

# ONLINE VALIDATION OF METROLOGICAL SOFTWARE USING THE TraCIM SYSTEM

*F. Härtig*<sup>1</sup>, *B. Müller*<sup>2</sup>, *K. Wendt*<sup>3</sup>, *M. Franke*<sup>4</sup>, *A. Forbes*<sup>5</sup>, *I. Smith*<sup>6</sup>

<sup>1</sup>Physikalisch-Technische Bundesanstalt, Braunschweig, Germany, frank.haertig@ptb.de

<sup>2</sup>Ostfalia University of Applied Science, Wolfenbüttel, Germany, bernd.mueller@ostfalia.de

<sup>3</sup>Physikalisch-Technische Bundesanstalt, Braunschweig, Germany, klaus.wendt@ptb.de

<sup>4</sup>Physikalisch-Technische Bundesanstalt, Braunschweig, Germany, matthias.franke@ptb.de

<sup>5</sup>National Physical Laboratory, Teddington, United Kingdom, alistair.forbes@npl.co.uk

<sup>6</sup>National Physical Laboratory, Teddington, United Kingdom, ian.smith@npl.co.uk

**Abstract** – Numerical algorithms used in metrology can be validated based on test data and corresponding reference results in order to improve the reliability of measurement results. This paper describes an innovative online based validation service. The concept is named Traceability for Computationally-Intensive Metrology (TraCIM). It allows end users and developers to get their software packages certified at the point of use. The service will be operated by European national metrology institutes (NMIs).

**Keywords:** validation of numerical algorithms, soft-gauges, computational traceability

## 1. INTRODUCTION

The main objective of the TraCIM project (*Traceability of Computationally-Intensive Metrology*) is to develop new technologies that allow users to validate their software directly at the point of use (e.g. on measuring systems) and at any time. To this end an entire infrastructure is provided [1].

The TraCIM concept is based on three essential pillars: the technical implementation, legal issues, and commercial requirements. The technical implementation provides a client-server concept. It allows a direct link between the NMIs and the end users. It is a fundamental principle that the TraCIM service is provided and hosted only by a NMI or an authorized organization such as a designated institute (DI). These institutions assume del credere liability and final guarantee for the correctness of results. Similar to the well known calibration chain which is related to physical standards the NMIs transfer the numerical accuracy of evaluation algorithms from the highest metrological authority to the individual application.

Today, a significant – and partly dominant – contribution to measurement uncertainty is due to the processing of the measurement data. Its causes are varied and the analysis algorithms used cannot practically be traced back to metrology institutes at all. The errors often come from the incorrect or insufficient implementation of formal mathematical requirements. More often than not, seemingly trivial problems are the cause.

In order to address computational errors, some metrology institutes already supply possibilities of checking analysis algorithms. These checks are often available for free. Industry, willing to demonstrate that its products can be trusted, is increasingly calling for a certificate issued by a trustworthy organization. Driven by this incentive, activities aimed at an online validation under the supervision of European metrology institutes have begun. To this end, not only technical, but also commercial and monetary preconditions had to be met in order to set up a complete and sustainable system.

## 2. THE TRACIM SERVICE

In association with and under the supervision of European metrology institutes, TraCIM aims to validate analysis algorithms in the field of metrology. In the following, they will be referred to as "algorithm tests" or simply "tests".

The kind of computations addressed are computations used to analyse and evaluate measurands of the International System of Units (SI) and their derived units. The medium of choice for communication between the service provider and the user is the Internet (Fig. 1).

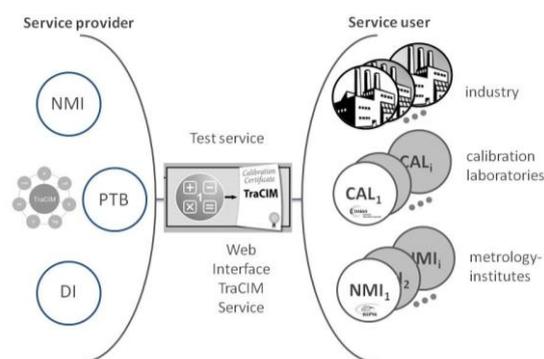


Fig. 1: TraCIM service.

TraCIM provides a technical infrastructure to perform the software tests. Each service provider is, however, solely responsible – and therefore held liable – for the extent of the algorithm tests provided, the business workflow, the maintenance of the datasets, and customer support in addition to

running the tests. For these reasons, each metrology institute should operate its own server within the TraCIM network.

The service is primarily focused on manufacturers of analysis software or measuring instruments. The service allows them to have their mathematical algorithms validated by an independent metrology institute. Above all, this strengthens the confidence in the products they offer on the market. In principle, it is possible to activate this service also for their customers, i.e. the end users of the software, in order to have updates, revisions or new software releases validated directly at the point of use at any time and immediately.

### 2.1 Specification of computational aims

It is only possible to verify and validate software when it is known what problem the software is intended to solve or task is intended to execute. Such a statement is essential both to act as the functional requirements for software developers, and to provide as basis for verification and validation of software implementations.

The specification of the computational aims should be unambiguous, complete, free from contradictions, and independent of the environment, such as hardware and software configurations, in which it is to be implemented. As part of the TraCIM project a generic procedure has been developed which provides a clear description of how a computational aim should be specified. A searchable database was set up that acts as a repository for specifications of computational aims. In this database information on mathematical models, descriptions of input and output parameters, properties and dependencies are to be found [2]. Among those for other applications, documents describing least squares and minimum-zone algorithms for fitting geometric elements to point clouds as applied in the area of coordinate metrology can be found in this database.

### 2.2 Test data

A general methodology for evaluating the accuracy of results produced by metrological software is the design and use of test data sets and corresponding reference results to

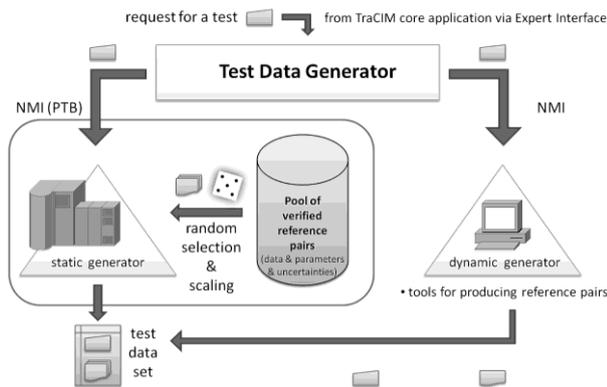


Fig. 2: Test data generator.

undertake black-box testing [3, 4, 5]. Data generators are commonly used in the process of testing software implementations. Generally, one of the following two approaches is implemented:

- Forward data generation: Input test data sets are generated and are used to produce corresponding

- reference output results by means of reference software.
- Reverse data generation: Reference output results are given and corresponding reference input data are generated by software specially developed for this purpose.

TraCIM allows the implementation of both approaches for dynamic generation of test data sets on demand. However, it is also possible to generate test data sets and the corresponding reference results in advance and store these reference pairs in a database (Fig. 2). On request reference pairs are then randomly selected from the data base. The benefit is that – prior to storage - all reference pairs generated can be cross checked with a second independent reference software, which increases the security and reliability of the data significantly.

As a prerequisite for broad acceptance of software testing, particularly for customers in industry, the reference pairs should reflect real world applications to be of practical and applied benefit. The test data must be well-designed and the results have to be unambiguous. In the domain of coordinate metrology it means that, for instance, point clouds are designed such that the corresponding geometric elements are well-defined and unique. Furthermore the data sets should be flexible, i.e. applicable for different length units such as nm, mm, and m depending on the needs of the users. The data sets should also be adaptable to measurements at the nano-, meso- or large scale.

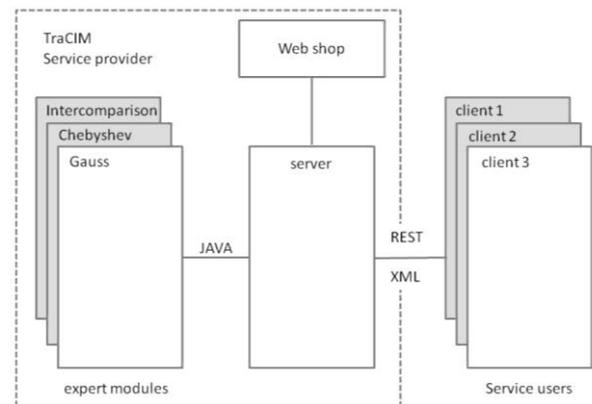


Fig. 3: TraCIM modules.

### 2.3 Architecture of the TraCIM system

The IT architecture of the TraCIM system consists of four central modules (Fig. 3).

The server is a JavaEE application running on a JBoss server which processes requests from client software (sending test data, receiving test results, and sending test reports), stores customer and order data, archives all data and documents electronically sent or received, and communicates with a Web shop and expert modules. Thereby, an expert module embodies a specific test task, i.e. different computational aims even from different metrological domains. The communication between server and client application runs via a REST interface that is a special type of HTTPS connection. Hereby, the data are embedded into an XML structure. Some elements are mandatory, such as information required for processing data received from service users and to assign test results correctly to test data.

On the other hand, free formats (such as binary formats, existing test data structures or newly defined formats) can be defined within the XML structure, in particular sub-structures for test data and test results. The expert module is solely responsible for defining these special formats.

The expert modules are applications which provide specific test data on demand, compare reference data to data calculated by the software under test and issue test reports. Each expert module essentially operates autonomously. Since individual tests may vary significantly from one test application to another, only few input/output parameters have been specified for the data exchange between the expert module and the core application of the server.

The Web shop provides methods for user registration and for ordering tests. Additionally a payment system may also be integrated, but to date has not been implemented.

The client software is an interface on the computer of the service user, which is preferably integrated in the user's application. It is responsible for connecting the software under test with the TraCIM server. The client interface allows the request and reception of test data and – after processing them – to send the calculated results directly to the TraCIM server. Client-server communication runs via a REST interface. Hereby, the data are embedded into an XML structure (see above).

#### 2.4. Registration

In order to get access to the TraCIM services a user must register and provide valid contact data like company name, address and e-mail address. It is essential to have these data for providing a service and carrying out business activities associated with the issue of certificates and with providing appropriate support. After registration, the account data, including a unique customer ID, user name and password generated automatically by the TraCIM system, are submitted to the service user by e-mail for further communication with the server and for ordering individual tests. Registration is free of charge for all service users and allows access to public test data.

#### 2.5. Implementation of service clients

Technical requirement for automatically performing a test is that a service user provides a client application for receiving test data and submitting calculated test results [6, 7]. For communication with the TraCIM server a client application must use an HTTPS (Hypertext Transfer Protocol Secure = encrypted HTTPS) connection that allows the sending and receiving of content in the form of character strings, in particular messages in XML format. Each HTTPS connection is generated from a specific URL (Uniform Resource Locator), i.e. for requests of test data or submitting of test results. Software packages for creation and configuration of HTTPS connections are available for different programming languages such as Java, C/C++ or C#.

Software vendors are advised to integrate the service client directly into the application software under test. They are thus able to perform tests fully automatically.

All service users are advised to check the fault-free and stable client-server communication before performing test runs subject to charge (see 2.6). For checking the correct communication between client application and TraCIM system public data are available free-of-charge.

#### 2.6. Ordering tests

PTB offers the TraCIM service for validation of metrological algorithms and operates its own server. A web interface provides a shop where customers can order different tests at any time. At present PTB offers software tests for the validation of:

- Least-squares fitting algorithms to data points,
- Chebyshev minimization algorithms (fitting of minimum-zone elements to data points),
- Algorithms for key comparison evaluation.

The costs vary depending on the type of test. In addition to individual tests, packages of 10 and 50 tests will be available soon. The service is subject to charge to cover the costs for issuing official certificate documents, management, maintenance and further expansion of services.

In order to check the correct implementation of the client-server communication regarding data content, data formatting and access rights a full set of public test data are available for each of the three software tests offered.

After receipt of the order and payment the service user will receive an e-mail with a unique order key. On request he can also obtain a special order key which can be used to access public test data. The order keys allow the service users to identify all tests ordered in the customer area of the Web shop.

#### 2.7. Requesting test data sets

After successfully completing the order and payment process, the test client can submit the order key to the TraCIM server by sending the following procedure message 'https://tracim.ptb.de/tracim/api/order/<\_ORDER\_KEY>/test'. On acceptance of the key, the server will send test data and administrative information to the service client together with an automatically generated, individual process key, which identifies the kind of test performed and also the set of test data used for testing. Fig. 4 illustrates the structure of the TraCIM message submitted to the client. The header is mandatory.

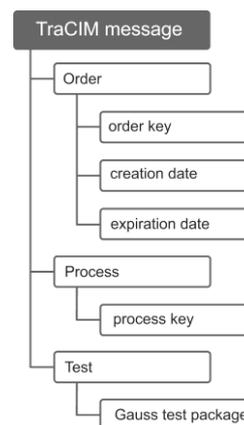


Fig. 4: Mandatory TraCIM header.

In order to test software for computing Gaussian least-squares fitting algorithms, different sets of data are sent to the client consisting of point clouds representing certain geometric elements such as lines, planes, circles, cylinder, cones or spheres. Fig. 5 illustrates the structure of the test data package. For the validation of Chebyshev minimum-

zone algorithms the sub-structure of the test data package sent to the service user is entirely identical.

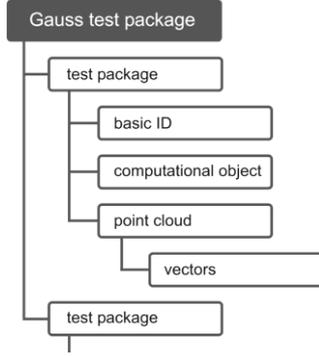


Fig. 5: Example of a test data package.

Service users who have ordered a software test for computing Gaussian least-squares fitting algorithms get a fixed set of 44 point clouds (test packages) representing measuring points on 3D lines, planes, 3D circles, cylinders, cone, and spheres. They represent full and partial geometric elements, i.e. the measuring points cover the entire surface of the feature elements or only a part of its surface. The number of points per data set varies between 8 and 50. The maximum deviations in shape vary for each feature between 20 and  $2 \times 10^{-6}$  relative to the overall volume completely enclosing all point clouds. The individual perturbations have been randomly generated. The size of the point clouds (maximum distance between any pair of points) is within 0.02 and 0.15 related to overall volume. At present all point coordinates supplied are within the value range of [-500 mm, +500 mm]. However, it is intended to adapt existing data sets, i.e. to scale them up or down, in order to make them more suitable for verifications especially where customers require test data sets which are more related to their fields of application.

Service users who have ordered a software test for verification of minimum-zone fitting algorithms receive a fixed set of 50 point clouds representing full and partial geometric elements such as 2D-lines, 2D-circles, planes, spheres and cylinders. The number of points per data set varies between 10 and 631.

## 2.8. Sending test data results

The service user evaluates the test data sets with his software and submits the calculated results along with the process key and an identification of the software under test in a XML structure to the TraCIM server by sending the following XML message through HTTPS with the URL 'https://tracim.ptb.de/tracim/api/test/<PROCESS\_KEY>/test'.

Apart from the calculated results, service users also have to submit information relating to name, release and revision of the software under test. They also have to submit information relating to the (expected) computational accuracy of their software. For each test value (see below) the maximum permissible error (*MPE*) has to be provided (Fig. 6).

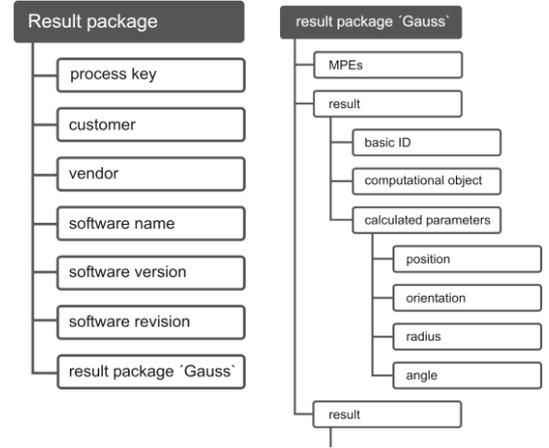


Fig. 6: Example of a result message.

## 2.9. Test evaluation

To verify the correctness of the calculated test parameters obtained by the software under test, they are compared with the corresponding reference values. Therefore, the difference  $d$  between the reference and calculated value is calculated for each parameter. The calculated parameters are considered to be sufficiently exact, if for any difference  $d$  the following applies in case of the Gaussian or Chebyshev test

$$|d| \leq MPE_d + U_d, \quad (1)$$

where  $U_d$  denotes the numerical test uncertainty, which quantifies how accurate the reference values are. This value has to be determined by the provider of the test (testing body).  $MPE_d$  refers to the maximum permissible error associated with each test criteria. The value is to be provided by the service user. In the case that all values  $d$  are smaller than the corresponding limits, a test certificate is issued stating that the software successfully passed the test.

However, in some cases the differences  $d$  are not just the difference of certain parameters. Since different parameter sets may describe the same geometrical element appropriate test values have been defined. These test values are denoted as 'orientation', 'location', 'radius', 'angle', and 'form deviation'. Service users have to specify *MPE* values for each of these test values.

In order to compare the calculated direction cosine  $n_{cal}$  to the reference orientation  $n_{ref}$ , the test value 'orientation' is calculated as the smaller angle between both lines:

$$d(n_{ref}, n_{cal}) = d_\delta = \arcsin \frac{\|n_{ref} \times n_{cal}\|}{\|n_{cal}\|}, \quad (2)$$

assuming that the vector  $n_{ref}$  is normalized, i.e.  $\|n_{ref}\| = 1$ .

In order to compare the calculated position  $p_{cal}$  of a line, cone or cylinder to the location of the reference element, the orthogonal distance  $d_p$  of point  $p_{ref}$  in or close to the 'centre' of the reference element to the geometric feature obtained by the software under test is calculated:

$$d(p_{ref}, p_{cal}, n_{cal}) = d_p = \frac{\|(p_{cal} - p_{ref}) \times n_{cal}\|}{\|n_{cal}\|}, \quad (3)$$

or when comparing planes

$$d(p_{ref}, p_{cal}, n_{cal}) = d_p = \frac{\langle p_{cal} - p_{ref}, n_{cal} \rangle}{\|n_{cal}\|}, \quad (4)$$

where  $\langle p, n \rangle$  denotes the Euclidean scalar product of two vectors.

When comparing the location of circles or spheres the distance is simply calculated by using the Euclidean norm:

$$d(p_{\text{ref}}, p_{\text{cal}}) = d_p = \|p_{\text{cal}} - p_{\text{ref}}\|, \quad (5)$$

$d_p$  denotes the test value ‘location’.

In order to check whether the radius  $r_{\text{cal}}$  of the calculated cone determined perpendicular to the cone axis at  $p_{\text{cal}}$  corresponds to the reference radius  $r_{\text{ref}}$  at position  $p_{\text{ref}}$ , the test value ‘radius’  $d_r$  is calculated according the following equation:

$$d(r_{\text{ref}}, r_{\text{cal}}, \alpha, p_{\text{ref}}, p_{\text{cal}}, n_{\text{ref}}, n_{\text{cal}}) = d_r, \quad (6a)$$

$$d_r = r_{\text{cal}} - r_{\text{ref}} - \tan\left(\frac{\alpha}{2}\right) \langle p_{\text{ref}} - p_{\text{cal}}, n_{\text{ref}} \rangle, \quad (6b)$$

where  $\alpha$  denotes the full apex reference angle of the cone. Please note that for cylinders, circles, and spheres the test value ‘radius’ is calculated according to equation (8).

For cones it must be checked if the direction cosines  $n_{\text{cal}}$  and  $n_{\text{ref}}$  point in the same direction, i.e. along the cone axis in the direction of decreasing radius:

$$d(n_{\text{ref}}, n_{\text{cal}}) = d_\beta = \arccos \frac{\| \langle n_{\text{ref}}, n_{\text{cal}} \rangle \|}{\|n_{\text{cal}}\|}, \quad (7)$$

This test value must also be smaller than the *MPE* value of orientation.

In order to compare the calculated radius  $r_{\text{cal}}$  of circles, spheres and cylinders with the corresponding reference radius  $r_{\text{ref}}$ , the difference is formed:

$$d(r_{\text{ref}}, r_{\text{cal}}) = d_r = r_{\text{cal}} - r_{\text{ref}}, \quad (8)$$

where  $d_r$  denotes the test value ‘radius’.

In the same way the calculated angle  $\alpha_{\text{cal}}$  of a cone is compared with the corresponding reference angle  $\alpha_{\text{ref}}$ :

$$d(\alpha_{\text{ref}}, \alpha_{\text{cal}}) = d_\alpha = \alpha_{\text{cal}} - \alpha_{\text{ref}}, \quad (9)$$

where  $d_\alpha$  denotes the test value ‘angle’.

In the case of verification of minimum-zone algorithms, the additional test value ‘form deviation’  $d_f$  is defined as follows:

$$d(f_{\text{ref}}, f_{\text{cal}}) = d_f = f_{\text{cal}} - f_{\text{ref}}, \quad (10)$$

where  $f_{\text{cal}}$  denotes the form error calculated by the software under test and  $f_{\text{ref}}$  the reference value.

Only if all test values  $d$  are smaller than the corresponding limit is a test certificate issued. In all other cases a report is submitted indicating the test data set and the test value where exceeding of the corresponding threshold was observed.

Due to technical reasons (storing input and output variables as well as intermediate results with finite precision) the reference parameters are not the mathematically exact solution of the Gaussian or Chebyshev fit for the supplied data sets. Hence, there remains an uncertainty for all test values  $d$  caused by the inaccuracy inherent to reference results. To estimate the uncertainty for the test values a kind of Monte-Carlo simulation was used. The procedure to calculate the uncertainty for the reference results is described in [8]. The uncertainties obtained vary depending on the type of geometric element and depending on whether the measuring points cover the feature completely or only partially. Tab. 1

shows the maximum uncertainties for test values  $d$  within the scope of the Gaussian test. They are very small for location parameters of lines and planes, because the arithmetic mean of the x, y, and z-coordinates of measuring points is the best estimate of a point on line or plane defining its location. They are large for partial feature elements.

geometric feature	orientation	location	radius	apex angle
line	$1 \times 10^{-9}$	$2 \times 10^{-15}$		
plane	$7 \times 10^{-10}$	$1 \times 10^{-15}$		
3D circle	$1 \times 10^{-9}$	$2 \times 10^{-10}$	$1 \times 10^{-10}$	
cylinder	$1 \times 10^{-8}$	$7 \times 10^{-11}$	$6 \times 10^{-11}$	
cone	$1 \times 10^{-8}$	$6 \times 10^{-10}$	$6 \times 10^{-11}$	$1 \times 10^{-8}$
sphere		$3 \times 10^{-11}$	$2 \times 10^{-11}$	

Tab. 1: Max. uncertainty of reference values

The maximum uncertainties listed in Tab. 1 are used to assess the consistency of the calculated with the reference results according equation (1) taking into account the test uncertainty  $U_d$ .

The  $MPE_d$  value in equation (1) is to be provided by the service user. If no values are provided the default values 0.0001 mm and 0.1 arc sec are used in evaluating the test value ‘location’ and ‘radius’, respectively ‘orientation’ and ‘angle’. These are also the smallest values accepted by PTB. Even smaller values would lead to an increased risk of erroneous test results. In order to minimize the risk which could be caused by an erroneous decision, in particular when software has not passed the test criteria, although it delivers valid results, it was decided that no *MPE* values smaller than the default values will be accepted.

### 3. THE TraCIM ASSOCIATION

The purpose of the TraCIM association is the promotion of science, research and European cooperation in the domain of traceability of mathematical evaluation algorithms in metrology. The TraCIM association under German law (TraCIM e.V.) offers a number of possibilities to fit best for the needs of an international cooperation. The organization structure is very flexible and can be adjusted according to the needs of the association and its members by the general assembly. The technical work, especially preparing and implementing technical decisions and tasks, will be carried out by technical committees open to all members as well as to non-members. They are established depending on need.

The correctness of test results, the liability for the service, as well as unambiguous rules and test specifications and acceptance conditions, are essential demands on software validation services. The association has therefore specified quality rules which have to be followed by all institutions offering tests under the brand name of TraCIM. The most important rules are:

- Each institute providing tests is liable for the correctness of the test concerning the reference pairs (test data and corresponding reference results) and evaluation of conformity,
- Input values are defined as error-free,

- Reference results must be provided by the test body including their associated uncertainty,
- Reference pairs (test data and reference results) need to be verified,
- Test runs must be unambiguous, i.e. provide a clear unambiguous result,
- Test procedures must be described in a precise and detailed way,
- Tests shall reflect common practical situations rather than theoretical or academic issues,
- All electronic data and documents sent or received are subject to archiving, in particular all test reports.

From an academic standpoint not all rules may be necessary. However, in the context of the regular provision of services, they appear appropriate and useful in order to avoid unnecessary risks associated with the service provided such as high cost for time-intensive consulting services, customer complaints, and also liability claims for incorrect test results.

#### 4. CONCLUSIONS

The use of validated software algorithms guarantees the correct calculation of measuring values. Since modern metrological applications often use complex algorithms for calculating measuring values, it is very important that all computational links are recognized explicitly and are known to be operating correctly. To fulfil this essential requirement a network of national metrology institutes and designated institutes provides an online service which allows users to test their software at the point of use and at any time by means of verified reference data. The service is still in the process of being established and will constantly be expanded to other metrological domains. Tests for the validation of least-squares fitting algorithms and Chebyshev minimization algorithms used in the field of coordinate metrology are already available at <https://tracim.ptb.de>.

#### ACKNOWLEDGMENTS

The TraCIM system is being set up within the scope of the European Metrology Research Project (EMRP) under the

denomination ‘Traceability for Computationally-Intensive Metrology’. The project is coordinated by the National Physical Laboratory (NPL). Furthermore, it involves five other European metrology institutes, four partners from industry and four universities.

#### REFERENCES

- [1] A. B. Forbes, I. M. Smith, F. Härtig, K. Wendt, *Overview of EMRP Joint Research Project NEW06 ‘Traceability for computational-intensive metrology; in: Advanced Mathematical and Computational Tools in Metrology and Testing X*, World Scientific Publishing. pp. 164-170, 2015.
- [2] TraCIM Computational Aims Database, [www.tracim-cadb.npl.co.uk](http://www.tracim-cadb.npl.co.uk)
- [3] R. Drieschner, B. Bittner, R. Elligen, F. Wäldele, *Testing coordinate measuring machine algorithms, Phase II*, Technical Report EUR 13417 EN, Commission of the European Communities (BCR Information) (Luxembourg), 1991
- [4] G. T. Anthony, B. P. Butler, M. Cox, M. G. Forbes, et al., *Chebyshev reference software for the evaluation of coordinate measuring machine*, Technical Report EUR 15304 EN, Commission of the European Communities (BCR Information) (Luxembourg), 1993
- [5] G. T. Anthony, H. M. Anthony, B. Bittner, B. P. Butler, M. G. Cox, R. Drieschner et al., *Reference software for finding Chebyshev best-fit geometric elements*, Precision Engineering, Vol. 19 (1), p. 28-36, 1996.
- [6] TraCIM manual ‘Validation of Gaussian Algorithm Implementation’, [https://tracim.ptb.de/tracim/customer/service\\_gauss.jsf](https://tracim.ptb.de/tracim/customer/service_gauss.jsf), 2015
- [7] TraCIM manual ‘User manual for Chebyshev algorithm testing’, [https://tracim.ptb.de/tracim/customer/service\\_chebyshev.jsf](https://tracim.ptb.de/tracim/customer/service_chebyshev.jsf), 2015
- [8] F. Keller, K. Wendt and F. Härtig, *Estimation of test uncertainty for TraCIM reference Pairs; in: Advanced Mathematical and Computational Tools in Metrology and Testing X*, World Scientific Publishing. pp. 187-399, 2015.