty framework and a Monte Carlo method agree to a stipulated degree of approximation. The parameters of interest can include the estimates y_j of the output quantities, the standard uncertainties $u(y_j)$ associated with the estimates, the correlation

coefficients $r(y_l, y_j)$ associated with pairs of estimates, and the size k_p of a $95\,\%$ covering ellipsoid[24] (cf. Section 5.7). For y_j and $u(y_j)$ a numerical tolerance for assessing the degree of approximation is calculated in terms of a number of significant decimal digits regarded as meaningful in the value $u(y_j)$, for $r(y_l, y_j)$ it is calculated in terms of a number of significant decimal digits regarded as meaningful in the value unity, and for k_p in terms of the number of significant decimal digits regarded as meaningful in the value of k_p (cf. Section 5.7). If the comparison is favourable for all parameters of interest, the GUM uncertainty framework has been validated in this instance.

7 Conclusions

Software specifications have been given for three aspects of uncertainty evaluation, viz.,

1. The GUM uncertainty framework [3];
2. A Monte Carlo method [4, 14] as an implementation of the propagation of distributions;
3. Validation of the GUM uncertainty framework using a Monte Carlo method [4, 14].

The specifications are not intended to be mandatory but indicative of the software units that are required for implementation of the above aspects.

The fourth edition of this document has been produced in the *Software Support for Metrology* (SS*f*M) programme 2007 to 2010. It is anticipated that future editions, extending and revising the document further and keeping it in line with the evolving SS*f*M best-practice guide [14], will be produced in subsequent SS*f*M programmes.

Software implementing the specifications described in this report has been developed at the National Physical Laboratory to support the use of the GUM [3] and the first supporting document to the GUM, GUM Supplement 1 [4], concerned with the use of a Monte Carlo method for measurement uncertainty evaluation. The software is available[25] in two forms.

Firstly, software is available as stand-alone executables that enable users to apply the approaches to uncertainty evaluation described in these documents to the four example problems considered in GUM Supplement 1 [15]. The software is intended to allow users to reproduce the results presented in tables and figures contained within GUM Supplement 1. It is also intended to help users learn about the methods for uncertainty evaluation described in the GUM and GUM Supplement 1 by enabling them to experiment with (a)

[24] When a coverage region is not required, the parameters of interest would include only the estimates y_j, associated standard uncertainties $u(y_j)$ and correlation coefficients $r(y_l, y_j)$.

[25] `www.npl.co.uk/mathematics-scientific-computing/software-support-for-metrology/software-downloads-(ssfm)`

different information about the input quantities in the measurement models defining the example problems, and (b) different values for the parameters controlling the application of those methods, e.g., the number of trials in an application of a Monte Carlo method.

Secondly, software is available as source code, written in the MATLAB programming language [38], for applying the GUM uncertainty framework and a Monte Carlo method to a univariate, real measurement function with a general number of uncorrelated input quantities, and validating the results from the GUM uncertainty framework using those provided by a Monte Carlo method. In the future it is planned to make available software to treat correlated input quantities and covering multivariate, real measurement functions.

Acknowledgements

The work described here was supported by the National Measurement Office of the UK Department of Business, Innovation and Skills as part of its NMS *Software Support for Metrology* programme. The authors thank Martyn Byng (Numerical Algorithms Group Ltd.) for comments on previous editions of this report.

References

[1] P. Mac Berthouex and L. C. Brown. *Statistics for Environmental Engineers*. CRC Press, USA, 1994.

[2] W. Bich, M. G. Cox, and P. M. Harris. Evolution of the "Guide to the Expression of Uncertainty in Measurement". *Metrologia*, 43(4):S161–S166, 2006.

[3] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP and OIML. *Guide to the expression of uncertainty in measurement* (GUM:1995 with minor corrections). Bureau International des Poids et Mesures, JCGM 100:2008.

[4] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP and OIML. *Evaluation of measurement data, Supplement 1 to the 'Guide to the expression of uncertainty in measurement' — Propagation of distributions using a Monte Carlo method*. Bureau International des Poids et Mesures, JCGM 101:2008.

[5] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP and OIML. *International vocabulary of metrology — Basic and general concepts and associated terms (VIM)*. Bureau International des Poids et Mesures, JCGM 200:2008.

[6] C. Bischof, A. Carle, P. Hovland, P. Khademi, and A. Mauer. *ADIFOR 2.0 Users' Guide*, 1998.

[7] C. Bischof, L. Roh, and A. Mauer. *ADIC – An extensible automatic differentiation tool for ANSI-C*. ANL/MCS-P626-1196.

[8] R. Boudjemaa, M. G. Cox, A. B. Forbes, and P. M. Harris. Automatic differentiation techniques and their application in metrology. Technical Report CMSC 26/03, National Physical Laboratory, Teddington, UK, 2003.

[9] G. E. P. Box and M. Muller. A note on the generation of random normal variates. *Ann. Math. Statist.*, 29:610–611, 1958.

[10] M. G. Cox, M. P. Dainton, and P. M. Harris. Testing spreadsheets and other packages used in metrology. Testing functions for the calculation of standard deviation. Technical Report CMSC 07/00, National Physical Laboratory, Teddington, UK, 2000.

[11] M. G. Cox, M. P. Dainton, and P. M. Harris. Software specifications for uncertainty evaluation and associated statistical analysis. Technical Report CMSC 10/01, National Physical Laboratory, Teddington, UK, 2001.

[12] M. G. Cox and P. M. Harris. Software specifications for uncertainty evaluation. Technical Report CMSC 40/04, National Physical Laboratory, Teddington, UK, 2004.

[13] M. G. Cox and P. M. Harris. Software specifications for uncertainty evaluation. Technical Report DEM-ES-010, National Physical Laboratory, Teddington, UK, 2006.

[14] M. G. Cox and P. M. Harris. SS*f*M Best Practice Guide No. 6, Uncertainty evaluation. Technical Report MS 6, National Physical Laboratory, Teddington, UK, 2010.

[15] M. G. Cox, P. M. Harris and I. M. Smith. Software for GUM Supplement 1: User Manual. Technical Report MS 1, National Physical Laboratory, Teddington, UK, 2008.

[16] T. J. Dekker. Finding a zero by means of successive linear interpolation. In B. Dejon and P. Henrici, editors, *Constructive Aspects of the Fundamental Theorem of Algebra*, London, 1969. Wiley Interscience.

[17] L. Devroye. *Non-Uniform Random Number Generation*. Springer, New York, 1986.

[18] EA. Expression of the uncertainty of measurement in calibration. Technical Report EA-4/02, European Co-operation for Accreditation, 1999.

[19] M. Evans, N. Hastings, and B. Peacock. *Statistical distributions*. Wiley, 2000.

[20] C. Faure. Automatic differentiation for adjoint code generation: An introduction to automatic adjoint code generation. Technical Report 3555, INRIA, 1998.

[21] C. Faure and Y. Papegay. Odyssée version 1.6: The user's reference manual. Technical report, INRIA, 1997.

[22] G. S. Fisher. *Monte Carlo*. Springer-Verlag, New York, 1996.

[23] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Academic Press, London, 1981.

[24] G. H. Golub and C. F. Van Loan. *Matrix Computations*. John Hopkins University Press, Baltimore, MD, USA, 1996. Third edition.

[25] A. Griewank and G. F. Corliss, editors. *Automatic Differentiation of Algorithms: Theory, Implementation and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, 1991.

[26] A. Griewank, D. Juedes, H. Mitev, J. Utke, O. Vogel, and A. Walther. ADOL-C: A package for the automatic differentiation of algorithms written in C/C++. *ACM TOMS*, 22(2):131–167, 1996.

[27] N. J. Higham. The accuracy of floating point summation. *SIAM J. Sci. Comput.*, 14(4):783–799, 1993.

[28] W. Kahan. Further remarks on removing truncation errors. *Comm. ACM*, 8, 1995.

[29] A. Kinderman, J. Monahan, and J. Ramage. Computer methods for sampling from Student's t-distribution. *Math. Comput*, 31:1009–1018, 1977.

[30] D. E. Knuth. *The Art of Computer Programming, Vol. 2: Semi-numerical algorithms*. Addison-Wesley, Reading, Ma., USA, 1981.

[31] P. L'Ecuyer and R. Simard. TestU01: A software library in ANSI C for empirical testing of random number generators. www.iro.umontreal.ca/ simardr/testu01/tu01.html.

[32] J. Leydold. Automatic sampling with the ratio-of-uniforms method. *ACM Trans. Math. Software*, 26:78–98, 2000.

[33] J. N. Lyness and C. B. Moler. Numerical differentiation of analytic functions. *SIAM J. Numer. Anal.*, 4:202–210, 1967.

[34] K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis*. Academic Press, London, 1979.

[35] G. Marsaglia. A current view of random number generators. In *Computer Science and Statistics: The Interface*, Amsterdam, 1985. Elsevier.

[36] G. Marsaglia and A. Zaman. The KISS generator. Technical report, Department of Statistics, University of Florida, USA, 1993.

[37] J. R. R. A. Martins, P. Sturdza, and J. J. Alonso. The complex-step derivative approximation. *ACM Trans. Math. Software*, 29:245–262, 2003.

[38] MATLAB. The Mathworks, Inc. www.mathworks.co.uk/products/matlab/.

[39] M. McCoy. A review on random number generation and methods for generating normal distributions. Technical Report 113/2000, National Engineering Laboratory, UK, 2000.

[40] B. D. Ripley. *Stochastic Simulation*. J. Wiley, New York, 1987.

[41] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, New York, 1999.

[42] R. Y. Rubenstein. *Simulation and the Monte Carlo Method*. J. Wiley, New York, 1981.

[43] M. J. Salter, N. M. Ridler, and M. G. Cox. Distribution of correlation coefficient for samples taken from a bivariate normal distribution. Technical Report CETM 22, National Physical Laboratory, Teddington, UK, 2000.

[44] R. S. Scowen. QUICKERSORT, Algorithm 271. *Comm. ACM*, 8:669, 1965.

[45] D. Shiriaev and A. Griewank. *Computational Differentiation: Techniques, Applications, and Tools*, chapter ADOL-F: Automatic differentiation of Fortran codes. SIAM, 1996.

[46] W. Squire and G. Trapp. Using complex variables to estimate derivatives of real functions. *SIAM Rev.*, 40:110–112, 1998.

[47] G. Strang and K. Borre. *Linear Algebra, Geodesy and GPS*. Wiley, Wellesley-Cambridge Press, 1997.

[48] B. N. Taylor and C. E. Kuyatt. Guidelines for evaluating and expressing the uncertainty of NIST measurement results. Technical Report TN1297, National Institute of Standards and Technology, USA, 1994.

[49] UKAS. The expression of uncertainty and confidence in measurement. Technical Report M 3003, United Kingdom Accreditation Service, 1997.

[50] B. A. Wichmann and I. D. Hill. Generating good pseudo-random numbers. *Computational Statistics and Data Analysis*, 51:1614–1622, 2006.

[51] G. Wübbeler, P. M. Harris, M. G. Cox, and C. Elster. A two-stage procedure for determining the number of trials in the application of a Monte Carlo method for uncertainty evaluation. *Metrologia*, 2010. Submitted.

A Use of symbolic-algebra packages

This appendix is concerned with the use of a symbolic-algebra package to provide partial derivatives of first order of a measurement function, from which sensitivity coefficients can be determined (Section 4.2). Table 32 specifies for a univariate, real measurement function, the use of a symbolic-algebra package to obtain sensitivity coefficients and indicates the input and output parameters associated with their determination. The input parameter f is used to specify the measurement function. Counterparts would apply for the other categories of measurement model.

Input parameters	
N	Number of input quantities
f	Function specifying the measurement function $Y = f(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$. Some packages may not be able to make use of $\boldsymbol{x}$ in that they return algebraic expressions for $(\partial f/\partial X_1, \ldots, \partial f/\partial X_N)$ rather than these expressions evaluated at $\boldsymbol{X} = \boldsymbol{x}$
Output parameters	
$\boldsymbol{f'}$	Functions $(f'_1, \ldots, f'_N)$ representing algebraic expressions for $(\partial f/\partial X_1, \ldots, \partial f/\partial X_N)$, the partial derivatives of first order of the function f
$\boldsymbol{C}$	$1 \times N$ vector of sensitivity coefficients, whose jth element is the partial derivative $\partial f/\partial X_j$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$. Some symbolic-algebra packages may produce just $\boldsymbol{f'}$ and it will be the user's responsibility to evaluate $\boldsymbol{f'}$ at $\boldsymbol{X} = \boldsymbol{x}$

Table 32: Sensitivity coefficients obtained using a symbolic-algebra package.

B Use of finite-difference formulae

This appendix is concerned with the use of finite-difference methods to evaluate the partial derivatives of first order of a measurement function, from which sensitivity coefficients can be determined (Section 4.2).

Numerical approximations to the values of the partial derivatives of first order can be obtained using finite-difference methods. Given a value i ($1 \leq i \leq N$), set all $X_k = x_k$, apart from X_i, i.e., hold all input quantities, apart from the ith, at their estimates. Denote the resulting function of X_i by $f_i(X_i)$.

A typical finite-difference approximation to $\partial f/\partial X_i$ evaluated at $\boldsymbol{x}$ is

$$\left.\frac{\partial f}{\partial X_i}\right|_{\boldsymbol{X}=\boldsymbol{x}} \approx \frac{f_i(x_i+\delta_i) - f_i(x_i)}{\delta_i},$$

where δ_i is a 'suitably small' increment in x_i (see below). Note that $f_i(x_i) \equiv f(\boldsymbol{x})$ will already have been formed in evaluating the measurement function at the estimates $\boldsymbol{x}$ of the input quantities.

The approximation can be perceived as follows. Consider the graph of $f_i(X_i)$. The formula gives the gradient of the chord joining the points $(x_i,\ f_i(x_i))$ and $(x_i+\delta_i,\ f_i(x_i+\delta_i))$. This gradient approximates the gradient of the tangent at $(x_i,\ f_i(x_i))$ to the graph of the function, which is of course the required derivative.

The choice of δ_i is important. If it is too great, the formula gives a large approximation error, i.e., the tangent and the chord point in appreciably different directions. If it is too small, the formula gives a large subtractive cancellation error, since the values of $f_i(x_i)$ and $f_i(x_i+\delta_i)$ will have many common leading digits.

A generally more accurate form, requiring an additional function evaluation, is

$$\left.\frac{\partial f}{\partial X_i}\right|_{\boldsymbol{X}=\boldsymbol{x}} \approx \frac{f_i(x_i+\delta_i) - f_i(x_i-\delta_i)}{2\delta_i}.$$

For a given value of δ_i, the magnitude of the approximation error is often reduced using this form. Thus the value of δ_i can be larger, affording a better balance between approximation and cancellation errors.

The GUM, in Clause 5.1.3, suggests the use of the second formula with $\delta_i = u(x_i)$. This choice can generally be expected to be acceptable, although there may be circumstances when it is not.[26]

Table 33 specifies, for a univariate, real measurement function, the use of a finite-difference formula to obtain sensitivity coefficients, and indicates the input and output parameters associated with their determination. The input parameter f is used to provide information

[26] For example, in cases where $u(x_i)$ is large and the non-linearity of f as a function of X_i is appreciable.

about the measurement function, and may take the form of a function for evaluating the measurement model as in Section 3, Table 5. Counterparts would apply for the other categories of measurement model.

Input parameters	
N	Number of input quantities
f	Function specifying the measurement function $Y = f(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
$\boldsymbol{u}$	Column vector $(u_1, \ldots, u_N)^\top \equiv (u(x_1), \ldots, u(x_N))^\top$ of standard uncertainties associated with the estimates of the input quantities. This parameter would be used if the GUM recommendation for estimating the sensitivity coefficients were adopted. It would not be used if a finite-difference formula were applied that attempted to provide a sensible compromise between loss of accuracy due to truncation (approximation) error and that due to subtractive cancellation
Output parameter	
$\boldsymbol{C}$	$1 \times N$ vector of sensitivity coefficients, whose ith element is an estimate obtained using finite differences of the partial derivative $\partial f / \partial X_i$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$

Table 33: Sensitivity coefficients obtained using a finite-difference formula.

The complex-step method [33, 37, 46] is similar to finite differences, but uses complex arithmetic to provide accurate values for sensitivity coefficients. The Taylor expansion for a complex function f takes the form

$$f(Z + W) = f(Z) + W f'(Z) + \frac{W^2}{2} f''(Z) + \frac{W^3}{3!} f'''(Z) + \frac{W^4}{4!} f^{\mathrm{iv}}(Z) + \cdots ,$$

where Z and W are complex numbers.[27] Taking $f = f_i$, $Z = x_i$ and $W = \mathrm{i}\delta_i$ where $\mathrm{i}^2 = -1$ and δ_i is real and small,

$$f_i(x_i + \mathrm{i}\delta_i) = f_i(x_i) + \mathrm{i}\delta_i f_i'(x_i) - \frac{\delta_i^2}{2} f_i''(x_i) - \mathrm{i}\frac{\delta_i^3}{3!} f_i'''(x_i) + \frac{\delta_i^4}{4!} f_i^{\mathrm{iv}}(x_i) + \cdots .$$

Taking real and imaginary parts,

$$\Re f_i(x_i + \mathrm{i}\delta_i) = f_i(x_i) - \frac{\delta_i^2}{2} f_i''(x_i) + \frac{\delta_i^4}{4!} f_i^{\mathrm{iv}}(x_i) + \cdots ,$$

and

$$\Im f_i(x_i + \mathrm{i}\delta_i) = \delta_i f_i'(x_i) - \frac{\delta_i^3}{3!} f_i'''(x_i) + \frac{\delta_i^5}{5!} f_i^{\mathrm{v}}(x_i) + \cdots .$$

[27] The Taylor expansion holds for what are termed analytical functions that have continuous derivatives of all orders, but these include almost all the function of interest to science.

From these last expressions, for δ_i small,

$$f_i(x_i) \approx \Re f_i(x_i + \mathrm{i}\delta_i), \qquad f_i'(x_i) \approx \frac{\Im f_i(x_i + \mathrm{i}\delta_i)}{\delta_i}, \tag{7}$$

both with a truncation error of order δ_i^2. Thus, both $f_i(x_i)$ and $f_i'(x_i)$ are obtained for one complex function evaluation. Unlike the use of a finite-difference formula, δ_i can be chosen to be *very* small with no concern about the loss of significant digits through subtractive cancellation since no subtraction is involved. The only practical restriction is that δ_i must not be chosen so small that it underflows, i.e., is replaced by zero in floating-point arithmetic. The value $\delta_i = 10^{-100}$ should be suitable for all but pathologically-scaled problems.

The success of this approach, i.e., the use of the approximation (7), depends on the availability of complex arithmetic and on the integrity of the inbuilt complex-valued functions. The complex step method is particularly suitable for implementation in software such as MATLAB [38] since the default data type is complex, and all the main intrinsic functions can be evaluated for complex arguments. Most functions can be worked with satisfactorily, but care must be taken to ensure that the intrinsic functions used in the function evaluation component behave in the appropriate way in complex arithmetic.

Table 34 specifies, for a univariate, real measurement function, the use of the complex-step method to obtain sensitivity coefficients, and indicates the input and output parameters associated with their determination. The input parameter f is used to provide information about the measurement function. In the complex-step method the function is evaluated for complex values of the input quantities $\boldsymbol{X}$. Counterparts would apply for the other categories of measurement model.

Input parameters	
N	Number of input quantities
f	Function specifying the measurement function $Y = f(\boldsymbol{X})$ in terms of the input quantities $\boldsymbol{X} = (X_1, \ldots, X_N)^\top$. The function is evaluated for complex values of the quantities $\boldsymbol{X}$
$\boldsymbol{x}$	Column vector $(x_1, \ldots, x_N)^\top$ of estimates of the input quantities $\boldsymbol{X}$
$\boldsymbol{\delta}$	Column vector $(\delta_1, \ldots, \delta_N)^\top$, where δ_i is a small real increment in x_i. δ_i must not be chosen so small that it underflows, i.e., is replaced by zero in floating-point arithmetic. The value $\delta_i = 10^{-100}$ should be suitable for all but pathologically-scaled problems
Output parameter	
$\boldsymbol{C}$	$1 \times N$ vector of sensitivity coefficients, whose ith element C_i is an estimate obtained using the complex-step method of the partial derivative $\partial f / \partial X_i$ of first order evaluated at $\boldsymbol{X} = \boldsymbol{x}$: $$C_i = \frac{\Im f_i(x_i + \mathrm{i}\delta_i)}{\delta_i}$$

Table 34: Sensitivity coefficients obtained using the complex-step method.

C Use of program differentiation techniques

Automatic differentiation (AD) is a set of techniques aimed at 'differentiating the program' that computes a function value [8, 20]. AD is an accurate method in the sense that AD applies the rules of calculus in a repetitive way to an algorithmic specification of a function and, in exact arithmetic, will produce the exact answer. Like symbolic algebra, AD relies on the fact that any program, no matter how complex, can be broken down into a finite combination of elementary operators such as arithmetic operations (e.g., $+$, $-$) and elementary functions (e.g., $\sin x$, $\cos x$, e^x) [7, 25].

There are two different approaches to differentiating program code: *operator overloading* and *source-to-source transformation*. In the operator overloading approach, the basic arithmetic operations and intrinsic functions are assigned routines that calculate the derivatives of the operator output in addition to the calculation of the function value. The source code of the function is progressively differentiated by calling these routines at the same time as each operation is performed in the evaluation of the function. Operator overloading is only allowed by a limited number of programing languages such as Fortran 90, ADA, C++ and Matlab. Examples of software packages that use this approach are ADOL-C [26] and ADOL-F [45].

The source-to-source approach defines a new source code for calculating the derivatives explicitly obtained from the program function evaluation source code. However, the implementation of this method requires considerable programming effort. Examples of software packages that use this approach are ADIFOR [6] and ODYSSEE [21].

D Use of Kahan summation

The natural way to evaluate, for example, the summation

$$S = \sum_{r=1}^{M} y_r$$

is *recursively*, i.e., by the following procedure:

$S = 0$
for $r = 1 : M$
    $S = S + y_r$
end

When implemented using floating-point arithmetic the *computed* result $\widetilde{S}$ will differ from the *mathematical* result S by an amount E that is the result of rounding errors associated with each floating-point operation (of which there will be M). Alternative procedures for undertaking the summation are available and have been designed with the aim of reducing the magnitude of E [27]. One such procedure is *Kahan summation* [28].

Suppose a and b are floating-point numbers with $|a| \geq |b|$, and let $\widetilde{s}$ denote the floating-point sum of a and b: $\widetilde{s} = \text{fl}(a + b)$.[28] Then,

$$\widetilde{e} = \text{fl}(-(((a + b) - a) - b)) = \text{fl}((a - \widetilde{s}) + b)$$

is an approximation to the error $(a + b) - \widetilde{s}$. Kahan's summation procedure uses this result to apply a correction $\widetilde{e}$ at every step of a recursive summation procedure for evaluating S. The procedure takes the following form:

$S = 0$
$e = 0$
for $r = 1 : M$
    $a = S$
    $b = y_r + e$
    $S = a + b$
    $e = (a - S) + b$
end

The method has two weaknesses: $\widetilde{e}$ is not necessarily the exact correction, and the addition $b = y_r + e$ is not performed exactly. Nevertheless, the use of the procedure brings a benefit in the form of an improved error bound compared with that for the (basic) recursive scheme for evaluating S [27].

[28] In general, $\text{fl}(p)$ denotes the result of computing expression p in floating-point arithmetic.

E Updating procedures in an implementation of a Monte Carlo method

A consideration for an implementation of a Monte Carlo method, as described in Section 5, is the need to store large quantities of data and to perform numerical operations on that data. For a univariate, real measurement model, an implementation of the method requires:

- Evaluating M values of the output quantity and storing those values (Section 5.4.1);
- Evaluating the arithmetic mean and standard deviation of the M values to provide the estimate of the output quantity and the associated standard uncertainty (Section 5.4.2);
- Sorting the M values, and storing the sorted values, to provide a discrete representation of the distribution function for the output quantity (Section 5.4.4);
- Evaluating a coverage interval in terms of the discrete representation (Section 5.4.6).

An approach is described here that does not require all M values of the output quantity to be stored at the same time. The approach is based on expressing the results (including the estimate of the output quantity, the associated standard uncertainty, etc.) for $M_0 + M_{\mathrm{seq}}$ Monte Carlo trials in terms of the results for M_0 trials and M_{seq} additional values of the output quantity. The procedure is applied iteratively starting with $M_0 = 0$. The approach only requires that the M_{seq} values of the output quantity are stored, where M_{seq} will generally be small compared to the total number M required, and avoids the need to sort the complete set of M values. It is also straightforward to integrate the approach with the adaptive Monte Carlo procedure described in Section 5.7 by taking $M_{\mathrm{seq}} = M_{\mathrm{adap}}$.

Suppose M_0 Monte Carlo trials have been undertaken, and the corresponding values $y_{0,r}$, $r = 1, \ldots, M_0$, of the output quantity are summarized by the arithmetic mean

$$y_0 = \frac{1}{M_0} \sum_{r=1}^{M_0} y_{0,r},$$

the variance

$$u^2(y_0) = \frac{1}{M_0 - 1} \sum_{r=1}^{M_0} (y_{0,r} - y_0)^2,$$

and the histogram defined by bins B_k and corresponding 'heights' $h_{0,k},\ k = 1, \ldots, b$, of the bars of the histogram (see Tables 23 and 24).

Suppose a further M_{seq} Monte Carlo trials are undertaken giving values $y_r,\ r = 1, \ldots, M_{\mathrm{seq}}$, of the output quantity. Then, the arithmetic mean of the combined set of $M_0 + M_{\mathrm{seq}}$ values is

$$y = y_0 + \frac{1}{M_0 + M_{\mathrm{seq}}} \sum_{r=1}^{M_{\mathrm{seq}}} (y_r - y_0), \tag{8}$$

and the variance is

$$u^2(y) = \frac{1}{M_0 + M_{\text{seq}} - 1}\left[(M_0 - 1)u^2(y_0) + M_0(y - y_0)^2 + \sum_{r=1}^{M_{\text{seq}}}(y_r - y)^2\right]. \qquad (9)$$

For the evaluation of y and $u^2(y)$ using formulae (8) and (9) it is not necessary to store the values of the output quantity from the first M_0 trials, but only the arithmetic mean y_0 and variance $u^2(y_0)$ evaluated in terms of those values.

The histogram for the combined set of values of the output quantity can similarly be updated. The new 'heights' of the bars of the histogram corresponding to the bins B_k, $k = 1, \ldots, b$, are given by

$$h_k = h_{0,k} + \frac{\text{card}(\{y_r \in B_k : r = 1, \ldots, M_{\text{seq}}\})}{\eta_k - \eta_{k-1}}, \qquad k = 1, \ldots, b. \qquad (10)$$

In cases where there are values y_r satisfying $y_r < \eta_0$ or $y_r > \eta_b$, those values do not lie in any of the bins B_k and are, consequently, not counted in the application of formula (10). It is necessary to store separately these values of the output quantity in order to determine a histogram for the complete set of values.

Let $y_r^{\text{L}},\ r = 1, \ldots, M_{\text{L}}$, denote the values to the left of η_0, with smallest values $y_{\min}^{\text{L}}$, and $y_r^{\text{R}},\ r = 1, \ldots, M_{\text{R}}$, those to the right of η_b, with largest value $y_{\max}^{\text{R}}$. Then, additional bars in the histogram for the complete set of values of the output quantity are defined by the bins

$$B_0 = [y_{\min}^{\text{L}}, \eta_0), \quad B_{b+1} = (\eta_b, y_{\max}^{\text{R}}]$$

with heights

$$h_0 = \frac{M_{\text{L}}}{\eta_0 - y_{\min}^{\text{L}}}, \quad h_{b+1} = \frac{M_{\text{R}}}{y_{\max}^{\text{R}} - \eta_b}.$$

Provided a reasonable (initial) choice of bins $B_k,\ k = 1, \ldots, b$, is made, it can generally be expected that the numbers M_{L} and M_{R} of values not contained in (one of) the bins are small. An approximation to the PDF for the output quantity is then given by the scaled histogram defined by bins B_k and probability densities $g_k,\ k = 0, \ldots, b + 1$, where

$$g_k = \frac{h_k}{M}, \qquad k = 0, \ldots, b + 1.$$

Let $y_{(r)}^{\text{L}},\ r = 1, \ldots, M_{\text{L}}$, and $y_{(r)}^{\text{R}},\ r = 1, \ldots, M_{\text{R}}$, denote the values y_r^{L} and y_r^{R} sorted into non-decreasing order. Corresponding to the (sorted) values

$$\boldsymbol{G} = \left(y_{(1)}^{\text{L}}, \ldots, y_{(M_{\text{L}})}^{\text{L}}, \eta_0, \ldots, \eta_b, y_{(1)}^{\text{R}}, \ldots, y_{(M_{\text{R}})}^{\text{R}}\right)$$

of the output quantity, assign cumulative probabilities

$$\boldsymbol{p} = \left(\frac{1/2}{M}, \ldots, \frac{M_{\text{L}} - 1/2}{M}, g_0, \ldots, \sum_{k=1}^{b} g_k, \frac{M - M_{\text{R}} + 1/2}{M}, \ldots, \frac{M - 1/2}{M}\right).$$

An approximation to the distribution function for the output quantity is then given by the piecewise-linear function joining the points $(G_k, p_k),\ k = 1, \ldots, M_{\mathrm{L}} + (b+1) + M_{\mathrm{R}}$. A coverage interval for the output quantity can be formed from this approximation to the distribution function in the manner described in Section 5.4.6.

For the case of a multivariate output quantity, the updating formulae (8) and (9) can straightforwardly be extended. Suppose M_0 Monte Carlo trials have been undertaken, and the corresponding values $\boldsymbol{y}_{0,r},\ r = 1, \ldots, M_0$, of the output quantities $\boldsymbol{Y}$ are summarized by the arithmetic mean $\boldsymbol{y}_0$ and covariance matrix $\boldsymbol{U}_{\boldsymbol{y}_0}$ (Section 5.5.2). Suppose a further M_{seq} Monte Carlo trials are undertaken giving values $\boldsymbol{y}_r,\ r = 1, \ldots, M_{\mathrm{seq}}$, of $\boldsymbol{Y}$. Then, the arithmetic mean of the combined set of $M_0 + M_{\mathrm{seq}}$ values is

$$\boldsymbol{y} = \boldsymbol{y}_0 + \frac{1}{M_0 + M_{\mathrm{seq}}} \sum_{r=1}^{M_{\mathrm{seq}}} (\boldsymbol{y}_r - \boldsymbol{y}_0), \tag{11}$$

and the covariance matrix is

$$\boldsymbol{U}_{\boldsymbol{y}} = \frac{1}{M_0 + M_{\mathrm{seq}} - 1} \left[(M_0 - 1)\boldsymbol{U}_{\boldsymbol{y}_0} + M_0 (\boldsymbol{y} - \boldsymbol{y}_0)(\boldsymbol{y} - \boldsymbol{y}_0)^{\top} + \widetilde{\Psi}\,\widetilde{\Psi}^{\top} \right], \tag{12}$$

where $\widetilde{\Psi}$ is the matrix Ψ containing the vectors $\boldsymbol{y}_r$, viz.,

$$\Psi = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{M_{\mathrm{seq}}}),$$

corrected for $\boldsymbol{y}$, i.e., $\boldsymbol{y}$ is subtracted from each column of Ψ. For the evaluation of $\boldsymbol{y}$ and $\boldsymbol{U}_{\boldsymbol{y}}$ using formulae (11) and (12) it is not necessary to store the values of the output quantity from the first M_0 trials, but only the arithmetic mean $\boldsymbol{y}_0$ and covariance matrix $\boldsymbol{U}_{\boldsymbol{y}_0}$ evaluated in terms of those values.

The histogram for the combined set of values of the output quantity can similarly be updated. Consider the case of a bivariate output quantity (as in Section 5.5.3). Let the 'heights' of the cuboids in the histogram for the M_0 values of $\boldsymbol{Y}$ corresponding to the bins B_{k_1,k_2} be denoted by $h_{0,k_1,k_2},\ k_1 = 1, \ldots, b_1,\ k_2 = 1, \ldots, b_2$. Then, following the further M_{seq} Monte Carlo trials, the new heights of the cuboids in the histogram are given by

$$h_{k_1,k_2} = h_{0,k_1,k_2} + \frac{\mathrm{card}(\{\boldsymbol{y}_r \in B_{k_1,k_2} : r = 1, \ldots, M_{\mathrm{seq}}\})}{(\eta_{1,k_1} - \eta_{1,k_1-1})\,(\eta_{2,k_2} - \eta_{2,k_2-1})}. \tag{13}$$

In cases where there are values $\boldsymbol{y}_r$ that do not lie in any of the bins B_{k_1,k_2} and are, consequently, not counted in the application of formula (13), it is necessary to store separately these values in order to determine a histogram for the complete set of values.

The determination of a coverage region for the output quantities $\boldsymbol{Y}$ in the form of an ellipsoid corresponding to coverage probability p is described in Section 5.5.4. It is based on sorting the values $\boldsymbol{y}_r$ of $\boldsymbol{Y}$ according to the 'distances' d_r of the transformed values $\dot{\boldsymbol{y}}_r$ from the estimate $\boldsymbol{y}$ of $\boldsymbol{Y}$. Specifically, the boundary of the coverage region is defined by the estimate $\boldsymbol{y}$, the covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$ associated with the $\boldsymbol{y}$, and the value $k_p = d_{(r^*)}$,

where r^* is the integer part of pM and $d_{(r)},\ r = 1, \ldots, M$, are the values d_r sorted into non-decreasing order.

An approach to calculating the coverage region that does not require the complete set of values $\boldsymbol{y}_r$ to be stored is as follows. It is based on determining the subset $\boldsymbol{y}_r^{\mathrm{t}},\ r = 1, \ldots, M_{\mathrm{t}}$, of values that lie outside the boundary of the coverage region corresponding to coverage probability $p - p_0$ for $0 < p_0 < p$. An updating procedure can be used to obtain approximately this subset. The subset corresponding to M_0 trials is augmented by the values from a further M_{seq} Monte Carlo trials that lie outside the boundary of a coverage region corresponding to coverage probability $p - p_0$ where the coverage region is specified in terms of the estimate and associated covariance matrix determined from the further M_{seq} values. The updating procedure is generally approximate because it is undertaken in terms of (approximations to) the required coverage region that are different each time. The choice of p_0 influences the degree of approximation as well as the storage requirements of the procedure. For p_0 close to p, all the values $\boldsymbol{y}_r$ are stored. For p_0 close to zero, approximately $100(1 - p)\,\%$ of the total number M of values are stored, but it is not guaranteed that the value from the complete set of values with index r^* is contained within the subset. The choice $p_0 = 0.05$ can be expected to deliver a compromise.

In terms of subset $\boldsymbol{y}_r^{\mathrm{t}},\ r = 1, \ldots, M_{\mathrm{t}}$, of values, $k_p = d_{(s^*)}$, where s^* is chosen such that $(M - M_{\mathrm{t}}) + s^*$ is the integer part pM and $d_{(r)},\ r = 1, \ldots, M_{\mathrm{t}}$, are the values d_r for the subset of values sorted into non-decreasing order. The boundary of the coverage region is then defined (approximately) by the estimate $\boldsymbol{y}$ and associated covariance matrix $\boldsymbol{U}_{\boldsymbol{y}}$ obtained from the complete set of values, e.g., using the formulae (11) and (12), and the value of k_p determined from the subset of values.