NPL Report CMSC 10/01

Report to the National Measurement System Policy Unit, Department of Trade and Industry

From the Software Support for Metrology Programme

Software Specifications for Uncertainty Calculation and Associated Statistical Analysis

By
M G Cox, M P Dainton and P M Harris

March 2001

Software Specifications for Uncertainty Calculation and Associated Statistical Analysis

M G Cox, M P Dainton and P M Harris
Centre for Mathematics and Scientific Computing

March 2001

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 and statistical modelling that has been produced as part of the UK Department of Industry's National Measurement System *Software Support for Metrology* (SS*f*M) Programme. The target audience is those who in their work wish to use software to assist in the evaluation of uncertainty.

ISSN 1471–0005

Authorised by Dr Dave Rayner,
Head of the Centre for Mathematics and Scientific Computing

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

Contents

List of Tables

List of Figures

1 Introduction

This document is intended to align with and help support established guides on uncertainty evaluation [1, 10, 29] and extend their functionality in a consistent manner. Also, it is complementary to the best-practice guide [6] on uncertainty and statistical modelling, produced as part of the UK's Software Support for Metrology programme. It contains 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 software specifications are not intended to be mandatory. They typify constituent parts of uncertainty evaluation. In particular, they indicate the inputs and outputs 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. Mainstream GUM, i.e., the "uncertainty budgeting" calculations as summarised in Clause 8 of the Guide to the Expression of Uncertainty in Measurement (GUM) [1]
2. More general calculations, consistent with the GUM, based on the use of Monte Carlo Simulation
3. The validation of (the use of) Mainstream GUM.

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 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 *uniform* random numbers 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 uniform generator for their function, and (c) in certain circumstances can be re-generated *exactly*, a *specific* uniform generator is recommended (Section 5.3). In addition, generators for Gaussian, Student's-t and multivariate Gaussian distributions are regarded as so fundamental that algorithmic statements of 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.[1]

The number of specifications will be increased in future editions of the document.

The specifications relate to the following combinations of measurement model types, there being $2 \times 2 \times 2 = 8$ model types in all:

- Models with univariate and multivariate measurands, i.e., a single (scalar) measurand and more than one (vector) measurand
- Explicit and implicit models, i.e., models that define the output quantity *directly*, viz., as *formulae*, and models in the form of *equations* that need to be *solved* for the output quantities given the input quantities (e.g., nonlinear least-squares adjustment)
- Real- and complex-valued models, i.e., models in which all the quantities are real-valued and those in which one or more of the quantities are complex-valued.

MetroS (the METROlogy Software environment) [2] is a system aimed at providing metrologists with access to software appropriate to their needs. It is a state-of-the-art solution to the provision of a metrology software library and is expected to set the model to be followed in the future. Software based on the specifications provided here will subsequently be incorporated in MetroS.

The coverage of the software specifications is divided according to whether they relate to Phase 1 (formulation) or Phase 2 (calculation) of the process of uncertainty evaluation (Section 1.1), or whether the procedure followed is

[1] Related aspects are treated in the Software Support for Metrology programme as part of the software testing project [7].

in accordance with Mainstream GUM ("uncertainty budgeting") [1, Clause 8], Monte Carlo Simulation [6] or the validation of Mainstream GUM [6].

Where a vector of values, e.g., the estimates $\mathbf{x} = (x_1, \ldots, x_n)^{\mathrm{T}}$ of n input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ 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 [13]. 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 a coverage probability of 95% and is couched this way. It is generally extensible to other coverage probabilities.

The report is organised as follows. In the remainder of this section the two phases of uncertainty evaluation, viz.,

- Formulation, Phase 1, and
- Calculation, Phase 2,

are discussed. In Section 2 the first phase and its outputs are described in terms of the information required for implementing the Mainstream GUM procedure and Monte Carlo Simulation. Section 3 covers the numerical evaluation of the measurement model according to the eight categories indicated earlier. In Sections 4, 5 and 6 specifications of software units are provided for the three aspects of uncertainty evaluation considered, Mainstream GUM, Monte Carlo Simulation and the validation of Mainstream GUM using Monte Carlo Simulation. Conclusions are given in Section 7.

1.1 The two phases of uncertainty evaluation

Uncertainty evaluation consists of two phases, *formulation* and *calculation*. In Phase 1 a measurement model is derived and the model inputs are quantified. Each model input quantity is represented by a probability density function (pdf). Each such pdf is typically expressed as

1. A continuous distribution (uniform, Gaussian, etc.)
2. A discrete distribution (a sufficient set of values of the input quantity, regarded as equally probable values of the quantity).

A continuous distribution arises from *assigning* a particular distributional form. According to the GUM, pdf's are obtained from an analysis of series

of observations [1, Clauses 2.3.2, 3.3.5] or are based on scientific judgement using all the relevant information available [1, Clauses 2.3.3, 3.3.5], [28].

If an input quantity is determined by a sufficiently large number q, say, of observations of it, those observations themselves can be used to "define" the pdf, as in the second category above. The input quantity is regarded as taking a value equal to any of the observed values with identical probability $1/q$.

Also falling into the second category, a pdf may be "defined" by the output from a previous uncertainty evaluation, as part of a multi-stage uncertainty evaluation process [6]. Such a pdf would be univariate in the case of a single output quantity from the previous stage. It would be multivariate in the case of more than one output from the previous stage. In the latter case the pdf would almost invariably be joint. A suitable continuous *multivariate* distribution (Gaussian, e.g., would be assigned), or the output itself would serve as a discrete representation of the joint pdf.

For an input quantity that is independent of the other input quantities, the Mainstream GUM procedure requires for its operation only three parameters of the pdf for that quantity:

- An estimate (a measure of location)
- The standard uncertainty of the estimate (a measure of dispersion)
- The corresponding degrees of freedom.

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

Software can support Phase 1 by providing estimates of location and dispersion and degrees of freedom from which a pdf can be constructed after making appropriate assumptions. In the case of a sufficiently large number of repeated observations of a set of the input quantities, the data can be used to define a covariance matrix or to define a multivariate Gaussian distribution, as appropriate. The specifications below relate to the (arithmetic) mean as a measure of location, and its standard deviation as a measure of dispersion. In addition, the determination of a covariance matrix is covered.

The output of Phase 1 is required to be in a form to act as inputs either to (the Phase 2 computations of)

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

- The Mainstream GUM ("uncertainty budgeting") calculation procedure, or to
- The Monte Carlo calculation procedure.

For the Mainstream GUM calculation phase, the Phase 2 inputs are the Phase 1 outputs identified above.

For Monte Carlo Simulation, the Phase 2 inputs are continuous pdf's, discrete pdf's or, where appropriate, joint pdf's for input quantities that are mutually dependent.

In order to operate Phase 2, therefore, software can assist with the following aspects.

The basis of Mainstream GUM is to "propagate the uncertainties" through the model to provide the uncertainties associated with the model output quantities. For this purpose it is required to

- Determine sensitivity coefficients
- Determine the combined standard uncertainty by propagating the covariance matrix of the input quantities through (a linearization of) the model to yield the covariance matrix of the output quantities
- Apply the Welch-Satterthwaite formula for effective degrees of freedom
- Calculate percentage points of a Gaussian or a Student's-t distribution in order to provide a coverage interval for the measurand.

These calculations would apply, as appropriate, for each of the model types.

For the Monte Carlo procedure it is necessary to "propagate distributions" through the model. For this purpose it is required to

- Sample from continuous pdfs or from discrete pdf's, or both
- Carry out an appropriate number of Monte Carlo trials
- Obtain a discrete representation of the pdf of the output quantity
- Obtain a coverage interval for the measurand from this discrete representation.

Again, these calculations would apply, as appropriate, for each of the model types.

The remaining software specifications are intended to support a recommended procedure for validating Mainstream GUM [6]. This procedure constitutes operating both Mainstream GUM and the Monte Carlo procedure for an uncertainty evaluation of concern, and carrying out an appropriate comparison of the results obtained.

2 Formulation phase

In Phase 1, the formulation phase of uncertainty evaluation, it is necessary to carry out various assignments of probability density functions (pdf's) to the input quantities of the measurement model. For the Mainstream GUM approach only the means and standard deviations of these pdf's, and covariances where appropriate, are used. For Monte Carlo Simulation the pdf's themselves are used. The way these assignments are undertaken depends on the information that is available about the input quantities. Two cases are considered.

If a set of q independent observations of the input quantity X_i is available, a *statistical analysis* of the observations is undertaken to determine an estimate x_i for X_i together with its standard uncertainty $u_i = u(x_i)$ and degrees of freedom ν_i. These are the quantities used in the Mainstream GUM approach (Section 4). To apply Monte Carlo Simulation a pdf with these parameters is assigned. The assignment of pdf's to the input quantities based on the analysis of repeated observations is described in Section 2.1. In the GUM the assignment is referred to as "a Type A evaluation of standard uncertainty".

If repeated observations are not available a pdf for the input quantity X_i is assigned on the basis of, for example, prior knowledge, experience or as the result of a previous uncertainty evaluation. The pdf is used directly in Monte Carlo Simulation (Section 5). To apply the Mainstream GUM approach parameters x_i, u_i and ν_i are derived from the pdf. The derivation for a number of common pdf's is described in Section 2.2, and is referred to in the GUM as "a Type B evaluation of standard uncertainty".

A variant of these analyses applies if some or all of the input quantities are correlated. For the Mainstream GUM approach a covariance matrix would be assigned to the relevant input quantities. For Monte Carlo Simulation a joint (multivariate) pdf would be assigned to these input quantities.

In practice both types of assignment will be required, each applying to a subset of the input quantities. For the Mainstream GUM approach a covariance matrix would be constructed from those covariance matrices

arising from the two types of analysis. For Monte Carlo Simulation a joint pdf would be constructed from those pdf's arising from the two types of analysis.

2.1 Formulation based on analysing repeated observations

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

Given a set of observations for an input quantity X_i, assign an estimate x_i for X_i together with the standard uncertainty u_i for that estimate and the corresponding degrees of freedom ν_i (Section 2.1.1).

Given a set of observations for a pair of input quantities that are mutually dependent, assign an estimate of the covariance for these quantities (Section 2.1.2).

These calculations are the basis for assigning a covariance matrix for a number of input quantities (Section 2.1.3).

2.1.1 Sample mean and its standard deviation

Suppose $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})^{\mathrm{T}}$ are q independent observations of the input quantity X_i. The estimate x_i of X_i and its standard uncertainty u_i are determined, respectively, as $\bar{x}_i$, the arithmetic mean of the observations, and s_i, the standard deviation of $\bar{x}_i$. The corresponding degrees of freedom ν_i is $q - 1$. Table 1 defines the sample mean of a set of values, the standard deviation of 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 Mainstream GUM procedure (Section 4).

Given the estimate x_i and its standard uncertainty s_i so obtained, a pdf for X_i is assigned as follows [6]:

- If the distribution underlying the observations is unknown, assign a Gaussian pdf[3] with mean x_i and standard deviation s_i
- If the distribution underlying the observations is known to be Gaussian, assign the distribution $x_i + s_i t$, where t is a Student's-t pdf with $\nu_i = q - 1$ degrees of freedom

[3]Equivalently, assign the distribution $x_i + s_i z$, where z is a standardized Gaussian pdf (with mean zero and unit standard deviation).

Input parameters	
q	Number of observations of the ith input quantity
$\mathbf{x}_i$	Column vector $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})^{\mathrm{T}}$ of independent observations of X_i
Output parameters	
$\bar{x}_i$	Sample mean, defined by $$\bar{x}_i = \frac{1}{q}\sum_{k=1}^{q} x_{i,k}$$
s_i	Standard deviation of the sample 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	
	It is important to use the above formula for s_i 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 the uncertainty evaluation to be valid [5]

Table 1: The sample mean, its standard deviation and degrees of freedom for the ith input quantity.

- If q is large and the observations show appreciable departure from Gaussian behaviour[4], assign a discrete pdf. The "observations" in this case may be the result of Monte Carlo Simulation applied to an earlier stage of a multi-stage model.

This is the information required for Monte Carlo Simulation (Section 5).

2.1.2 Covariance of two sample means

Suppose $(x_{i,k}, x_{j,k})^{\mathrm{T}}$, $k = 1, \ldots, q$, are q *pairs* of independent observations of the input quantities X_i and X_j. An estimate x_i of X_i, together with its standard uncertainty s_i, may be determined as the mean and its standard deviation for the set of observations $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})^{\mathrm{T}}$ as in Section 2.1.1, and similarly for X_j. The covariance of X_i and X_j is estimated by the covariance $u_{i,j} = u(x_i, x_j)$ of the sample means x_i and x_j. Table 2 defines this covariance in terms of repeated observations of the two input quantities, and specifies the input and output parameters associated with its determination.

2.1.3 Covariance matrix for input quantities

Suppose observations $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})^{\mathrm{T}}$ are available for *all* input quantities X_i, $i = 1, \ldots, n$, and that these are assembled into the $n \times q$ matrix X. Let X' be obtained from X by correcting for the sample means, i.e., the mean $\bar{x}_i$ of the ith row is subtracted from all elements $x_{i,j}$, $j = 1, \ldots, q$, in that row. Then, the covariance matrix V_x of covariances $u_{i,j}$ for the input quantities $\mathbf{X} = (X_1, X_2, \ldots, X_n)^{\mathrm{T}}$ is estimated by

$$\frac{1}{q(q-1)} X'(X')^{\mathrm{T}}.$$

Table 3 defines the covariance matrix for input quantities $\mathbf{X}$ in terms of repeated observations for $\mathbf{X}$, and specifies the input and output parameters associated with its determination.

Given the estimates x_i $(= \bar{x}_i)$, $i = 1, \ldots, n$, for the input quantities $\mathbf{X}$ and their covariance matrix V_x, a multivariate Gaussian distribution is assigned for the purpose of Monte Carlo Simulation. This multivariate distribution is discussed in Section 5.3.4.

In practice there may be "simultaneous" observations $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})^{\mathrm{T}}$ for *some* of the X_i, e.g, for $i = 2$, 4 and 5. In this case the above covariance

[4]The departure would be assessed with a normality test or visualization using a histogram, for example.

Input parameters	
q	Number of observations of the ith and of the jth input quantities
$\mathbf{x}_i$	Column vector $(x_{i,1}, x_{i,2}, \ldots, x_{i,q})^{\mathrm{T}}$ of independent observations of X_i
$\mathbf{x}_j$	Column vector $(x_{j,1}, x_{j,2}, \ldots, x_{j,q})^{\mathrm{T}}$ of independent observations of X_j paired with those for X_i
Output parameters	
$u_{i,j}$	Sample covariance of X_i and X_j, defined by $$u(x_i, x_j) = \frac{1}{q(q-1)} \sum_{k=1}^{q} (x_{i,k} - \bar{x}_i)(x_{j,k} - \bar{x}_j),$$ where $\bar{x}_i$ and $\bar{x}_j$ are the sample means of the sets of observations $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively
Numerical analysis	
	It is important to use the above formula for $u(x_i, x_j)$ rather than the mathematically equivalent formula $$u(x_i, x_j) = \frac{1}{q-1} \left(\frac{1}{q} \sum_{k=1}^{q} x_{i,k} x_{j,k} - \bar{x}_i \bar{x}_j \right).$$ 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: The covariance of the sample means for the ith and jth input quantities.

Input parameters	
n	Number of input quantities
q	Number of observations of each of the input quantities
X	An $n \times q$ matrix containing the observations $x_{i,j}$, where $x_{i,j}$ is the jth measurement of the ith input quantity
Output parameters	
V_x	Covariance matrix for the input quantities $\mathbf{X}$, defined by $$\frac{1}{q(q-1)} X'(X')^{\mathrm{T}},$$ where X' is X corrected for the sample means

Table 3: The covariance matrix for the input quantities $\mathbf{X}$.

considerations would apply to this "group". Any other such groups would be handled similarly, and the complete $n \times n$ covariance matrix constructed from these groups, together with the variances for those input quantities that are mutually independent.

2.2 Formulation based on assigning probability distributions

If an input quantity is based on non-statistical information, a pdf such as a uniform distribution would be assigned. For the Monte Carlo approach this pdf is used directly. For the Mainstream GUM approach the mean and the standard uncertainty would be extracted from the pdf and used. Table 4 indicates how the mean x_i, the standard uncertainty u_i and the degrees of freedom ν_i would be obtained from the pdf's for some common distributions, viz., Gaussian, uniform (or rectangular), triangular and U-shaped.

Distribution	Parameters	Probability density function	x_i	u_i	ν_i
Gaussian $N(\mu, \sigma^2)$	Mean μ Standard deviation σ	$\frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}$	μ	σ	∞
Uniform	Limits a and b	$\begin{cases} 0, & x < a, \\ \frac{1}{2w}, & a \le x \le b, \\ 0, & b < x. \end{cases}$	h	$\frac{w}{\sqrt{3}}$	∞
Triangular	Limits a and b	$\begin{cases} 0, & x < a \\ \frac{x-a}{w^2}, & a \le x \le h, \\ \frac{b-x}{w^2}, & h \le x \le b, \\ 0, & b < x. \end{cases}$	h	$\frac{w}{\sqrt{6}}$	∞
U-shaped [14]	Limits a and b	$\begin{cases} 0, & x < a \\ \frac{1}{\pi\sqrt{w^2-(x-h)^2}}, & a \le x \le b, \\ 0, & b < x. \end{cases}$	h	$\frac{w}{\sqrt{2}}$	∞

Table 4: Parameters of common probability density functions, where w denotes $(b-a)/2$ and h denotes $(b+a)/2$.

For example, suppose the input quantity X_i is assigned a uniform distribution between the limits a and b. Information on how to generate pseudo-random numbers from this distribution for the purpose of Monte Carlo Simulation is given in Section 5.3.1. For this distribution

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

(Table 4), and this is the information required for the Mainstream GUM

procedure. For other pdf's assigned to X_i, Table 4 would be used in a similar way.

3 Numerical evaluation of the model

The classification of models in Section 1 covers eight types of measurement model. In principle, any model will naturally fall into one, and only one of these categories. Some 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-valued model can always be converted into a real-valued model, by replacing each complex-valued quantity by two real-valued quantities, its real and imaginary parts. Again, doing so is not necessarily desirable for purposes of model evaluation (but see the end of this section regarding uncertainties).

For the categories of *explicit* models, $Y = f(\mathbf{X})$ (univariate) and $\mathbf{Y} = \mathbf{f}(\mathbf{X})$ (multivariate), the calculation of the measurement result y or $\mathbf{y}$ requires the evaluation of the *formula* f or $\mathbf{f}$. Tables 5 and 6 specify the evaluation of these models, and the input and output parameters required.

The calculation of the measurement result Y or $\mathbf{Y}$ for the corresponding categories of *implicit* models, $h(Y, \mathbf{X}) = 0$ and $\mathbf{h}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}$, requires the *solution* of an equation or a system of equations. For a univariate model, it is necessary to solve a single equation. For a multivariate model, a system of equations is to be solved. Tables 7 and 8 specify the evaluation of these models, and the input and output parameters required.

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

An issue that requires consideration as part of any implementation for these categories of complex-valued models is the way the "(standard) uncertainty" of a complex-valued quantity is stored. If X_i is complex-valued with real and imaginary parts X_i^R and X_i^I, the "(standard) uncertainty" is described by the 2×2 covariance matrix

$$V_i = \begin{bmatrix} u^2(x_i^R) & u(x_i^R, x_i^I) \\ u(x_i^R, x_i^I) & u^2(x_i^I) \end{bmatrix},$$

where $u^2(x_i^R)$ and $u^2(x_i^I)$ are, respectively, the variances of the estimates

x_i^R and x_i^I of the real and imaginary parts, and $u(x_i^R, x_i^I)$ is the covariance of these parts. Furthermore, the covariance matrix V_x for the complete set of input quantities $X_i,\ i = 1, \ldots, n$, is a $2n \times 2n$ matrix. Consequently, although the quantities themselves may be regarded as complex-valued, it is necessary to store the corresponding uncertainty information using real-valued matrices, and operate on them using real arithmetic.

Input parameters	
n	Number of input quantities
f	Function specifying the model $Y = f(\mathbf{X})$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
Output parameters	
y	Value obtained by evaluating the formula $$y = f(\mathbf{x})$$

Table 5: Numerical evaluation of the univariate, explicit, real-valued measurement model $Y = f(\mathbf{X})$.

Input parameters	
n	Number of input quantities
m	Number of output quantities
$\mathbf{f}$	Function specifying the model $\mathbf{Y} = \mathbf{f}(\mathbf{X})$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
Output parameters	
$\mathbf{y}$	Column vector of values $(y_1, \ldots, y_m)^{\mathrm{T}}$ obtained by evaluating the formula $$\mathbf{y} = \mathbf{f}(\mathbf{x})$$

Table 6: Numerical evaluation of the multivariate, explicit, real-valued measurement model $\mathbf{Y} = \mathbf{f}(\mathbf{X})$.

Input parameters	
n	Number of input quantities
h	Function specifying the model $h(Y, \mathbf{X}) = 0$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ and the output quantity Y
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
$\mathbf{t}$	Computational control parameters such as an initial estimate of y and technical parameters relating to the equation-solving software
Output parameters	
y	Value obtained by solving the equation $$h(y, \mathbf{x}) = 0$$
Numerical analysis	
	A zero-finding algorithm [8, 12], such as the bisection algorithm in cases where suitable lower and upper bounds are known for the measurement result, can be used to solve the equation

Table 7: Numerical evaluation of the univariate, implicit, real-valued measurement model $h(Y, \mathbf{X}) = 0$.

Input parameters	
n	Number of input quantities
m	Number of output quantities
$\mathbf{h}$	Function specifying the model $\mathbf{h}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ and the output quantities $\mathbf{Y} = (Y_1, \ldots, Y_m)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
$\mathbf{t}$	Computational control parameters such as an initial estimate of $\mathbf{y}$ and technical parameters relating to the equation-solving software
Output parameters	
$\mathbf{y}$	Column vector of values $(y_1, \ldots, y_m)^{\mathrm{T}}$ obtained by solving the equations $$\mathbf{h}(\mathbf{y}, \mathbf{x}) = \mathbf{0}$$
Numerical analysis	
	An iterative algorithm such as Newton's method [12], starting from a suitable estimate of the measurement result, can be used to solve the system of equations

Table 8: Numerical evaluation of the multivariate, implicit, real-valued measurement model $\mathbf{h}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}$.

4 Mainstream GUM: calculation phase

4.1 Procedure

For Mainstream GUM, the outputs of the formulation phase, Phase 1, are (Section 2):

- Estimates $\mathbf{x} = (x_1, \ldots, x_n)^{\mathrm{T}}$ of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
- Standard uncertainties $\mathbf{u} = (u_1, \ldots, u_n)^{\mathrm{T}}$ of these estimates
- Corresponding degrees of freedom $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)^{\mathrm{T}}$
- Where appropriate, the covariances between input quantities that are mutually dependent.

This information is conveniently represented using the three quantities $\mathbf{x}$, $\boldsymbol{\nu}$ and V_x, where V_x is a *covariance* matrix that holds the standard uncertainties for the estimates $\mathbf{x}$ and the covariances of these estimates. V_x is an $n \times n$ matrix, whose (i, j)th element contains the covariance $u(x_i, x_j)$ of x_i and x_j, with $u(x_i, x_i) = u_i^2$.

The three quantities $\mathbf{x}$, $\boldsymbol{\nu}$ and V_x, together with the measurement model and the required coverage probability p (e.g., 0.95, or 95%), constitute the inputs to the calculation phase, Phase 2, for the Mainstream GUM procedure.

For *univariate* measurement models, the following steps constitute the calculation phase:

1. Form numerically the measurement result by evaluating the model at the estimates of the input quantities.

   See Section 3.

2. Form the partial derivatives of first order of the measurement model with respect to the input quantities, and determine numerical values for the sensitivity coefficients by evaluating these partial derivatives at the estimates of the input quantities.[5]

   See Section 4.2.

3. Determine the combined standard uncertainty, viz., the standard deviation of the model output quantity, by combining the values of the

[5] For an *implicit* 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.

input standard uncertainties, the input covariances and the sensitivity coefficients.

See Section 4.3.

4. Use the Welch-Satterthwaite formula to calculate ν_{eff}, the effective degrees of freedom of the output quantity, from the standard uncertainties and degrees of freedom of the input quantities, the sensitivity coefficients and the combined standard uncertainty.[6]

   See Section 4.4.

5. Compute the coverage factor corresponding to ν_{eff} and the required coverage probability p as a percentage point of the (standardized) Gaussian distribution ($\nu_{\text{eff}} = \infty$) or a Student's-$t$ distribution ($\nu_{\text{eff}} < \infty$). Hence, compute the expanded uncertainty, and thus an interval having a stipulated coverage probability containing the measurand, by forming the product of this coverage factor and the combined standard uncertainty.

   See Section 4.5.

The computational flow of the Mainstream GUM calculation phase, indicating the inputs to the phase and the primary output, a coverage interval, is given in Figure 1. This figure applies in the case of a univariate, explicit, real-valued model with no covariance effects. Other 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. However, no information is given here regarding steps 4 and 5 for such models. The extension of steps 4 and 5 to the evaluation of coverage *regions* for multivariate measurement results is not straightforward. Consideration of this matter, which is a topic for research [6], will be given in future versions of this report.

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

[6] Mainstream GUM does not state how ν_{eff} is to be calculated when the input quantities are correlated.

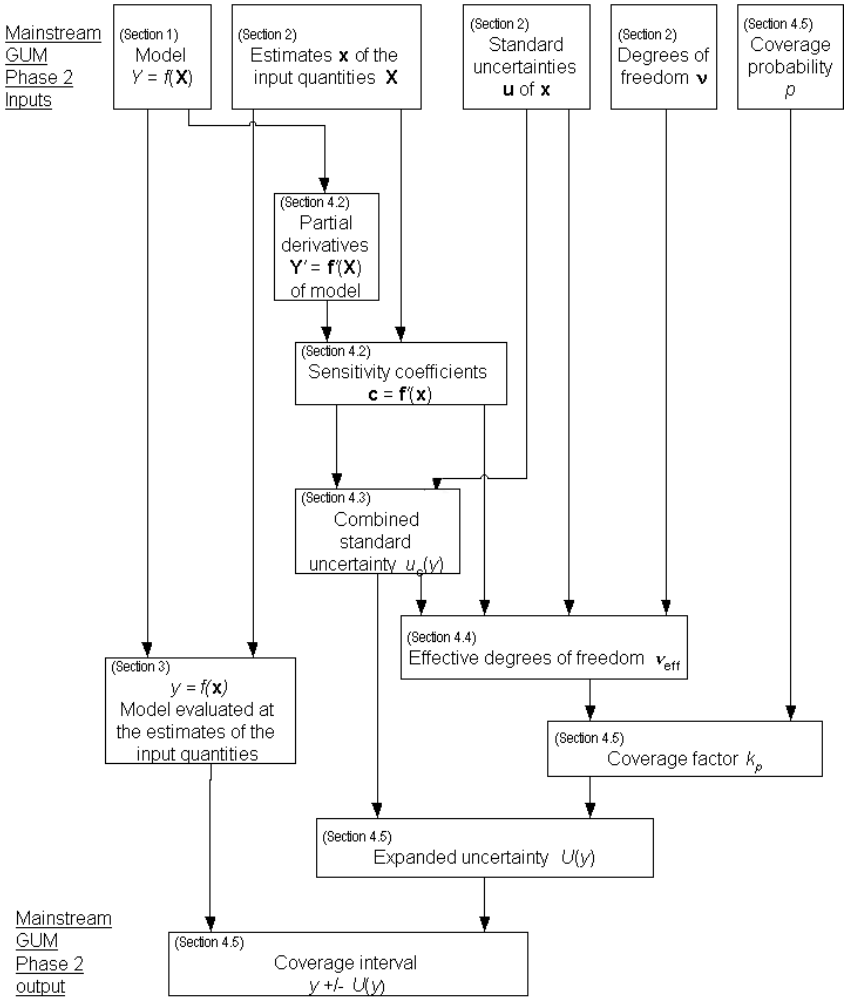

Figure 1: Phase 2, calculation, of uncertainty evaluation using Mainstream GUM. The stages shown relate to the case of a univariate, explicit, real-valued model with mutually independent input quantities.

4.2 Sensitivity coefficients

The Mainstream GUM procedure covers the univariate, explicit, real-valued model of measurement $Y = f(\mathbf{X})$. The *sensitivity coefficients* used by that procedure are denoted here by the (row) vector[7] $C = (c_1, \ldots, c_n)$, whose jth element c_j is the partial derivative $\partial f/\partial X_j$ of $f(\mathbf{X})$ evaluated at $\mathbf{X} = \mathbf{x}$. 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.

	Input parameters
n	Number of input quantities
f	Function specifying the model $Y = f(\mathbf{X})$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
	Output parameters
C	A $1 \times n$ vector of sensitivity coefficients, whose jth element is the partial derivative $\partial f/\partial X_j$ of $f(\mathbf{X})$ evaluated at $\mathbf{X} = \mathbf{x}$

Table 9: Sensitivity coefficients for a univariate, explicit, real-valued measurement model $Y = f(\mathbf{X})$.

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

	Input parameters
n	Number of input quantities
m	Number of output quantities
$\mathbf{f}$	Function specifying the model $\mathbf{Y} = \mathbf{f}(\mathbf{X})$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
	Output parameters
C	An $m \times n$ matrix of sensitivity coefficients, whose (i, j)th element is the partial derivative $\partial f_i/\partial X_j$ of $\mathbf{f}(\mathbf{X})$ evaluated at $\mathbf{X} = \mathbf{x}$

Table 10: Sensitivity coefficients for a multivariate, explicit, real-valued measurement model $\mathbf{Y} = \mathbf{f}(\mathbf{X})$.

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

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

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

where the partial derivatives are evaluated at $\mathbf{X} = \mathbf{x}$ and $Y = y$, with y satisfying $h(y, \mathbf{x}) = 0$. 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 *multivariate*, implicit, real-valued model of measurement is $\mathbf{h}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}$. The matrix C of sensitivity coefficients is determined as the solution to the linear system of equations

$$H_y C = H_x,$$

where the matrices H_x and H_y contain, respectively, the partial derivatives of the components h_i of $\mathbf{h}$ with respect to the input quantities X_j and output quantities Y_j, evaluated at $\mathbf{X} = \mathbf{x}$ and $\mathbf{Y} = \mathbf{y}$, with $\mathbf{y}$ satisfying $\mathbf{h}(\mathbf{y}, \mathbf{x}) = \mathbf{0}$. 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-valued* models. It is necessary to form partial derivatives 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 implicit models).

The sensitivity coefficients can be formed

1. Manually, by the algebraic differentiation of, e.g., $f(\mathbf{X})$ with respect to each component X_i, $i = 1, \ldots, n$, of $\mathbf{X}$, followed by the substitution of the values of $\mathbf{x}$ for $\mathbf{X}$
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.

The manner in which the partial derivatives required in forming the sensitivity coefficients C in Tables 9–12 requires careful consideration. These derivatives and hence the sensitivity coefficients can be difficult to determine by hand for models that are complicated. The above options of using a

Input parameters	
n	Number of input quantities
h	Function specifying the model $h(Y, \mathbf{X}) = 0$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ and the output quantity Y
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
y	Estimate of the output quantity Y that satisfies $h(y, \mathbf{x}) = 0$
Output parameters	
C	A $1 \times n$ 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 are evaluated at $\mathbf{X} = \mathbf{x}$ and $Y = y$

Table 11: Sensitivity coefficients for a univariate, implicit, real-valued measurement model $h(Y, \mathbf{X}) = 0$.

Input parameters	
n	Number of input quantities
m	Number of output quantities
$\mathbf{h}$	Function specifying the model $\mathbf{h}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ and the output quantities $\mathbf{Y} = (Y_1, \ldots, Y_m)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
$\mathbf{y}$	Column vector $(y_1, \ldots, y_m)^{\mathrm{T}}$ of estimates of the output quantities $\mathbf{Y}$ that satisfy $\mathbf{h}(\mathbf{y}, \mathbf{x}) = \mathbf{0}$
Output parameters	
C	An $m \times n$ matrix of sensitivity coefficients that solves $$H_y C = H_x,$$ where the matrices H_x and H_y contain, respectively, the partial derivatives of the components h_i of $\mathbf{h}$ with respect to the input quantities X_j and output quantities Y_j, evaluated at $\mathbf{X} = \mathbf{x}$ and $\mathbf{Y} = \mathbf{y}$.
Numerical analysis	
	It is important to solve the above system of equations using a numerically stable procedure, such as Gaussian elimination with a pivoting strategy [13]

Table 12: Sensitivity coefficients for a multivariate, implicit, real-valued measurement model $\mathbf{h}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}$.

symbolic-algebra package or finite-difference formulae can be attractive in such circumstances. There are learning overheads associated with the use of a symbolic-algebra package: its use can be justified if the user needs to address a sufficient number of complicated 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 and B.

4.3 Combined standard uncertainty

For a univariate, explicit, real-valued measurement model, the combined standard uncertainty $u_c(y)$ for the measurement result is obtained from the formula (cf. [1])

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

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

$$V_y = C V_x C^{\mathrm{T}}, \tag{1}$$

where $V_y = u_c^2(y)$, the variance of the output quantity, $C = (c_1, \ldots, c_n)$, the row vector of sensitivity coefficients, and V_x is the $n \times n$ covariance matrix for the input quantities.

Using this notation, Formula (1) for V_y applies for all categories of measurement model. For univariate, real-valued models, V_y estimates the variance of the output quantity; for other measurement models, it estimates the covariance matrix for the output quantities. Table 13 specifies the evaluation of combined standard uncertainty, and indicates the input and output parameters necessary for its determination.

4.4 Effective degrees of freedom

Table 14 specifies the Welch-Satterthwaite formula for evaluating the effective degrees of freedom ν_{eff} for a univariate, real-valued output quantity Y in the case that the input quantities $\mathbf{X}$ are mutually independent.

Input parameters	
n	Number of input quantities
m	Number of output quantities
V_x	An $n \times n$ covariance matrix for the input quantities
C	An $m \times n$ matrix of sensitivity coefficients
Output parameters	
V_y	An $m \times m$ covariance matrix for the output quantities obtained by evaluating $$V_y = CV_xC^{\mathrm{T}}$$

Table 13: Combined standard uncertainty for a univariate ($m = 1$) or a multivariate (general m) model.

Input parameters	
n	Number of input quantities
$\mathbf{u}$	Column vector $(u_1, \ldots, u_n)^{\mathrm{T}} \equiv (u(x_1), \ldots, u(x_n))^{\mathrm{T}}$ of standard uncertainties of the estimates $\mathbf{x} = (x_1, \ldots, x_n)^{\mathrm{T}}$ of the input quantities
C	A $1 \times n$ (row) vector of sensitivity coefficients, whose jth element is the partial derivative $\partial f/\partial X_j$ of $f(\mathbf{X})$ evaluated at $\mathbf{X} = \mathbf{x}$
$\boldsymbol{\nu}$	Column vector $(\nu_1, \ldots, \nu_n)^{\mathrm{T}}$ of degrees of freedom. If x_i is estimated as the mean of a set of q repeated observations, ν_i is taken as $q - 1$. If x_i is estimated as the mean (mid-point) of a uniform distribution with accurately-known endpoints, ν_i is taken as infinite (∞). (Since ∞ cannot be represented as such as an input quantity, 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 [1] associated with assigning the degrees of freedom to estimates of the input quantities. Each case has to be treated on its own merits
$u_{\mathrm{c}}(y)$	Combined standard uncertainty for measurement result
Output parameters	
ν_{eff}	Effective degrees of freedom determined from the Welch-Satterthwaite formula $$\frac{u_{\mathrm{c}}^4(y)}{\nu_{\mathrm{eff}}} = \sum_{i=1}^{n} \frac{c_i^4 u^4(x_i)}{\nu_i}$$

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

4.5 Expanded uncertainty

Table 15 specifies the calculation of expanded uncertainty $U(y)$ for a univariate, real-valued output quantity Y in the case that the input quantities $\mathbf{X}$ are mutually independent. The expanded uncertainty is evaluated as the product of the combined standard uncertainty $u_{\mathrm{c}}(y)$ (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).

The calculation of expanded uncertainty depends on knowledge of the distribution of the output quantity. In the Mainstream GUM approach, a Student's-t distribution with ν_{eff} degrees of freedom is assigned to the random variable

$$t = \frac{y - Y}{u_{\mathrm{c}}(y)}.$$

It follows that k_p is the percentage point $t_p(\nu_{\mathrm{eff}})$ of the Student's-t distribution such that the probability that $|t|$ is no greater than $t_p(\nu_{\mathrm{eff}})$ is equal to p. 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 $\nu_{\mathrm{eff}} \geq 473$, $k_p = 1.96$, correct to two decimal places, the corresponding value for the standardized Gaussian distribution $N(0, 1)$.

Input parameters	
$u_{\mathrm{c}}(y)$	Combined standard uncertainty for measurement result y
ν_{eff}	Effective degrees of freedom determined from the Welch-Satterthwaite formula
p	Coverage probability (typically 0.95 or 95%)
Output parameters	
$U(y)$	Expanded uncertainty, defined by $$U(y) = k_p u_{\mathrm{c}}(y),$$ where $u_{\mathrm{c}}(y)$ is the combined standard uncertainty for the estimate y of Y of the output quantity, and 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 ($\nu_{\mathrm{eff}} = \infty$) or the Students'-$t$ distribution ($\nu_{\mathrm{eff}} < \infty$)

Table 15: Expanded uncertainty.

5 Monte Carlo Simulation: calculation phase

5.1 Procedure

For Monte Carlo Simulation, the outputs of the formulation phase, Phase 1, are (Section 2) the pdf's $\mathbf{g}(\mathbf{X}) = (g_1(X_1), \ldots, g_n(X_n))^{\mathrm{T}}$ of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$.[8]

The pdf's, together with the measurement model and the required coverage probability p (e.g., 0.95, or 95%), constitute the inputs to the calculation phase, Phase 2, for the Monte Carlo Simulation procedure.

For *univariate* measurement models, the following steps constitute the calculation phase:

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

   See Section 5.2.

2. Generate M samples of the input quantities.

   See Section 5.3.

3. For each sample, evaluate the model to give the corresponding value of the output quantity.

   See Section 5.4.

4. Form the measurement result and its standard uncertainty from this set of model values.

   See Section 5.5.

5. Sort the values of the output quantity into non-decreasing order, and use the sorted values to provide an estimate of the distribution function of the output quantity.

   See Section 5.6

6. Form the 0.025-and 0.975-fractiles of this distribution function, taking these fractiles as a 95% coverage interval for the measurand.

   See Section 5.7

Figure 2 shows the procedure diagrammatically.

The modifications required for other types of univariate model are straightforward. Multivariate models are considered in Section 5.8.

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

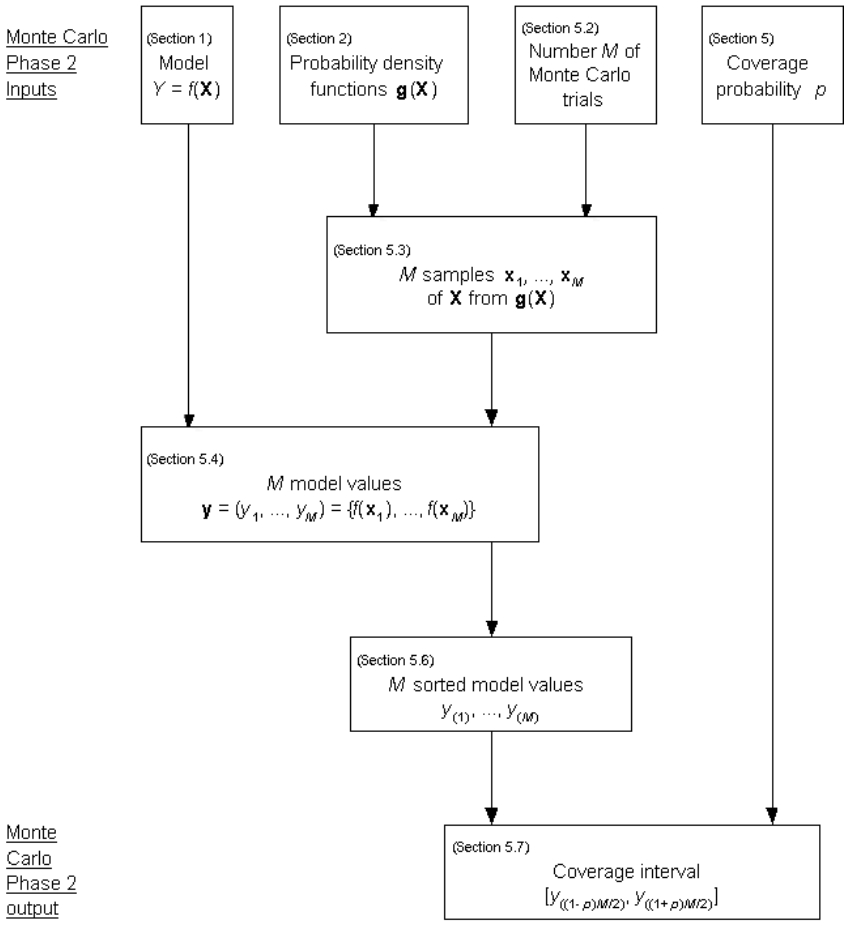

Figure 2: Phase 2, calculation, of uncertainty evaluation using Monte Carlo Simulation. The stages shown relate to the case of a univariate, explicit, real-valued model.

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 made *a priori*, in which case there will be no direct control over the accuracy delivered by the Monte Carlo procedure. The reason is that the number needed to provide a prescribed accuracy will depend on the "shape" of the pdf of the output quantity and the level of probability required. Also, the calculations are *stochastic* in nature, being based on random sampling. However, in the experience of the authors a value of the order of 50,000 is often appropriate to deliver two-figure accuracy in providing a 95% coverage interval.

Because there is no guarantee that this or any specific number will suffice, it is recommended to use instead a process that selects M adaptively, i.e., as the trials progress. Some guidance in this regard is available [3, 6]. The main advantage of such a process is that it provides a sensible compromise between achieving the required number of correct figures and taking a number of trials that it economical consistent with this requirement [6].

The procedure above and Figure 2 would require modification for an adaptive choice of M.

5.3 Sampling from the probability density functions

In an implementation of the Monte Carlo process M samples $\mathbf{x}_i$, $i = 1, \ldots, M$, are obtained of the input quantities $\mathbf{X}$ from the probability density functions assigned to these input quantities.

In general, some of these pdf's will be continuous (uniform, Gaussian, etc.) and some will be discrete (based on large numbers of repeated observations or on Monte Carlo results from a previous uncertainty evaluation in a multi-stage process). A joint (multivariate) Gaussian pdf would be assigned when appropriate (Section 2.1.3).

Recommendations concerning the manner in which this sampling should be carried out are given here for the commonest distributions, viz., the uniform, the Gaussian, the Student's-t and the multivariate Gaussian. It is possible to prepare software for sampling from almost any distribution, and indeed to develop a general framework for doing so [6]. Such considerations will be covered in a future edition of this document.

The manner in which sampling can be carried out from a discrete distribution is also indicated. As part of this process it is required to sample at random from the integers 1 to q, for arbitrary positive q.

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

5.3.1 Uniform distribution

The ability to generate pseudo-random numbers from a (continuous) uniform distribution is fundamental in its own right, and also as the basis for generating numbers from any distribution (as indicated below) using an appropriate algorithm or formula. In the latter regard, the quality of the numbers generated from a non-uniform distribution depends on that of the uniform generator and on the properties of the algorithm employed. The quality of the non-uniform generator can therefore be expected to be correlated with that of the uniform generator. A good uniform generator and a good algorithm can be expected to provide a good non-uniform generator. A poor uniform generator and a good or bad algorithm can be expected to provide a poor non-uniform generator. It is thus especially important that the underlying uniform generator is sound (cf. [18]). Unless the user is sure of the pedigree of a uniform 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 uniform generator, that has been shown to perform well in these tests and that is straightforward to implement, is given in this section.

Table 16 defines relevant aspects of the functioning of a procedure for generating uniform pseudo-random numbers in the interval $[0,\ 1]$, specifying the input, input-output and output parameters associated with their determination.

A uniform pseudo-random number x in the interval $[a,\ b]$ can be formed by determining $a + (b - a)z$, where z is a uniform pseudo-random number in the interval $[0,\ 1]$.

Randomness tests A review [21] has been carried out on the use of random numbers in solving problems using Monte Carlo Simulation. It draws conclusions concerning, in particular, the best methods to employ for generating uniform 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[9] and passing

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

Input parameters	
q	Number of uniform pseudo-random numbers to be generated
Input-output parameters	
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 realised 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 seeds 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 parameters	
z	q "draws" from a variable uniformly distributed between zero and one

Table 16: Uniform pseudo-random number generation.

a set of statistical tests for randomness.[10]

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 (see, e.g., [11, 17, 22, 24]) and are not covered here.

The KISS[11] generator [20] is a combination of a congruential generator and two shift register generators. A version in the C programming language is available [23, p42] and in Fortran [20].

The Hill-Wichmann generator [15] is a combination of three congruential generators. Both this and the KISS generator satisfy all the above standard and Die Hard tests [21] [23, p41] and are recommended. The authors of this document have favourable experience of applying the Hill-Wichmann generator.

[10] The tests include the so-called *standard tests* [17], 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* [19], that include the overlapping M-tuple test, the overlapping permutation test, the parking lot and lattice test and the birthday-spacing test.

[11] Keep It Simple, Stupid!

A recommended uniform random number generator Table 17 defines the Hill-Wichmann generator for generating uniform pseudo-random numbers in the interval [0, 1].

Input parameters	
None	
Input-output parameters	
i_A, i_B, i_C	Integer parameters required as input quantities and that are changed by the procedure. Set to integer values between 1 and 30000 before the first call. Do not disturb between calls. Subsequent values of these quantities are not usually of immediate concern to the user. The parameters provide the basis by which the pseudo-random numbers are produced. They may be realised as global variables and thus not explitly appear as parameters of the procedure
Output parameters	
Z	Uniform random number between zero and one
Computation	
1. Form $i_A = 171 i_A \bmod 30269$ 2. Form $i_B = 172 i_B \bmod 30307$ 3. Form $i_C = 170 i_C \bmod 30323$ 4. Form $Z = (i_A/30269 + i_B/30307 + i_C/30323) \bmod 1$	

Table 17: The Hill-Wichmann uniform pseudo-random number generator.

5.3.2 Gaussian distribution

The procedure in Table 18 provides a straightforwardly implementable approach [4] to generate values from the standardized Gaussian distribution $N(0, 1)$ using the Box-Muller transform.

To sample from the Gaussian distribution $N(\mu, \sigma^2)$, take $X = \mu + \sigma Z$, where Z is a standardized Gaussian variable.

5.3.3 Student's-t distribution

The procedure in Table 19 provides an approach [16], [23, p63] to generate values from the Student's-t distribution with ν degrees of freedom, that is

Input parameters	
None	
Output parameters	
Z_1, Z_2	Two draws from a Gaussian variable with zero mean and unit standard deviation
Computation	

1. Generate uniform random variates V_1 and V_2 between zero and one
2. Form $Z_1 = \sqrt{-2 \log V_1} \cos 2\pi V_2$ and $Z_2 = \sqrt{-2 \log V_1} \sin 2\pi V_2$
3. Take Z_1 and Z_2 as *two* standardized Gaussian variates

Table 18: The Box-Muller Gaussian pseudo-random number generator.

also straightforward to implement. No derivation is given here; full details are available [16].

Input parameters	
ν	Degrees of freedom
Output parameters	
Z	Draw from a Student's-t distribution with ν degrees of freedom
Computation	

1. Generate uniform random variates V_1 and V_2 between zero and one
2. If $V_1 < 1/2$, set $Z = 1/(4V_1 - 1)$ and $V = V_2/Z^2$; otherwise set $Z = 4V_1 - 3$ and $V = V_2$
3. If $V < 1 - |Z|/2$ or $V < (1 + Z^2/\nu)^{-(\nu+1)/2}$, accept Z as a Gaussian variate; otherwise repeat from Step 1

Table 19: A Student's-t pseudo-random number generator.

For a mean μ and standard deviation σ and ν degrees of freedom, take $X = \mu + \sigma Z$, where Z is a Student's-t variable with ν degrees of freedom.

Sampling from the Student's-t distribution can be used in Monte Carlo Simulation when synthesising pdf's assigned to input quantities represented by repeated observations that can be regarded as drawn from a Gaussian distribution (Section 2.1.1). This use has been criticized in some quarters in

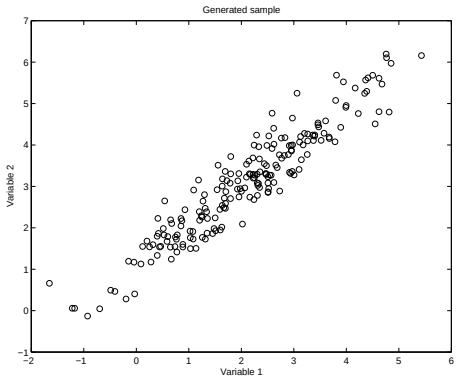

Figure 3: Points sampled from a bivariate Gaussian distribution with positive correlation.

that coverage intervals determined for the measurand can be pessimistically long. Conversely, were a Gaussian distribution to be assigned, i.e., with no account taken of the degrees of freedom that the Student's-t distribution is intended to accommodate, the resulting coverage intervals could be optimistically short. This matter remains a discussion topic by researchers. It is recommended that a decision is agreed and recorded with the details of the uncertainty evaluation in any one instance. In particular, despite the statement in the GUM that uncertainty evaluations should be realistic, if in doubt it might be wiser and certainly cautious to use the Student's-t distribution, accompanying the results of uncertainty evaluation by a suitable qualifying statement.

5.3.4 Multivariate Gaussian distribution

The most important multivariate distribution is the multivariate Gaussian distribution. An $n \times 1$ vector of means $\boldsymbol{\mu}$ and a covariance matrix V of order n define the parameters of the n-dimensional Gaussian distribution.

Values can be sampled from this multivariate (or joint) Gaussian distribution [25, 27] using the procedure in Table 20.

Figure 3 shows 200 points generated from a bivariate Gaussian distribution, using the MULTNORM generator [25], with mean $\boldsymbol{\mu} = (2, 3)^{\mathrm{T}}$ and covariance

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

Similar generators are available elsewhere [9].

Note that in this figure the points "span" an elongated angled ellipse. Were

Input parameters	
n	Dimension of the multivariate Gaussian distribution
$\boldsymbol{\mu}$	$n \times 1$ vector of means
V	Covariance matrix of order $n \times n$
q	Number of multivariate Gaussian pseudo-random numbers to be generated

Output parameters	
X	An $n \times q$ matrix, the jth column of which is a draw from the multivariate Gaussian distribution

Computation

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

2. Form q samples from the n-dimensional standardized Gaussian distribution $N(0,1) \times \ldots \times N(0,1)$. Here, $N(\mu, \sigma^2)$ denotes the Gaussian pdf with mean μ and standard deviation σ. Doing so simply means generating an $n \times q$ array Z of standard Gaussian variates

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

Table 20: A multivariate Gaussian random number generator.

the off-diagonal elements of V to be replaced by zero, the points would span a circle. Were the diagonal elements made unequal, and the off-diagonal elements kept at zero, the points would span an ellipse whose axes were parallel to the axes of the graph.

5.3.5 Sampling from discrete probability density functions

Discrete probability density functions arise when large numbers of repeated observations are to be used in their own right as a "definition" of their underlying distribution. They also arise in the form of Monte Carlo results from a previous uncertainty evaluation in a multi-stage process.

Suppose the ith input quantity is represented by a discrete probability density function in the form of values $x_{i,k}$, $k = 1, \ldots, q$. A Monte Carlo sample is given simply by selecting one of these values at random, with equal probability.

This process is often known as *re-sampling*.

Now suppose the n input quantities $\mathbf{X}$ are represented by a discrete probability density function in the form of values $x_{i,k}$, $k = 1, \ldots, q$, $i = 1, \ldots, n$. If the input quantities are mutually *independent*, a Monte Carlo sample is obtained by sampling independently from the n sets of values $\{x_{i,k} : k = 1, \ldots, q\}$ as above. The Monte Carlo sample is of the form $(x_{1,k_1}, \ldots, x_{n,k_n})^{\mathrm{T}}$ where $k_1, \ldots, k_n$ are integers drawn at random from $\{1, \ldots, q\}$. If the input quantities are mutually *dependent*, however, the n sets of values cannot be sampled independently. The Monte Carlo sample is of the form $(x_{1,k}, \ldots, x_{n,k})^{\mathrm{T}}$ where k is an integer drawn at random from $\{1, \ldots, q\}$.

The following paragraph indicates how a random integer can be chosen.

Generating a random integer When re-sampling it is necessary to "draw" a value at random from a set of q values $\mathbf{v} = (v_1, \ldots, v_q)^{\mathrm{T}}$, say, each having equal probability $1/q$. This task is accomplished by selecting v_ℓ, where ℓ is an integer drawn at random from the integers $\{1, \ldots, q\}$. Such an ℓ can be produced using

$$\ell = \lfloor qz + 1 \rfloor,$$

where z is a "draw" from the uniform distribution over $[0,\ 1]$ and $\lfloor w \rfloor$ denotes the largest integer that is no greater than w.

5.4 Evaluation of the model

The model is evaluated for each of the M samples from each of the n input pdf's. Specifically, denote the M samples by $\mathbf{x}_1, \ldots, \mathbf{x}_M$, where the ith sample $\mathbf{x}_i$ contains values $x_{i,1}, \ldots, x_{i,n}$, with $x_{i,j}$ a "draw" from the pdf for X_i. Then, for a univariate, explicit, real-valued model, the model values are

$$y_i = f(\mathbf{x}_i),\ i = 1, \ldots, M.$$

For a univariate, implicit, real-valued model, the ith model value y_i is determined as the solution to the equation

$$h(y_i, \mathbf{x}_i) = 0.$$

The case of a univariate, explicit, real-valued measurement model is covered below, although the presentation would apply similarly for the corresponding implicit model. Multivariate models are considered in Section 5.8.

The information in Section 3 provides advice on model evaluation.

Note that in Monte Carlo Simulation the model is evaluated for each sample of the input quantities and hence for a range of values (that may be a number of "standard deviations" away from the estimates of the input quantities). This is in contrast to the Mainstream GUM procedure in which the measurement model is evaluated only at the estimates of the input quantities. For this reason some issues may arise regarding the numerical procedure used to evaluate the model, e.g., ensuring its convergence (where iterative schemes are used) and numerical stability.

5.5 Measurement result and its standard uncertainty

The mean of the M values $y_1, \ldots, y_M$ of the output quantity is taken as the measurement result y. The standard deviation of these values is taken as the combined standard uncertainty $u_{\mathrm{C}}(y)$ of this measurement result. Table 21 specifies the evaluation of the measurement result and its (combined) standard uncertainty, and indicates the input and output parameters necessary for their determination.

The value y so obtained has the smallest standard deviation over all possible choices of measurement result. However, the value will not in general agree with the model evaluated at the estimates of the input quantities. Agreement (in a practical sense) will be achieved for a large value of M when the model is linear in the input quantities.

Whether this general lack of agreement is important depends on the application.

Input parameters	
M	Number of samples (equal to the number of Monte Carlo trials)
$\mathbf{y}$	Model values $(y_1, \ldots, y_M)^{\mathrm{T}}$ obtained by evaluating the measurement model for each of the M samples $\mathbf{x}_i$ from each of the n input pdf's, i.e., $y_i = f(\mathbf{x}_i)$
Output parameters	
y	Measurement result: the arithmetic mean of the model values, defined by $$y = \frac{1}{M}\sum_{k=1}^{M} y_i$$
$u_{\mathrm{C}}(y)$	Combined standard uncertainty: the standard deviation of the model values, defined by $$u_{\mathrm{C}}^2(u) = \frac{1}{M-1}\sum_{k=1}^{M}(y_i - y)^2$$
Numerical analysis	
	It is important to use the above formula for $u_{\mathrm{C}}(y)$ rather than the mathematically equivalent formula $$u_{\mathrm{C}}^2(y) = \frac{M}{M-1}\left(\frac{1}{M}\sum_{k=1}^{M} y_i^2 - y^2\right).$$ For cases in which $u_{\mathrm{C}}(y)$ is very much smaller than $\lvert y\rvert$ (in which case the $y_i,\ k = 1, \ldots, M$, 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 $u_{\mathrm{C}}(y)$ may have too few correct significant figures for the uncertainty evaluation to be valid [5]

Table 21: The measurement result and its standard uncertainty.

5.6 Distribution function of the output quantity

An estimate of the distribution function of the output quantity is obtained simply as follows. Order the values y_i of the output quantity provided by the Monte Carlo trials and assign probabilities to the ordered values that are uniformly spaced between zero and one [6]. Table 22 specifies the evaluation of an empirical estimate of the distribution function of the output quantity, and indicates the input and output parameters necessary for its determination.

	Input parameters
M	Number of samples (equal to the number of Monte Carlo trials)
$\mathbf{y}$	Model values $(y_1, \ldots, y_M)^{\mathrm{T}}$ obtained by evaluating the measurement model for each of the M samples $\mathbf{x}_i$ from each of the n input pdf's, i.e., $y_i = f(\mathbf{x}_i)$
	Output parameters
$\tilde{\mathbf{y}}$	Model values $(y_{(1)}, \ldots, y_{(M)})^{\mathrm{T}}$ sorted into non-decreasing order
$\mathbf{p}$	Probabilities $(p_1, \ldots, p_M)^{\mathrm{T}}$, defined by $$p_i = (i - 1/2)/M.$$ The function $p = G(y)$ determined as the join of the points $(y_{(i)}, p_i)$, $i = 1, \ldots, M$, provides an empirical estimate of the distribution function of Y. ($G(y)$ is defined only for values of y 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.)
	Numerical analysis
	It is recommended that a sorting algorithm that takes a number of operations proportional to $M \log M$ be used (e.g., [26]). A naive algorithm would take a time proportional to M^2 and may make the computation time unacceptable

Table 22: The distribution of the output quantity.

5.7 Coverage interval

Once the distribution function $G(y)$ has been estimated, almost any statistical information associated with the measurand can be determined.

In particular, a 95% coverage interval $(y_{\mathrm{low}},\ y_{\mathrm{high}})$ for the measurand is given by the 0.025- and 0.975-fractiles of $G(y)$, i.e., the values of y given by $G^{-1}(0.025)$ and $G^{-1}(0.975)$.

For practical purposes, $(y_{(\lfloor 0.025M \rfloor)}, y_{(\lceil 0.975M \rceil)})$, where $\lfloor v \rfloor$ is the largest integer no greater than v and $\lceil v \rceil$ is the smallest integer no smaller than v, can equivalently be taken as this interval.

5.8 Multivariate Monte Carlo

The above treatment applies to univariate real-valued models. It can straightforwardly be adapted to apply Monte Carlo Simulation to multivariate or complex-valued models. There is not, however, a ready counterpart of a coverage interval.

The multivariate model is

$$\mathbf{Y} = \mathbf{f}(\mathbf{X}).$$

M samples $\mathbf{x}_i$ of the inputs $\mathbf{X}$ are taken as before. For each $\mathbf{x}_i$, evaluate the model as previously, except now the output values $\mathbf{y}_i = \mathbf{f}(\mathbf{x}_i)$ are $m \times 1$ vectors.

Assemble these output vectors into an $m \times M$ matrix[12]:

$$\Upsilon = (\mathbf{y}_1, \ldots, \mathbf{y}_M).$$

From this matrix the covariance matrix $V_{\mathbf{y}}$ is estimated from

$$V_y = \frac{1}{M-1} \Upsilon' (\Upsilon')^{\mathrm{T}},$$

where Υ' is Υ corrected for the sample means over all M trials, i.e., with the mean of the jth row subtracted from all elements in that row, for $j = 1, \ldots, M$. Table 23 specifies the evaluation of the covariance matrix V_y in terms of the output vectors $\mathbf{y}_i$, 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 Mainstream GUM. (In fact, it provides more than Mainstream GUM, since that procedure does not in general cover multivariate models.) The matrix Υ provides much richer information, however, in the following sense. Any column of Υ corresponds to the values of the outputs for one choice (sample) of the inputs. *Any* derived quantity can be determined from this single set of output values. This quantity can be calculated for all columns, the resulting $1 \times M$ row vector providing an empirical sampling distribution for that quantity. In particular, fractiles of the values in that vector can be used, as above, to

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

Input parameters	
m	Number of output quantities
M	Number of samples (equal to the number of Monte Carlo trials)
Υ	An $m \times M$ matrix containing the model output values $\mathbf{y}_i$, i.e., $$\Upsilon = (\mathbf{y}_1, \ldots, \mathbf{y}_M).$$
Output parameters	
V_y	Covariance matrix for the output quantities $\mathbf{Y}$, defined by $$\frac{1}{M-1}\Upsilon'(\Upsilon')^{\mathrm{T}},$$ where Υ' is Υ corrected for the sample means

Table 23: The covariance matrix for the output quantities $\mathbf{Y}$.

provide a coverage interval for the derived quantity. 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 Υ is a very valuable array, being the basis of almost unlimited statistical information.

6 Validation of Mainstream GUM

Mainstream GUM has some limitations [1, 6]. Although the procedure can be expected to work well in many circumstances, it is difficult to quantify the effects of the approximations involved, viz., linearization, the Welch-Satterthwaite formula and the assumption that the output quantity is Gaussian (i.e., that the Central Limit Theorem is applicable). Therefore, it is important to be able to validate any cases of doubt. In such instances it is recommended that both Mainstream GUM and the Monte Carlo approach are applied and the results compared. If the comparison is favourable, Mainstream GUM can be used for sufficiently similar problems in the future. Otherwise, consideration can be given to using instead Monte Carlo Simulation or some other approach that is free from the above limitations.

Specifically, the two steps below should be carried out, followed by the comparison process of Section 6.1.

1. Apply Mainstream GUM to yield a 95% coverage interval $y \pm U(y)$ for the measurand.

2. Apply Monte Carlo Simulation to yield a standard uncertainty $u(y)$ for the measurement result and the endpoints y_{low} and y_{high} of a 95% coverage interval for the measurand.

6.1 Comparison of results

A recommended comparison procedure is based on the following objective: determine whether the coverage intervals for Mainstream GUM and Monte Carlo Simulation agree to a specific absolute accuracy. This accuracy is assessed in terms of the endpoints of the coverage intervals and corresponds to that given by rounding the standard uncertainty of the measurement result to two (say) *significant decimal digits*.

The procedure is as follows:

1. Determine r to be the number of figures after the decimal point in $u(y)$, after this value has been rounded to two, say, significant decimal digits. Let
$$\delta = \frac{1}{2}10^{-r}.$$

2. Compare the coverage intervals obtained by Mainstream GUM and Monte Carlo Simulation to determine whether the required number of correct digits in the Mainstream GUM coverage interval has been obtained. Specifically, determine the quantities
$$|y - U(y) - y_{\text{low}}|$$
$$|y + U(y) - y_{\text{high}}|,$$
viz., the absolute values of the differences of the respective endpoints of the two coverage intervals. Then, if both these quantities are no larger than δ the comparison is successful and Mainstream GUM has been validated in this instance.

Example. The measurement result for a nominally 100 g standard of mass [1, Clause 7.2.2] is $y = 100.021\ 47$ g and the standard uncertainty of y, rounded to two significant figures, is $u(y) = 0.000\ 35$ g, The number of digits in $u(y)$ after the decimal point is $r = 5$. Thus, the value $\delta = 5 \times 10^{-6}$ should be used in the comparison.

7 Conclusions

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

1. The Mainstream GUM ("uncertainty budgeting") procedure of the Guide to the Expression of Uncertainty in Measurement [1],

2. A general approach based on Monte Carlo Simulation, that will form the basis of a supplement to the Guide to the Expression of Uncertainty in Measurement [6], and

3. Mainstream GUM validation, the use of Monte Carlo Simulation to validate Mainstream GUM in circumstances where there is doubt concerning its applicability [6].

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

This document will be revised and extended as part of the ongoing Software Support for Metrology programme.

Acknowledgements

This report constitutes part of the deliverable of Project 1.2, "Uncertainties and statistical modelling" within the UK Department of Industry's National Measurement System *Software Support for Metrology* Programme 1998–2001.

The National Engineering Laboratory made input to this work.

References

[1] Guide to the Expression of Uncertainty in Measurement. International Organisation for Standardisation, Geneva, Switzerland, 1995. ISBN 92-67-10188-9, Second Edition.

[2] R. Barker. Software Support for Metrology Best Practice Guide No. 5. Software Re-use: Guide to MetroS. Technical report, National Physical Laboratory, Teddington, UK, 2000.

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

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

[5] M. G. Cox, M. P. Dainton, and P. M. Harris. Testing functions for the calculation of standard deviation. Technical Report CMSC 07/00, National Physical Laboratory, Teddington, UK, 2000.

[6] M. G. Cox, M. P. Dainton, and P. M. Harris. Software Support for Metrology Best Practice Guide No. 6. Uncertainty and Statistical Modelling. Technical report, National Physical Laboratory, Teddington, UK, 2001.

[7] M. G. Cox and P. M. Harris. Guidelines to help users select and use software for their metrology applications. Technical Report CMSC 04/00, National Physical Laboratory, Teddington, UK, 2000.

[8] 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.

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

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

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

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

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

[14] I. A. Harris and F. L. Warner. Re-examination of mismatch uncertainty when measuring microwave power and attenuation. *Proc. IEEE*, 128:35–41, 1981.

[15] I. D. Hill and B. A. Wichmann. Algorithm AS183. An efficient and portable pseudo-random number generator. *Appl. Statist.*, 31:188–190, 1982.

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

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

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

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

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

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

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

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

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

[25] 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.

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

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

[28] 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, 1994.

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

A Use of symbolic-algebra packages

This appendix is concerned with the use of a symbolic-algebra package to provide partial derivatives of a model, from which sensitivity coefficients can be determined (Section 4.2).

Table 24 specifies for a univariate, explicit, real-valued model, 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 model.

Counterparts would apply for the other categories of measurement model.

Input parameters	
n	Number of input quantities
f	Function specifying the model $Y = f(\mathbf{X})$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$. Some packages may not be able to make use of this information in that they return algebraic expressions for $(\partial f/\partial X_1, \ldots, \partial f/\partial X_n)$ rather than these expressions evaluated at $\mathbf{X} = \mathbf{x}$
Output parameters	
$\mathbf{f}'$	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
C	A $1 \times n$ vector of sensitivity coefficients, whose jth element is the partial derivative $\partial f/\partial X_j$ of $f(\mathbf{X})$ evaluated at $\mathbf{X} = \mathbf{x}$. Some symbolic-algebra packages may produce just $\mathbf{f}'$ and it will be the user's responsibility to evaluate $\mathbf{f}'$ at $\mathbf{X} = \mathbf{x}$

Table 24: Sensitivity coefficients obtained using a symbolic-algebra package.

When a symbolic-algebra package system is used, care needs to be taken that the mathematical expressions generated are stable with respect to their evaluation at the estimates of the input quantities. For instance, suppose that (part of) a model is

$$Y = (X_1 - K_0)^4,$$

where K_0 is a specified constant. An automatic system might involve expansions such as Taylor series to generate the partial derivative of Y in the form

$$\frac{\partial Y}{\partial X_1} = 4X_1^3 - 12X_1^2K_0 + 12X_1K_0^2 - 4K_0^3, \tag{2}$$

and perhaps not contain a facility to generate directly or simplify this

expression to the *mathematically* equivalent form

$$\frac{\partial Y}{\partial X_1} = 4(X_1 - K_0)^3, \tag{3}$$

that would typically be obtained *manually*.

Suppose the estimate of X_1 is $x_1 = 10.1$ and $K_0 = 9.9$. The value c_1 of the resulting sensitivity coefficient is $4(x_1 - K_0)^3 = 0.032$, correct to two significant figures. Both formulae (2) and (3) deliver this value to this number of figures. The second, more compact, form is, however, much to be preferred. The reason is that Formula (3) suffers negligible loss of numerical precision when it is used to evaluate c_1, whereas, in contrast, Formula (2) loses figures in forming this value. To see why this the case, consider the contributions to the expression, evaluated and displayed here to a constant number of decimal places (corresponding to seven significant figures in the contribution of greatest magnitude):

$$\begin{aligned} 4x_1^3 &= 4121.204, \\ -12x_1^2K_0 &= -12118.79, \\ 12x_1K_0^2 &= 11878.81, \\ -4K_0^3 &= -3881.196. \end{aligned}$$

The sum of these values constitutes the value of c_1. To the numerical accuracy held, this value is 0.028, compared with the correct value of 0.032. The important point is that a value of order 10^{-2} has been obtained by the sum of positive and negative values of magnitude up to order 10^4. Almost inevitably, a loss of some six figures of numerical precision has resulted, as a consequence of *subtractive cancellation*.

For different values of x_1 and K_0 or in other situations the loss of figures could be greater or less. The concerning matter is that this loss has resulted from such a simple model. The effects in the case of a sophisticated model or a multi-stage model [6] could well be compounded, with the consequence that there are dangers that the sensitivity coefficients formed in this way will be insufficiently accurate. Therefore, care must be taken in using sensitivity coefficients that are evaluated from the expressions provided by some software for algebraic differentiation. Such a system, if used, should evidently be chosen with care. One criterion in making a choice is whether the system offers comprehensive facilities for carrying out algebraic simplification, thus ameliorating the danger of loss of figures. Even then, some form of validation should be applied to the numerical values so obtained.[13]

Some systems operate in a different way. They "differentiate code" rather

[13] Numerical analysis issues such as that discussed here will be addressed as part of SSfM-2.

than differentiate a formula. Acceptable results can be produced by the user by carefully constructing the code that defines the function.

B Use of finite-difference formulae

This appendix is concerned with the use of a finite-difference formula to provide partial derivatives of a model, from which sensitivity coefficients can be determined (Section 4.2).

Numerical approximations to the values of derivatives can be obtained using finite-differences techniques. Given a value i ($1 \leq i \leq n$), set all $X_\ell = x_\ell$, apart from X_i, i.e., assign the estimated values of the inputs to the input quantities, apart from the ith. Denote the resulting function of X_i by $f_i(X_i)$.

A typical finite difference approximation to $\partial Y/\partial X_i$ evaluated at $\mathbf{x}$ is

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

where δ_i is a "suitably small" increment in x_i (see below). Note that $f_i(x_i) \equiv f(\mathbf{x})$ will already have been formed in evaluating the model at the estimates $\mathbf{x}$ 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 figures.

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

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

For a given value of δ_i, the magnitude of the approximation error is generally 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 exceptional circumstances.

Table 25 specifies, for a univariate, explicit, real-valued model, 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 about the measurement model, and may take the form of a function for evaluating the 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 model $Y = f(\mathbf{X})$ in terms of the input quantities $\mathbf{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$
$\mathbf{x}$	Column vector $(x_1, \ldots, x_n)^{\mathrm{T}}$ of estimates of the input quantities $\mathbf{X}$
$\mathbf{u}$	Column vector $(u_1, \ldots, u_n)^{\mathrm{T}} \equiv (u(x_1), \ldots, u(x_n)^{\mathrm{T}}$ of standard uncertainties of 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 used that attempted to provide a sensible compromise between the loss of accuracy due to trancation (approximation) error and that due to subtractive cancellation
Output parameters	
C	A $1 \times n$ vector of sensitivity coefficients, whose jth element is an estimate obtained using finite differences of the partial derivative $\partial f / \partial X_j$ of $f(\mathbf{X})$ evaluated at $\mathbf{X} = \mathbf{x}$

Table 25: Sensitivity coefficients obtained using a finite-difference formula.