VALIDATION OF METROLOGICAL SOFTWARE

Halina Nieciąg [1], *Zbigniew Chuchro*[2]

[1] The Institute of Advanced Manufacturing Technology, Cracow, Poland, hnieciag@ios.krakow.pl
[2] The Institute of Advanced Manufacturing Technology, Cracow, Poland, zbigniew.chuchro@ios.krakow.pl

Abstract: The problems involving the validation of metrological software are explored in this paper. This important module of measuring systems fulfills the mathematical processing of measured data. The concept of validation and their realization is considered based on building a conceptual model of software components under examination. The presented method helps software validation through estimating intervals specified within which the estimates of quantities indirectly measured should lie.

Keywords: modelling, validation, software.

1. INTRODUCTION

Software engineering considers software to be a separate product. Quality assurance regarding the software means that software shows correctness and consistency with respect to provided references or standards. The basis of objective conclusions can be measurements of quality metrics in real environment of software implementation and activity.

The process of software evaluation is a validation process [1]. In this article we present the validation involved metrological programs, since they are most frequently used components of software in coordinate measurements.

Establishing whether the validated software is capable to give results within the stated intervals depends strongly upon the measurement task, becoming therefore a starting point for studies leading to an estimation of acceptance criteria in the form of permitted ranges. For software addressed for coordinate metrology applications bias and precision are basic measures indicating result's accuracy. Intuitively rather not in formal way speaking a precision expresses a potential imperfection in activity of measurement model due to our lack of knowledge while a bias is a recognizable imperfection.

In order to deal with validation problem the conception had been formulated based on building a conceptual model of software components under examination and their analysis. The analysis incorporated the parameters estimated in fundamental metrological programs. They have been developed to perform parametric identification of substitute geometric features [2].

2. VALIDATION PLAN

Validation strategy establishing is the first, most important operation in validation process that consists in a choice of appropriate suitable testing model in general. Unfortunately, the universal rules for correct validation of complex measuring systems do not exist.

Usually two approaches are taken. First an „axiomatic" approach relays on modeling technique and leads to the validation of theoretical model underlying the validated software. Second, an „empirical" is applicable on closed and unavailable for experimenter software structure, where the software being tested may not be apparent. The robust and effective apparatus for simulations experiments, based on black box concept, or formalized methodology of experiments with reference artifacts decide these empirical methods are very popular. This way of testing seems to be indispensable in such cases, when building of sufficient adequate model of object is difficult or just impossible, e.g. in testing of software packages, which were developed for special purposes. The modeling techniques give more depth and exhaustive knowledge about inspected object, and it is sometimes the preliminary step of validation process (pre-validation).

The validation is realized with regard to the integrated testing rules, i.e. component testing. The modular approach is applied, substituting the validated original by their mathematical representations.

3. DECOMPOSITION RULE

The software quality evaluation of real measuring systems by a complete examination is practically unfeasible because of its complexity. System decoupling enables replacing an evaluation of entire software product by the particular layers design analyzing. Highest quality demands are formulated with respect to the metrological software integrated with such measuring systems like CMM-s. The subjects under consideration are programs being the consistent parts of this software module. The programs of concern are programs for parametric identification of such features as a straight line, a circle, a sphere, a plane, a cylinder and a cone.

4. THE COMPONENT MODEL

The ability of realization the wide variety of measuring tasks by them allows the module decomposition and producing the model in mathematical term for each relevant task separately. Each identified by program feature can be defined as a function, taking general implicit form (1).

$$F(\mathbf{u},\mathbf{p}) = 0 \,, \tag{1}$$

where $\mathbf{u} = [x, y, z]^{\mathrm{T}}$, $\mathbf{u} \subset \Re^k$, is a vector modelling the input, i. e. quantities measured directly, a vector of intrinsic parameters $\mathbf{p} = [p_1, p_2, ..., p_m]^{\mathrm{T}}$, $\mathbf{p} \subset \Re^m$, denotes program outcomes, i.e. quantities measured indirectly[1]. The choice of parameters representing geometric feature depends on the type of it. The function of the form (1) represents quite often a nonlinear model with respect to both input quantities and parameters.

5. PROGRAMS

The estimation of unknown parameters is a task, realized by each program being a consistent part of analyzed software module. Each program minimizes an objective function Q (2), defined as a norm of orthogonal residuals between given feature model of the form (1) and observations of n-independent variables, known after measurements.

$$\min_p(Q) = \min_p(\|r(p)\|_2^2) = \min r(p)^T r(p) \,, \tag{2}$$

where $r(p,u) = f(p,u) - d$ denotes a vector of residuals, $f(p,u)$ is a redefined function (1), d is a vector of observations.

Analyzed programs were based upon the Gauss-Newton method. The basis of this method is local approximation of Q function using Taylor series expansion with its first and second order terms (2), within the small neighbourhood of fixed point $\widetilde{p}$, close to a stationary point (corresponding to the minimum of Q).

$$\nabla Q \cong \nabla^1 Q(p) + \nabla^2 Q(p)(p - \widetilde{p}) = 0 \,, \tag{3}$$

where $\nabla^2 Q = \nabla r(p)^T \nabla r(p) + o(p) \cong \mathbf{J}(p)\mathbf{J}(p)$, and $\mathbf{J}$ stands an Jacobian matrix of objective function Q.
Referring to the fact, that the gradient of function should equal to zero at stationary point, an estimation of optimal parameters p relies on substituting j^{th} iteration of non-linear model by its linear approximation and successively refining this approximation by a correction in each iteration step near approximated parameter p. Newton's correction in subsequent j-th iterations follows from solving the equations given in (4).

$$\mathbf{J}(p)^j \delta p - (-\mathbf{r}(p)^j) = 0 \,. \tag{4}$$

---

[1] For the sake of clearness only the matrices will be next representing by boldface letters

Searching of the Newton's correction can be interpreted as linear approximation of the residual function $r(p)$ with its linear representation $\mathbf{J}(p)^j \delta p + \mathbf{r}(p)^j$.

The equations system (4), being the base for further parameters assessments in program, is representing the source functional relations between input and output model variables.

The results of this local linear approximation have simple geometric interpretation. They form in n-data space a solution vector lying in plane tangent to m-dimensional surface representing function at fixed value $\widetilde{p}$ of m-components.

6. ERRORS – VARIABLES MODEL

In our analysis any i^{th} measured data point is associated with three Cartesian coordinates in the form of vector $u_i = (x_i, y_i, z_i)^{\mathrm{T}}$. The direct measured quantities in expression (1) one considers to be realizations of separate random variables of random vector $U = (X, Y, Z)^{\mathrm{T}}$. One assumes that each data point is associated with $\Delta u_i = (\Delta x_i, \Delta y_i, \Delta z_i)^T$, i^{th} - realization of random error vector. A vector estimate $\widehat{p}^j : \widehat{p}|u = \{x, y, z\}$ obtained at j^{th} iteration is assessed from given n-data, mapping a real object, including the errors. Therefore the vector values are considered to be observations of random variables P, inferred from input quantities values and their errors.

An estimator of bias and precision concept is proposed of the form (5).

$$\widehat{p}^j = p^* + (\widehat{p}^j - p^*) = p^* + \varepsilon^j + \Delta_s \,. \tag{5}$$

The random error ε^j unknown from sign and value after measurement is resulting from random effects associating with actual experiment realization. The systematic error Δ_s is arising rather due to the systematic effects e.g. an offset in input data being in relationship with an experiment strategy. It can be intrinsic to measurement strategy by particular choice of probed regions or density of measurement pattern. We introduce into analysis the quantity potential true value p^*. It is representing the entire measured surface, when the number of data points and replicate trials in experiment hypothetically increase up to infinity (6) or one can deduce with a suitably small uncertainty which might be zero that p^* is a true value of estimate for sufficiently data representation of measured profile (6).

$$Bias[\widehat{p}] = \mathsf{E}[\widehat{p}^j - p^*] = \widehat{p} - p^* = \Delta_s \,. \tag{6}$$

Experiment replications for the same model, under repeatability conditions, indicate that the estimates resulting from random character of input data thereby are changing randomly in neighbourhood of expected value $\widehat{p}$:

$$\mathsf{E}[\widehat{p}^j] = \widehat{p} \,.$$

7. BIAS ESTIMATION

A bias in estimates cannot be determined exactly because it depends on true values of p quantities (7).

$$Bias[P] = \mathsf{E}[\widehat{p}^j - p] = \widehat{p} - p . \qquad (7)$$

We cannot say with absolutely certainty that we know exactly the correct value of parameter's estimate even if we know the limits within which form deviations of profile have to be kept or possible random variations. The computed values vary for any real object and in any real experiment. The aim of analysis is then to estimate a bias, basing on the available knowledge.

Many contributors to uncertainty of parameter estimate one can observe or suspect. Complexity of coordinate measurement as well as susceptibility of measurement result to many factors makes rational appointing and quantifying only these having significant impact into statement of uncertainty. Helpful in analysis of importance of factors essentially influencing the measurements results is a "fish diagram", known as Ishikawa diagram.

The following consideration was applied. The universal measurement task realized for Gaussian features by metrological software consists of two stages: assessing the estimates of measured quantities, following collecting a set of input data. One reduces a number of potential sources of inaccuracy into two main factors affecting the results: associated with each program quality (simplified model, data processing techniques) and relating to coming data, the factors that are not avoidable and they cannot be prevent from happening. They influence both bias and precision in estimates.

7.1. Bias decomposition

Thus, we perceive two not to excluding sources of bias in estimates, damping or eliminating others. These sources are a bias present in experimental data and therefore affecting the parameters estimates and a bias due to the nonlinearity of the assumed model. This latter does not depend upon experimental sample quality directly but it depends mainly on particular parameterization used in mathematical model. Both of them, parameterization and model, are apparently arbitrary for a given object in coordinate technique.

Two components of overall bias in parameters are provided by (8).

$$Bias[P] = \mathsf{E}[\widehat{p}^{(j)} - p^* + p^* - p] = \left| Bias[\widehat{p}] \right| + \left| Bias[p^*] \right| . \quad (8)$$

This distinction between these two main bias components is in conformance with experience and observations of phenomena accompanying the coordinate measurements.

7.2. Bias due nonlinearity

We analyze a bias due to nonlinearity since estimators of parameters for nonlinear model, obtained using linear regression methods obviously have not the optimal properties:

$$Bias[p^*] = \mathsf{E}[p^* - p] . \qquad (9)$$

We utilize to this end a quantity called parameter effects curvature (PE) and techniques developed by Bates and Watts [4] derived from Box theory [3]. This measure as well as intrinsic curvature (IN) is used in fact to characterize a degree of nonlinearity in measurement model to be analyzed, i.e. closeness of linear approximation to the nonlinear model.

The nonlinearity can be assessed generally by higher order terms neglected in linear function approximation made by means of Taylor series expansion (3). It can be quantitatively expressed by matrix, say $\mathbf{H}$, based on Hessian structure. It consists of mxm partial second derivatives in fact but one considers only $m(m+1)/2$ distinct elements because of Hessian symmetry. The elements of this matrix are calculated at each point of n-dimensional space of sample giving n-vectors. They can be projected onto a tangent plane replacing the surface spanning the function $f(p)$ values (see Fig.1.a) on which the n-vectors of Jacobian matrix are already located and onto an orthogonal plane to it also.
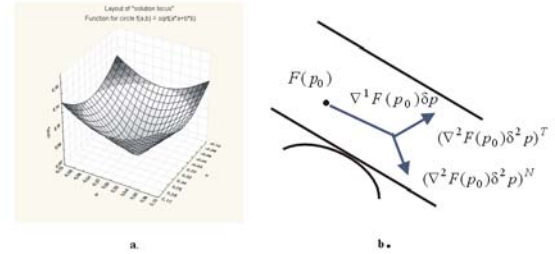

Fig. 1. a. m-dimensional surface of expected values of model function b. Decomposition schema

Tangential components can be related to nonlinearity in estimated parameters, which are of the form of curved parameters lines (see Fig. 1. b).

According to figure one can decompose a vector representing the function in tangent plane. First the matrix compounded of matrices $\mathbf{J}^2$ and $\mathbf{H}$ is evaluated at given point $\widetilde{p}$. In order to obtain the tangential and orthogonal components we apply the QR factorization technique. This method factors a matrix having $m(m+3)/2$ size evaluated in n-sample space values as a product of orthogonal matrix $\mathbf{Q}$ by upper triangular matrix $\widetilde{\mathbf{R}}$ [4], such that:

$$[\mathbf{J}|\mathbf{H}] = \mathbf{Q} \begin{bmatrix} \widetilde{\mathbf{R}} \\ 0 \end{bmatrix} . \qquad (9)$$

The first nxm columns of $\mathbf{Q}$ matrix are the vectors related to tangent plane and $nx(n-m)$ columns are to orthogonal one. In this way the projected matrix can be attainable. We form thus matrix $\mathbf{A}$ multiplying the matrix of second derivatives by matrix $\mathbf{Q}^T$.

Following [3-4], an approximate bias in the maximum likelihood parameters one can be computed with useful formula (10).

---

[2] For the convenience we designate Jacobian matrix $\mathbf{J}$

$$Bias[p^*] = \frac{-\sigma^2}{2} \mathbf{L}(\sum_{l=1}^{m} a_{ll}^T), \qquad (10)$$

where: matrix $\mathbf{L} = \widetilde{\mathbf{R}}^{-1}$, and $\sigma = \|d - f(x, \widehat{p})\|_2$ represents estimate of standard deviation under usual assumptions. It is worth to note that in practice we do neither use relative measures nor a scaling factor.

7.3. Bias due data

An evaluation of this kind of bias is based on perturbation analysis. We do not distinguish between several sources of bias in input variables like the software being supplied by data does not recognize them as well. However the knowledge concerning the possible errors assigned to given area of identified object should be in our disposal.

Assuming the sufficient number of replications of the experiments, the expected values of functions on both sides of equations (4) are biased by factors resulting from biased data:

$$\mathsf{E}[\mathbf{J}(\widehat{p}, u)] \cong \mathsf{E}[\mathbf{J}(\widehat{p}, \overline{u})] = \mathbf{J}(p^*, \overline{u}) + \Delta\mathbf{J}, \text{ and}$$

$$\mathsf{E}[r(\widehat{p}, u)] \cong \mathsf{E}[r(\widehat{p}, \overline{u})] = r(p^*, \overline{u}) + \Delta r,$$

where $\overline{u}$ is an arithmetic mean of u.

Taking into account the imposed structure of matrices $\Delta\mathbf{J}$, Δr, as well as knowing the bounds of systematic perturbations of input data, the unknown bias can be found as a solution to the problem of minimization of the term (11).

$$Bias[\overline{p}] \leq \underset{\delta p}{argmin} \|\Delta\mathbf{J}\delta p - \Delta r\|_2^2. \qquad (11)$$

7.4. Bias aggregation

Triangle inequality can be used to provide the bounds to overall bias (7). Following (8) we get:

$$\|Bias[p]\|_2 \leq \|Bias[\overline{p}]\|_2 + \|Bias[p^*]\|_2.$$

Therefore the root sum squared combination of corresponding components can be used to estimate a bias limit.

8. PRECISION ESTIMATION

In precision estimation the same model of input and output variables like in bias analysis is applied as well. Following similar consideration, the system of equations (4) is analyzed again. For enough much sample size one assumes that expected values can be replaced with their mean values $\widehat{p} = \overline{p}$ and $\widehat{u} = \overline{u}$, i.e. using an average of values obtained from repeated trials of experiment. This model is investigated only in the nearest neighborhood of $\overline{u} : O_u(\overline{u})$, where we expect the true values can lie.

It implicates that $\overline{p} + \Delta p \in O_p(p)|u = \overline{u} + \Delta u$, where components of vector $\Delta p = (\Delta p_1, ..., \Delta p_m)^T$ are small deviations of p from point $\overline{p}$.

From an analysis (4) at given point it turns out after many derivations [5] that parameter estimates in least squares sense (2) with respect to the model (1) yields the relation (12).

$$\Delta p \cong \mathbf{M}\Delta u, \qquad (12)$$

where $\mathbf{H}$ denotes derived conversion matrix.

The equation (12) expresses, via $\mathbf{M}$ matrix, the linear relation of the variation of parameters random variables to the partial deviations of the observations random variables.

Corresponding covariance parameters matrix is then simply given from its conventional definition, taking the form:

$$\Theta = \mathbf{M} \cdot \Psi \cdot \mathbf{M}^T, \qquad (13)$$

where:

matrix Ψ is a data covariance matrix, presumed known.

The variables of Δp_l, $l = (1, ..., m)$, under holding the presumed conditions, are the linear functions of the vector $\Delta u = (\Delta u_1, ..., \Delta u_k)^T$. The confidence intervals associated with the estimated parameters are then simply assessed basing on confidence regions of probability distribution function of Δp_l.

9. NUMERICAL UNCERTAINTY

Quantifying the software impact on computed results uncertainty leads to an analysis of its numerical uncertainty [6]. Numerical uncertainty of solution is subject to a lot of approximations and limitations in software designing and implementation (e.g. effects of floating point operations, translations, normalizing). Again instead of distinguishing these contributors in numerical analysis, much easy the comparison methodology can be employed, regardless the sources of numerical inaccuracy. An uncertainty can be estimated basing on an interval assessed by comparing the software calculations after applying the data sets against corresponding "true" results. The mathematical model for each geometric feature, expressing the relation between input and output quantities, is known. "True" value is replaced thus with model value, which is assumed in experiment plan. But even basing on mathematical formula no one relies on the model values if the reference data set does not meet special conditions concerning the way the solutions were achieved in numerical algorithms.

This problem can be stated as follows. The objective is to design data sets such that the data are varying according to the random phenomena and to established strategy. At the same time actual computed values should be indeed unchanged and equal to stated model values, excepting their numerical uncertainty.

Numerical uncertainty evaluation can be distinguished in the following way. Least squares estimates of these parameters minimizing the objective function Q (2) are an

projection of actual, dependent variable onto plane, defined by independent variables being measured. Hence the least squares solution should be characterized by property:

$$\mathbf{J}^T r(u, p) = 0 . \qquad (14)$$

An equation (14) imposes conditions for each independent variable in minimized function. They determine the particular requirements involving data set designing [7]:

$$\sum_{i=1}^{n} r_i = 0 , \sum_{i=1}^{n} u_i r_i = 0 , \qquad (15)$$

where u_i denotes i^{th} value of one from k independent variables, n is a number of observations equal to number of data.

The null spaces technique [8] can provide a solution to such problems (15). Null space of mxn-dimensional matrix $\mathbf{J}$ is a set of all vectors with m components for which product of this matrix and the vector equals zero. To get the vector values a factorization of transpose of Jacobian matrix has been employed by means of singular value decomposition (16) [9].

$$\mathbf{J}^T = \mathbf{USV} . \qquad (16)$$

For presumed matrix $\mathbf{J}$ the columns v_i of $\mathbf{V}$ matrix, such that corresponding s_i are equal to zero form an orthogonal basis of null space of $\mathbf{J}^T$. Basing on null space properties, the next step in data set generation is scaling residuals by choosing the factor from normal distribution. The generated points should lie at computed distances from feature surface specified by the model parameters. The absolute differences between model and actual values give the interval, from which the uncertainty u_N can be assessed.

10. THE PRACTICAL EXAMPLE

According to standard [2] validated program can be accepted as providing results fit for their intended use if none of the differences, designated conventionally by q, between test results and reference values does not exceed the prescribed ranges of errors, agreed as specification for particular application area of software. Due to these quality metrics that would yield the requirements an uncertainty should be finally established that is consistent with philosophy of Guide [6].

We can expect that the values obtained from the estimation may give us pessimistic judgment of uncertainty. Indeed, using ranges of perturbations and triangle inequality one can expect overestimated effects of combined bias. Greater reliability of results may be obtained by planning an experiment such that it more mimics the reality.

Let us illustrate the concept of accuracy estimation with the example for circle program.

10.1. Specification

For a circle in plane we take the intrinsic parameters $p = (x_o, y_o, R)^T$ having model values: center in x

axis x_o =0,000 mm, center in y axis y_o =0,000 mm, radius R=25.000 mm. Suppose the data set $\{x_i, y_i\}_1^n$ consists of n =8 points, uniformly distributed on the circular profile. They are subject to perturbations assigned to given points (measured workpiece and strategy) and to random errors (measuring device, environment, operator). Let the range of perturbations in data in each axis amounts to 0,002 mm, the random effects have properties of white noise with performance $N (0, \sigma_x = \sigma_y =0,002$ mm). The residual function for a circle feature has known form:

$$r_i(x_i, y_i, x_o, y_o, R) = f_i - R ,$$

where: $f_i = \sqrt{(x_i - x_o)^2 + (y_i - y_o)^2}$,

The matrices $\mathbf{J}$ and $\mathbf{H}$ are arranged by partially differentiating with respect to x_o, y_o, R .

The reference data set (Table 2, Fig.2) respecting condition (14) makes possible the numerical uncertainty we get.
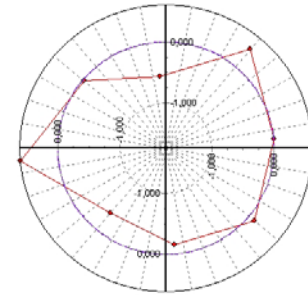

Fig. 2. Distribution of 8 points on a circle profile

Table 1. The reference data set.

Number of point	x_i , mm	y_i ,mm
1	24,9049	2,1789
2	16,3089	19,4362
3	2,1295	24,3406
4	-19,1308	16,0526
5	-25,5168	-2,2324
6	-15,8404	-18,8778
7	2,1658	-24,7554
8	19,2379	-16,1426

10.2. Numerical results

Computed values of the bias (9), bias (11) are shown in Table 2.

For estimating a precision given (13) the mean values of $\{\bar{x}_i, \bar{y}_i\}_1^8$ is assessed after 100 generations of data sets and respectively $(\bar{x}_o, \bar{y}_o, \bar{R})$ are calculated based upon values acquired through circle fitting to these generated data sets. Corresponding uncertainty U_P, numerical U_N and overall U are shown in Table 2 as well.

Table 2. The simulation results.

Contributors	x_o , mm	y_o ,mm	R ,mm
$Bias(p^*)$	1,061E-4	1,060E-4	2,10E-4
$Bias(\bar{p})$	0,050 E-4	0,0601 E-4	0,0901 E-4
U_P	2,03E-3	2,02 E-3	2,9 E-3
U_N	2,68E-6	1,72E-6	1,01 E-6
U	2,093E-3	2,109E-3	3,043E-3

11. DISCUSSION

Modeling technique and simulation investigations conducted on presumed models is the ground of validation approach presented in this article. Several methods have been proposed. Generally, the procedure to follow in uncertainty analysis relies on identifying all relevant and significant sources of errors affecting accuracy, qualifying them to contributing a bias and a precision separately, creating an appropriate mathematical model individually for each of them and finally aggregating all representations as contributors to combined uncertainty. An uncertainty due to numerical round-off effects is taken into consideration in uncertainty analysis as well.

The methods may be useful in formulating the criteria for the assessment of compliance of software results with requirements. Basic rule of standard procedures [2] is that the outcomes of software subject testing have to be compared with corresponding reference values. It allows establishing if there are no essential discrepancies between them. For that reason in the context of metrological software validation the uncertainty estimation through determination of coverage interval is important.

Nevertheless our approach has this advantage that an effort has been made to respect a bias in uncertainty statement. Highly significant attention is put on dominating random effects in majority standards in respect that the measurement results are susceptible to many unidentified sources of inaccuracy. Taking strong dependence of results inaccuracy on measurement strategy into account each arranged measurement in coordinate metrology may be regarded as particular measurement method and therefore be evaluated by appropriate metrics [10]. The measurement results according to this particular "method" may be considered to comply with requirements regarding prescribed bias limits (results are reliable) and a precision (results are certain) if they were specified for this "intended use" of software. The authors claim systematic errors as recognizable with sign and value components of results, which can and should be use in corrections. The bias analyzed here characterizes better the properties of indirect measured variables depending on given measurement model and on properties of direct measured variables.

The properties of least squares estimate are still under interest because of big popularity of least squares estimators in practice. We have been referred to early works directed to

this area, exploring and applying Box theory and Bates-Watts technique to deal with problem of bias estimation.

However the question appears of haw to combine a bias and a precision in uncertainty if the bias varies depending on many factors and an assessment it's limit is only possible. This kind of information estimated from judgment and experience according to recommendations [6] is usually classified as uncertainty of type B. Authors suggest these two components should be separately estimated, tested and apart from uncertainty interval separately reported in validation certification.

12. CONCLUSIONS

Summing up the formal model for bias and precision estimation developed here due to computer implementation allows carrying out practical software examination as well as utilizing it in software validation.

Unlike to others methods applied in validation alternatively, it makes possible monitoring of chosen variable behavior, and allowing an analysis of the impact into overall uncertainty of certain separately considered factors.

Disadvantageous aspects of the evaluation using the modeling technique are unfortunately appearing likewise. This way of validation needs the model building and, as also in such cases, its adequacy remains a factor deciding on effectiveness of usage, that is an undoubtedly shortcoming. The validation requires the pre-validation, checking if a coherence of the simulated and real data appears, on which the inference about accuracy is to be carried on.

The methods used in software validation are under development. The procedures are still improved and brought to maturity like the metrological software is at the same time changing. It is worth pointing out that similar analysis can be extended to other nonlinear models and therefore it can be useful in many software applications in metrology.

REFERENCES

[1] PN-EN ISO/IEC 17025-1: 2005, "General requirements for the competence of testing and calibrations laboratories", PKN Warszawa.
[2] PN-EN ISO 10360-6:2003, "Estimation of errors in computing Gaussian associated features", PKN Warszawa.
[3] M. J. Box, "Bias in nonlinear estimation", J. Roy. Statist. Soc., Ser. B, no 3, pp. 171-201, 1971.
[4] D. M. Bates, D. G. Watts, "Relative curvature measures of nonlinearity", J. Roy. Statist. Soc., Ser. B, no 1, pp. 1-25, 1980.
[5] H. Nieciąg, "Accuracy analysis of metrological programs output parameters for coordinate measuring machines", Scientist J., no 87, ISSN 1896-9216, 2007.
[6] "Guide to the Expression of Uncertainty in Measurement", ISO, Geneve 1995.
[7] http://www.npl.co.uk/.
[8] http://en.wikipedia.org/.
[9] W.H. Press, S. A. Teukolsky, "Numerical Recipes in C. The Art of Scientific Computing", C. U. Press, 1995.
[10] ISO 5725-1: 1998, "Accuracy (trueness and precision) of measurement methods and results – Part 1: General principles and definitions".
[11] ISO/IEC Guide 99: 2007, "International vocabulary of metrology-Basic and general concepts and () terms (VIM) ".