A new approach to CMM software validation

A Forbes[1], P Harris[1], F Haertig[2], U Lunze[3], I Smith[1] and K Wendt[2]
[1]*National Physical Laboratory, UK*
[2]*Physikalische-Technische Bundedanstalt, Germany*
[3]*Wests¨achsiche Hoschshule Zwickau, Germany*

Abstract

Traceability requires that measurement results can be linked to references (such as measurement units) through a documented unbroken chain. If the chain involves computation, as it does in almost all modern measuring systems, it is necessary that all computational links are recognised explicitly and known to be operating correctly. Considerable effort has been and continues to be dedicated to software validation to assess the performance of software. To date, however, activity has mainly involved the offline assessment of software. This paper describes a current European Metrology Research Programme Joint Research Project (JRP) that will deliver a new approach, providing an online capability for software validation and certification, giving users the potential to validate software every time it is used. While the JRP will deliver an online validation system for software from any metrology domain, there is a focus on applications from coordinate metrology. The new approach, and its application to fitting geometric elements and freeform surfaces to data according to the least squares (Gaussian) and minimum zone (Cheby-shev) criteria, are described.

1 Introduction

A key requirement of traceability is that measurement results can be linked to references, e.g., measurement units, through a documented unbroken chain. To an ever-increasing degree, such chains nowadays involve computation, and it is important that computational links are known to be operating correctly. Analogous to physical artefacts are reference data sets (sometimes referred to as ‘numerical artefacts’) that may be used to test that software components in a measurement chain are operating correctly.

The requirement for traceability is particularly acute in coordinate metrology involving complex geometries. Since there are of the order of 40 000 coordinate measuring machines in the EU alone, each relying on complex algorithms to measure a range of artefacts, the impact of poorly performing software is potentially large. In both Europe and the USA there have been initiatives in software validation using data sets to assess the performance of CMM software for fitting geometric elements to data. These activities have primarily involved the offline assessment of the software.

The Joint Research Project (JRP) NEW06 'Traceability for computationally-intensive metrology' (referred to as 'TraCIM') [8, 16] is currently being funded by the European Metrology Research Programme (EMRP) [5]. The main objective of the JRP, which runs from June 2012 to May 2015, is to deliver new technology that will allow software from any metrology domain to be validated at point of use. The JRP-consortium brings together significant expertise in the fields of software validation, information and communications technologies (ICT), and coordinate metrology, and comprises National Metrology Institutes (NMIs) from the UK, Germany, The Netherlands, Italy, Poland and Slovenia, industrial partners Hexagon, Mitutoyo, Werth and Zeiss and unversities Westsächsische Hochschule Zwickau, Ostfalia Hochschule für angewandte Wissenschaften, Huddersfield and York.

The aim of the JRP is to develop a generic ICT infrastructure that in principle can be used to validate software from any metrology area. JRP-Partners undertook a review of the metrology areas in which computation plays a critical part. This review, together with the JRP's focus on coordinate metrology, resulted in ten priority application areas being identified. From the length domain, these are Least squares geometric element fitting, Chebyshev geometric element fitting, Evaluation of surface texture parameters, and Least squares non-uniform rational B-splines (NURBS) fitting.

2 Specification of computational aims

In order to undertake verification and validation of software, there must exist a clear statement of the problem that the software is intended to solve or the task that the software is intended to execute. This statement is essential both to act as the user and functional requirements for the software developer, and to provide a basis for verification and validation of the software implementation.

JRP-Partners have developed a procedure that provides a clear description of how a computational aim should be specified. The specification of the computational aim includes information contained in the following fields: Title; Keywords; Mathematical area; Dependencies; Input parameters; Output parameters; Mathematical model; Signature; Properties; References. The ideal situation is that the specifications are given in documentary standards, and are agreed by the international community. In the length domain, many of these calculations are specified to a lesser or greater degree of ambiguity in standards, e.g., [9, 10, 11, 12], but one of the outputs of the JRP is set in motion the development of new standards that provide very concrete specifications of the required computations relating to Chebyshev and freeform surface fitting, for example. Project partner, the University of York, is also looking a formal specification languages that encode the computational aims. A repository for specifications of computational aims is provided by a computational aims database [17]. The database has been populated by specifications of computational aims for each of the ten priority application areas.

3 Generation of reference data

A common approach to testing software implementations of computational aims involves the use of data generators [1, 4]. A data generator, also implemented in software, though not necessarily in the same programming language as the software under test, is used to produce a reference pair, comprising reference input data and reference output data. The reference input data is processed by the software under test to produce test output data that is compared (in an appropriate way) with the reference output data. Repeating this process a number of times on different datasets allows a statement to be made about the quality of the software under test.
Data generators typically adopt one of the following approaches:

- Forward data generation, which refers to the process of taking reference input data and using it to produce corresponding reference output data.
- Reverse data generation, which refers to the process of taking reference output data and using it to produce corresponding reference input data [1, 2, 3, 6, 7].

To implement forward data generation requires the use of reference software that processes input data to produce output data. The development of such software is often complicated and costly. Reverse data generation typically requires an understanding or analysis of the computational aim such that output data can be processed to produce input data. In many cases reverse data generation is more simple to implement than forward data generation, particularly since the need to develop reference software is avoided.

Generation of reference data plays a key role in the provision of the TraCIM software validation service. Typically in coordinate metrology, software will be developed that is intended to solve only a limited subset of the possible fitting problems. For example, a piece of software may be designed to apply to data all of which lie within a particular volume, or there may be a constraint on the (upper limit of the) number of points that the software can process, or uncertainty information associated with the measured data may be available. Data generation processes should be implemented to be able to generate reference data sets having specific properties that are relevant to the software for which they will be used to undertake validation. It is important that reference data sets are generated that resemble those that are likely to occur in reality.

3.1 Least squares fitting

Approaches for generating reference data using inverse data generation for least squares orthogonal distance regression (LSODR) for geometric elements are relatively well-known. Given reference output data comprising values for the parameters that define the shape and position of a geometric element, the null space method [3, 7] can be used to determine corresponding reference input data.

Reverse data generation for fitting freeform surfaces according to the least squares criterion are described in [6]. The inputs to the data generation are simply ξ' points on the surface of the artefact specified by parameters α along with the normal vectors ν at ξ . From this information, points ξ are generated according to $\xi = \xi + 6$ ν where the 6 are drawn at random but satisfy conditions that ensures the best fit surface is given by parameters α. The 6 can be chosen to represent typically form error that occurs in practice. A general approach is to ensure the form errors are spatially correlated so that if ξ' is close to ξ'_3 then 6 and 63 are statistically correlated so that 6 is likely to be close to 63 [7, 15]. Customised data sets, e.g., that incorporate a particular type of form error or that include systematic error behaviour of a CMM, can also be generated straight forwardly.

3.2 Minimum zone fitting

Calculation of minimum zone geometric elements is a common task for CMM software, with many applications in tolerance assessment relating to international standards reliant upon it [10]. Chebyshev orthogonal distance regression (ChODR) problems are much more difficult to solve than their Gaussian counterparts, and correspondingly, the generation of reference data for ChODR is also more complex than that for LSODR. One particular complication concerns ensuring that the generated solution to a ChODR problem is a global solution and not just a local one.

JRP-Partners have developed a two-step approach for data generation that combines aspects of forward and reverse data generation. In the first step, reverse data generation, based on a mathematical analysis of the computation aim [7], is applied to reference output data to determine a candidate for reference input data. In the second step, independent reference software is applied to the reference input data. If the reference software does not identify a ‘better’ solution, the reference input and output data are considered to form a valid reference pair that may then be used by the software validation service.

4 Development of performance metrics

In order to evaluate the performance of software under test, metrics appropriate to the computational aim being addressed by the software must be developed. Such metrics should take into account:

- The numerical uncertainty associated with the reference data, i.e., how close the reference input and output data is to the true mathematical solution.
- Characteristics of the measurement data likely to arise in practice, such as the simulated measurement uncertainty associated with the reference input and/or output data.
- A maximum permissible error (MPE), or other specification, that applies in the relevant metrology domain.

Issues to do with the uncertainties associated with the reference data are considered by [13, 14]. Suppose the solution outputs α are sufficiently smooth function $\alpha = A(\xi)$ of the input data ξ, so that A has continuous first derivatives with respect to x_i. Let S be the $n \times m$ sensitivity matrix with $S_{ji} = \frac{\partial \alpha_j}{\partial \psi_i}$, $i = 1, \ldots, m$, and $j = 1, \ldots, n$. For many problems, the conditions that $\alpha = A(\xi)$ can be written as a set of equations involving ξ and α of the form $\gamma(\xi, \alpha) = 0$. These conditions define α implicitly as a function of ξ. In this case, we have

$$HS + J^T = 0, \quad J_{ik} = \frac{\partial g_k}{\partial x_i}, \quad H_{kj} = \frac{\partial g_k}{\partial a_j},$$

so that $S = -H^{-1}J^T$.

For the case of LSODR with geometric elements, the optimality conditions are $\gamma(\{\xi_i\}, \alpha) = 0$ with

$$g_k(\{\xi_i\}, \alpha) = \sum_{i=1}^{m} d_i \frac{\partial d_i}{\partial a_k}, \quad d_i = d(\xi_i, \alpha),$$

$$\frac{\partial g_k}{\partial a_j} = \sum_{i=1}^{m} \left(d_i \frac{\partial^2 d_i}{\partial a_k \partial a_j} + \frac{\partial d_i}{\partial a_k}\frac{\partial d_i}{\partial a_j} \right), \quad \frac{\partial g_k}{\partial x_i} = d_i \frac{\partial^2 d_i}{\partial a_k \partial x_i} + \frac{\partial d_i}{\partial x_i}\frac{\partial d_i}{\partial a_k},$$

with similar expressions for $\partial g_k/\partial y_i$ and $\partial g_k/\partial z_i$.

Uncertainty associated with the input ξ due to rounding or simulated measurement uncertainty can be propagated through to the solution α. If V_ξ is the variance matrix associated with the input ξ, then the variance matrix associate with the solution α is given by

$$V_\alpha = S V_\psi S^T.$$

If software under test supplies an estimate $\hat\alpha$ of α, then the validity of $\hat\alpha$ can be assessed by determining the squared distance of $\hat\alpha$ from the reference solution α, relative to the variance V_α, by evaluating

$$\chi^2(\hat\alpha) = (\hat\alpha - \alpha)^T V_\alpha^{-1} (\hat\alpha - \alpha).$$

Evaluating the numerical uncertainty associated with reference datasets for the ChODR problem is also more complicated, partly because the solution parameters are not necessarily a smooth function of the outputs.

5 Development of ICT infrastructure

The difference between standard approaches to validating CMM sofware and the TraCIM approach is the use of an ICT infrastructure to provide an online validation at the point of use [8]. The TraCIM system allows a customer of the software validation service, e.g., a software developer, to interface with the TraCIM server as follows:

- The TraCIM server delivers reference input data sets to the customer. (All data exchange is implemented securely.)
- The customer applies the software under test to the data sets to obtain corresponding test output data sets. These data sets are delivered to the TraCIM sever so that they can be compared with corresponding reference output data sets.

- The user is provided with information, e.g., a certificate, on the performance of the software under test.

All of these processes are done automatically, once software modules are in place to communicate with the TraCIM system. Industrial partners in the project have implemented (or are implementing) the required interfaces to able to receive online certification.

6 Conclusions

The EMRP JRP 'Traceability for computationally-intensive metrology' begins to address the need to provide a validation service for software used in computational chains within metrology. This paper provides a summary of the main steps that are being undertaken to achieve goal of developing and establishing an ICT infrastructure for software validation. Computational aims from coordinate metrology are a particular focus for the JRP and the examples of least squares and minimum zone geometric element fitting are discussed.

A commercial online software validation service is currently available for least squares geometric element fitting. It is intended that the service for minimum zone geometric element fitting will be available in early 2015, with services for other application areas to follow.

Acknowledgements

This work has been undertaken as part of the EMRP Joint Research Project NEW06 'Traceability for computationally-intensive metrology'. The EMRP is jointly funded by the EMRP participating countries within EURAMET and the European Union.

References

[1] B. P. Butler, M. G. Cox, A. B. Forbes, S. A. Hannaby, and P. M. Harris. A methodology for testing the numerical correctness of approximation and optimisation software. In R. Boisvert, editor, *The Quality ofNumerical Software: Assessment and Enhancement*, pages 138–151. Chapman and Hall, 1997.

[2] M. G. Cox, M. P. Dainton, A. B. Forbes, P. M. Harris, H. Schwenke, B. R. L. Siebert, and W. Woeger. Use of Monte Carlo simulation for uncertainty evaluation in metrology. In P. Ciarlini, M. G. Cox, E. Filipe, F. Pavese, and D. Richter, editors, *Advanced Mathematical and Computational Tools in Metrology V*, pages 94–106, Singapore, 2001. World Scientific.

[3] M. G. Cox and A. B. Forbes. Strategies for testing form assessment software. Technical Report DITC 211/92, National Physical Laboratory, Teddington, December 1992.

[4] R. Drieschner, B. Bittner, R. Elligsen, and F. W¨aldele. Testing coordinate measuring machine algorithms, Phase II. Technical Report EUR 13417 EN, Commission of the European Communities (BCR Information), Luxembourg, 1991.

[5] EURAMET. *European Metrology Research Programme*. Website: www.emrponline.eu.

[6] A. B. Forbes, P. M. Harris, and I. M. Smith. Correctness of free form surface fitting software. In D. G. Ford, editor, *Laser Metrology and Machine Performance VI*, pages 263–272, Southampton, 2003. WIT Press.

[7] A. B. Forbes and H. D. Minh. Generation of numerical artefacts for geometric form and tolerance assessment. *Int. J. Metrol. Qual. Eng.*, pages 145–150, 2012.

[8] F. H¨artig, M. Franke, and K. Wendt. Validation of CMM evaluation software using TraCIM. In F. Pavese, A. Chunovkina, M. B¨ar, N. Fischer, and A. B. Forbes, editors, *Advanced Mathematical and Computational Tools for Metrology X*, Singapore, 2014. World Scientific. Submitted.

[9] ISO. *ISO 10360–6:2001 Geometrical Product Specifications (GPS) – Acceptance and reverification tests for coordinate measuring machines (CMM) – Part 6: Estimation of errors in computing Gaussian associated features*. International Organization for Standardization, Geneva, 2001.

[10] ISO. *ISO 1101: Geometric Product Specifications (GPS) – Geometric Tolerancing – Tolerances ofform, orientation, location and run out*. International Organization for Standards, Geneva, 2005.

[11] ISO. *ISO 25178 Geometric product specifications (GPS) – Surface texture: Areal, Part 2: Terms, definitions and surface texture parameters*. International Organization for Standardization, Geneva, 2012.

[12] ISO. *ISO 25178 Geometric product specifications (GPS) – Surface texture: Areal, Part 3: Specification operators*. International Organization for Standardization, Geneva, 2012.

[13] G. J. P. Kok and I. M. Smith. Approaches for assigning numerical uncertainty to reference data pairs for software validation. In F. Pavese, A. Chunovk-ina, M. B¨ar, N. Fischer, and A. B. Forbes, editors, *Advanced Mathematical and Computational Tools for Metrology X*, Singapore, 2014. World Scientific. Submitted.

[14] H. D. Minh, I. M. Smith, and A. B. Forbes. Determination of the numerical uncertainty for numerical artefacts for validating coordinate metrology software. In F. Pavese, A. Chunovkina, M. B¨ar, N. Fischer, and A. B. Forbes, editors, *Advanced Mathematical and Computational Tools for Metrology X*, Singapore, 2014. World Scientific. Submitted.

[15] P. J. Scott and A. B. Forbes. Mathematics for precision engineering. *Phil. Trans. Roy. Soc. A*, 370:4066–4088, 2012. doi:10.1098/rsta.2011.0379.

[16] TracCIM. *Traceability in Computationally-Intensive Metrology (TraCIM)*. Website: www.tracim.eu/emrp/1390.html.

[17] TracCIM. *TraCIM Computational Aims Data Base*. Website: www.tracim-cadb.npl.co.uk.