

Symbolic Computation for Evaluation of Measurement Uncertainty

Peng Wei, Qing Ping Yang, Mohd Rizal Salleh, Barry Jones
School of Engineering and Design, Brunel University, UB8 3PH, United Kingdom
Phone: 44 1895 265854, Fax: 44 1895 269763, Email: empgpw@brunel.ac.uk

Abstract - In recent years, with the rapid development of symbolic computation, the integration of symbolic and numeric methods is increasingly applied in various applications. This paper proposed the use of symbolic computation for the evaluation of measurement uncertainty. The general method and procedure are discussed, and its great potential and powerful features for measurement uncertainty evaluation has been demonstrated through examples.

Keywords - Uncertainty of Measurement, Evaluation of Measurement Uncertainty, Symbolic Computation, Numeric Computation.

1. INTRODUCTION

Since the publication of the ISO Guide to the Expression of Uncertainty in Measurement (GUM) in 1993 [1], the importance of measurement uncertainty and its evaluation has been increasingly recognized. According to the GUM, the mathematical model for the measurand concerned plays a key role in the evaluation of measurement uncertainty. In deed, the definition of sensitivity coefficient of each uncertainty source is based upon partial derivative, which generally implies the use of mathematical models.

To facilitate the evaluation and automate the computation, several software tools have become commercially available to assist practical evaluations [2]. Most of these software tools are based on pure numerical methods using traditional programming languages (e.g. C/C++) or spreadsheets (e.g. Microsoft Excel) [2-3]. Despite its overwhelming popularity for scientific computing, pure numerical methods have some considerable limitations in uncertainty evaluation.

The major limitations of pure numeric computation in uncertainty evaluation include the following:

1)The mathematical model is the starting point and the key to the uncertainty evaluation. It is highly desirable to allow the use of any common symbols or user-defined symbols directly in the representation of the mathematical model. Uncertainty evaluation using pure numeric computation often limits the use of symbols or even only allows the symbols predefined by the software.

2)This also applies to the intermediate calculations of individual standard uncertainty. The use of user defined symbols for intermediate calculations is not allowed or limited in most of the existing uncertainty evaluation tools.

3)The numerical computation of sensitivity coefficients usually calculates the partial derivatives according to the definition, and the approximate result is thus sensitive to the

algorithms and rounding errors. Further it requires that the values of all the variables are already known, and the computation will have to be repeated if any of the values changes.

4)The results of the calculation can be directly affected by errors in the previous steps, which can trigger a chain reaction for the following steps [4].

5)Numeric computation also makes it less obvious to see the influences and contribution of various factors to the final measurement uncertainty.

The above mentioned limitations of pure numeric computation tend to make uncertainty evaluation appear more difficult and less user friendly. The direct evidence of these limitations can be easily seen from the limited capacity for expressing and processing the mathematical models in the existing software tools. They are also likely to impede the wide application of uncertainty evaluation.

The main difficulty lies in that the mathematical model for the measurement varies in each case, and that it has to be defined by the user at the run time and can not, in general, be predefined in the software.

2. PROPOSED APPROACH

This paper proposes and discusses the integration of symbolic computation and numeric computation in uncertainty evaluation.

Symbolic computation as a completely different analytical method has been increasingly applied as an alternative computing tool [5]. Compared with numeric computation, symbolic computation represents and manipulates information in symbolic form, i.e. it directly processes the symbols rather than the values of all the variables in the calculation.

Symbolic computation aims at the automation of mathematical calculation and is able to overcome most of the above limitations associated with numeric computation with the following advantages:

1)It simplifies mathematical calculations and makes it natural to use mathematical expressions and models.

2)It puts more emphasis on the analytical process than pure numerical methods, making the logic easily understood. [4]

3)This computation approach requires as little input data as possible.

More specifically, the authors proposed the use of symbolic computation for the mathematical model of

measurement and its subsequent processing. The symbolic representation of the mathematical model is crucial since both the user and the software can readily understand. Common mathematical functions and expressions can be directly entered and understood by the symbolic computation engine.

The calculation of sensitivity coefficients is a very important step in measurement uncertainty evaluation. They are defined as partial derivatives with respect to individual input variables. Their derivation from mathematical models may not be very straightforward for many users. Symbolic computing can generate the sensitivity coefficients automatically, thus significantly simplify the evaluation process.

The uncertainty evaluation will have to generate numerical values, but symbolic computation can minimize the use of numeric calculations, usually only in the final step. The intermediate steps can all be calculated by symbolic computation, which could eliminate possible calculation errors in the intermediate steps.

Ultimately, the use of symbolic computation will make the uncertainty evaluation easier and more user-friendly, facilitating its use in more applications.

3. RESULTS

Symbolic computation was implemented using the mathematical software, Maple 10. The feasibility of performing evaluation of measurement uncertainty using symbolic computation has been studied through examples, initially created as Maple worksheets.

The worksheet is composed of three main sections, as illustrated in Fig. 1, which shows an example concerned with DC current measurement [6].

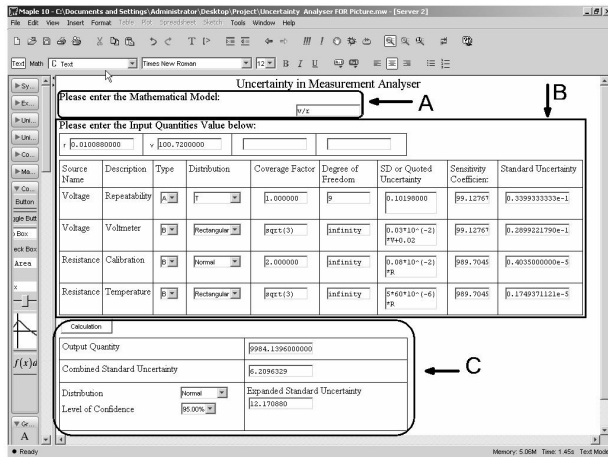


Fig. 1. Worksheet Main Interface

Section A

This section is used to define the mathematical model. The user is required to enter the mathematical model in the

‘Mathematical Expression Component’ according to the measurement process concerned. Variables can be represented by any symbol, either commonly defined or user defined. Symbols for standard operations can be directly used.

In this DC current measurement case, the measured current I is calculated according to Ohm’s law,

$$I = f(V, R) = \frac{V}{R} \quad (1)$$

where V is the voltage, and R is the resistance.

The user can directly type any valid mathematical expression in this section (Fig. 2), using standard symbols,

e.g. $\frac{V}{R}$ or user defined symbols e.g. $\frac{A}{B}$.

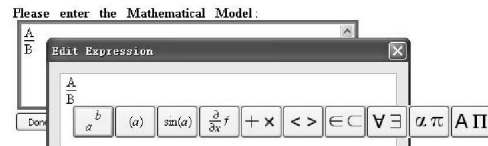


Fig. 2. Defining Mathematical Model

Maple 10 provides three commands which can be used to extract symbols from an expression:

- *indets* - find indeterminates of an expression
- *op* - extract operands from an expression
- *nops* - number of operands of an expression

Using the commands above, input quantities can be extracted from the mathematical expressions entered by the user.

The general method of symbolic evaluation of measurement uncertainty is as follows:

1) Assign the mathematical expression to a variable, *model*.

2) Extract the variable symbols from the expression stored in *model* using the *indets* command and save them in an array *z*.

3) Calculate all the sensitivity coefficients
 $C_1 = \text{diff}(\text{model}, z[1])$

...

$C_n = \text{diff}(\text{model}, z[n])$

If the final numerical results are required,
 $C_1 = \text{eval}(\text{diff}(\text{model}, z[1]), z[1]=x_1)$

...

$C_n = \text{eval}(\text{diff}(\text{model}, z[n]), z[n]=x_n)$

where *n* is the number of input variables; *z*[*i*] is the variable in the array *z*; *x*_{*i*} is the value of the input variable entered by the user.

4) Calculate the individual standard uncertainties (*u*₁, *u*₂, ..., *u*_{*n*}) according to the user inputs for each uncertainty source.

5) Combine all the uncertainty sources together using the sensitivity coefficients and standard uncertainties calculated above.

6) Calculate the expanded uncertainty.

In symbolic computation, the symbols themselves will directly enter the subsequent computations in stead of their values.

The calculation is automatic and easily scalable to deal with more complex mathematical models. For instance, in Torque Tester Calibration [6], the mathematical model is

$$T_m = M \cdot \left(1 - \frac{\rho_a}{\rho_m}\right) \cdot g \cdot d \cdot (1 + \alpha \cdot \theta) + e \quad (2)$$

Once the model is defined, the rest of the computation remains the same. The details of derivation and computing process are hidden from the user and the use of numeric computation is minimized. Numeric computation is only used when it is necessary to produce the final numerical values.

Of course it is possible to display any intermediate calculation result, e.g. the sensitivity coefficient of the first input quantity M is given by:

$$C_1 = \text{diff}(T_m, z[1]) = \text{diff}(T_m, M)$$

$$C_1 = \left(1 - \frac{\rho_a}{\rho_m}\right) \cdot g \cdot d \cdot (1 + \alpha \cdot \theta) \quad (3)$$

The results in symbolic form show clearly the influences of all the input variables.

Section B

This section collects the information about each individual source of uncertainty, including type of evaluation, probability distribution, coverage factor, degree of freedom, standard deviation and quoted uncertainty. The user can choose different options and enter numeric values or expressions using symbols.

Measurement data can be directly entered into section B. In the DC current measurement case, ten measurements of voltage can be entered as an array, i.e. voltage=[100.68, 100.83, 100.79, 100.64, 100.63, 100.94, 100.60, 100.68, 100.76, 100.65], its mean can be calculated as `'describe[mean](voltage)'` and its Standard Deviation: `'describe[standarddeviation](voltage)'`;

For Type A evaluation, the degree of freedom depends upon the number of measurements with a coverage of 1. For Type B evaluation, coverage factor and degree of freedom may be determined from its probability distribution.

Numerical or symbolic expressions can be entered into the quoted uncertainty cell. For Type A evaluation it is simply the standard uncertainty or standard deviation. In the DC current measurement example, the quoted uncertainty for the voltage measurement can be entered as `'0.03%*V+0.02'`.

The Sensitivity Coefficients and Standard Uncertainty in the last two columns can be calculated separately for each uncertainty source or together with the final evaluation results in section C.

Section C

This section displays the results of the evaluation, including the mean value of the output quantity or the measurand, combined standard uncertainty and the expanded uncertainty. The user can also choose the level of confidence and distribution for the combined standard uncertainty. The results can be updated by clicking the Calculation button.

The combined standard uncertainty of various input quantities is generally given as,

$$u_c^2(y) = \sum_{i=1}^n \sum_{j=1}^n \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} u(x_i, x_j) \quad (4)$$

The background process will calculate the combined standard uncertainty, together with all symbols in Section A and section B. In the DC current measurement, the combined standard uncertainty is:

$$u_c^2\left(\frac{V}{R}\right) = \left(\frac{\partial}{\partial V}\left(\frac{V}{R}\right)\right)^2 u_1^2(V) + \left(\frac{\partial}{\partial R}\left(\frac{V}{R}\right)\right)^2 u_2^2(V) + \left(\frac{\partial}{\partial R}\left(\frac{V}{R}\right)\right)^2 u_3^2(R) + \left(\frac{\partial}{\partial R}\left(\frac{V}{R}\right)\right)^2 u_4^2(R) \quad (5)$$

$$= \left(\frac{1}{R}\right)^2 u_1^2(V) + \left(\frac{1}{R}\right)^2 u_2^2(V) + \left(-\frac{V}{R}\right)^2 u_3^2(R) + \left(-\frac{V}{R}\right)^2 u_4^2(R)$$

Through symbolic computing, the use of symbols and related mathematical expressions is maximized with simplified and automatic calculation procedures. The symbolic results can highlight the relationship between the input quantities and the contribution of various factors to the final measurement uncertainty.

4. CONCLUSIONS

Expressing the mathematical model symbolically, it is able to process the model and evaluate measurement uncertainty both symbolically and numerically.

The symbolic evaluation of measurement uncertainty is natural, easy to use, and powerful to handle complicated cases. Its great potential for measurement uncertainty evaluation has been demonstrated through examples.

Although this paper discussed the symbolic computation using Maple's powerful mathematical computing engine, it is possible to use other technologies for symbolic computation. For example, Java is capable of symbolic computing [7]. With its excellent portability across different computer platforms and network, it is very suitable for symbolic evaluation of measurement uncertainty. This will offer many advantages and create novel applications.

ACKNOWLEDGMENT

The authors would like to thank T. Li who provided valuable technical support and suggestions.

REFERENCES

- [1] ISO, "Guide to the Expression of Uncertainty in Measurement (GUM)", Geneva, Switzerland, 1993. (Reprinted 1995).
- [2] S. N. Rasmussen, Danish Technological Institute, "Software tools for the expression of uncertainty in measurement", Metro Trade Workshop on Traceability and Measurement Uncertainty in Testing, Berlin, January 30-31, 2003.
- [3] B. D. Hall, "Uncertainty calculations using object-oriented computer modelling", *Meas. Sci. Technol.* **10** 380-386 doi:10.1088/0957-0233/10/5/307, 1999.
- [4] I. Munteanu, D. Ioan, "Symbolic Computation with Maple V for Undergraduate Electromagnetics", *IEEE Transactions on Education*, Vol. 44, No. 2, pp. 3, 2001.
- [5] A. Juozapavičius, "Symbolic Computation: Systems and Applications", *Nonlinear Analysis: Modelling and Control*, Vilnius, IMI, Vol 3, 1998.
- [6] Singapore Institute of Standards and Industrial Research, "Guidelines on the Evaluation and Expression of the Measurement Uncertainty", pp. 45-54, 1995.
- [7] L. Bernardin, B. Char and E. Kaltofen, "Symbolic Computation in Java: an Appraisalment". *Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation*, pp. 237-44, 1999.