Improvement of Simulation Method in Validation of Software of the Coordinate Measuring Systems

Halina Nieciąg

Faculty of Mechanical Engineering and Robotics, AGH University of Science and Technology, 30 Mickiewicza Av. , 00-059 Cracow, Poland, hnieciag@agh.edu.pl

Software is used in order to accomplish various tasks at each stage of the functioning of modern measuring systems. Before metrological confirmation of measuring equipment, the system has to be validated. This paper discusses the method for conducting validation studies of a fragment of software to calculate the values of measurands. Due to the number and nature of the variables affecting the coordinate measurement results and the complex character and multi-dimensionality of measurands, the study used the Monte Carlo method of numerical simulation. The article presents an attempt of possible improvement of results obtained by classic Monte Carlo tools. The algorithm LHS (Latin Hypercube Sampling) was implemented as alternative to the simple sampling schema of classic algorithm.

Keywords: Validation, coordinate measuring technique, Monte Carlo simulation.

1. INTRODUCTION

MODERN COORDINATE equipment for industrial applications are complex measurement systems. They include traditional and hybrid measuring machines, mobile coordinate instruments, optical and laser tracking systems [1].

One common feature of coordinate measuring systems, regardless of different constructions, configurations and designation, is the digital form of measurement information and the algorithmisation of the operations being performed.

The measurands reflect the geometric features of a product, which is modelled using lines and surfaces. The first stage of coordinate measuring is direct measurement, meaning the determination of locations of points belonging to the physical object being measured within a fixed coordinate system. Digital data are then further analysed, while the results of their digital processing are interpreted, visualised and reported by the system. The first stage involves the equipment layer and the system software layer, while the next stage is predominantly software-based. The quality of the resulting measurement information thus strongly depends on software quality.

One form of the metrological confirmation [2], which aims to determine that the equipment used in the measurement meets the requirements for its intended use, is the validation of the measurement system. Validation is a special case of confirmation, in which the conformance of the characteristics of the object under validation is determined not in the relation to the declared specification, but to the predetermined criterion for a given application in real-life conditions of operation of this object [3].

Software validation is an essential stage of measurement system validation, with special attention paid to the studies and analyses of the precision of algorithms used to calculate measurand values. Evaluation of the quality of information obtained from multi-level processing involves the evaluation of measurement uncertainty of measurands. The standard involving the issues of testing the software for calculating geometry is PN-EN ISO 10360-6 [4]. However, it appears not to be sufficient, e.g. in a context of changing dynamically the deviations of the measured object.

This paper presents a fragment of research into this issue, concerning a module of applications used to determine the numeric values of parameters of measured geometric shapes. A software fitting the data to spherical surface has been subject to validation tests. The analysis was based on a simulation model, in which all variables were treated as random. It allowed for inferring the behaviour of the tested software in various circumstances of its operation.

In order to estimate an expected value and a coverage interval characterizing the accuracy of output parameters, two sampling methods have been applied: classic Monte Carlo sampling method and LHS stratified method. These methods were examined taking into consideration their effectiveness and ability to improve LHS estimators over these from classic simulation schema. Evaluation of the results has shown a slight improvement in the size of coverage intervals estimated from LHS samples, but the advantage of LHS algorithm became apparent in reduction of the expected value variance.

2. MEASUREMENT MODEL

The product development process is based on the geometric model, built in the design and construction process. Being consistent with the design concept, the geometrical software of the system allows for the verification of real shapes by using embedded algorithms of parametric identification of the geometrical elements, so-called associated features. The algorithms based on Gaussian criterion allow calculating the values of characteristic parameters of measured geometric shapes.

The measurement model for each element, showing the relationship between input and output quantities, can be a mathematical model, which can be described by a general formula (1):

$$h[\ X, P\ ] = 0\ , \tag{1}$$

where: $X = [X, Y, Z]^T$ represents a vector of k–input quantities ($k=3$) or the coordinates of points measured in a coordinate system, output values P have the form of m–dimensional vector of model parameters, i.e. the quantities characterising the size, locations, etc. of measured shapes.

The essence of the model is to create an image of an idealised form of real objects [5] - physical objects or software units, or others.

Owing to this image, we gain a medium to foresee the behaviour of real software through the use of the computer simulation tools, reality simulation.

3. THE CONCEPT OF VALIDATION

The results of measurements, and thus the values of the measurands P, (1), depend in coordinate measurements not only on input quantities X, but also on many other factors, such as measurement strategy, tolerance class, product's surface, datum, location within the measurement space, etc. In coordinate measurements, these factors are classified according to five main categories assigned to locations of their occurrence. This division is aimed at facilitating the identification of factors and the choice of the appropriate method of determining their influence on the model's output quantities.

The model should reflect the significant aspects of the object being modelled, but the factors influencing it are not explicitly included in formula (1), even though they can be qualified as input quantities, as measurement results depend on them. The occurrence of certain factors during the measurement, as well as their level, form the conditions of model (1) realization. The arrangement of these factors determines the intended use of the software of the measuring system.

The system's software can thus be considered as fit for purpose if the results obtained in this particular, deliberate application of it appear to be consistent with reference results, that serve as a computational standard, i.e. describe the behaviour of a perfect calculating software through a model (1).

Validation, as the process of software quality assessment, carried out basing on the model, has three very important stages (Fig.1.).

The first stage is the determination of the measure of the discrepancy between the calculations of the real software and the reference results, while the third one involves the determination, when this discrepancy is acceptable for the software's intended use. The first stage requires the adoption of the metric, which enables a comparison of experimental and standard results as a quantitative measure of comparison.

The third stage is to decide on the adequacy. But before this decision is made, the criterion of adequacy of the software for a particular application must be specified during the second stage of validation.

The determination of discrepancy requires a separate analysis of the measurement model, used in given circumstances in the coordinate system. Flexibility of the coordinate measuring processes arbitrarily carried out by the system's user makes it harder to use a single universal course of action. The diversity of features characteristics

that are measured, as well as methods to obtain these goals, impose the need to define the formulation of a specific – task measurement, the so-called *feature based measurement*. It is a measurement goal separate from other tasks and defined by the measurement model, where the quantities related to this model may change in a given way and within given boundaries.
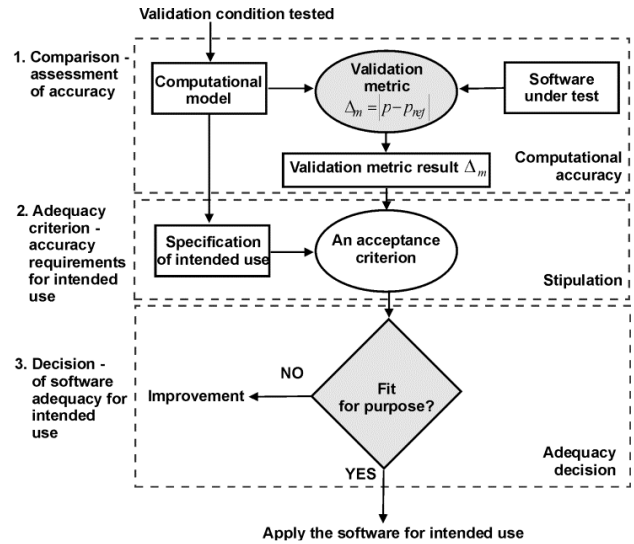

Fig.1. The validation diagram.

3.1. Geometric software validation metric

Metric is the quantitative measure of inaccuracy, specifying the degree of discrepancy between the measured values and the corresponding reference values. Metric does not have a precise definition within validation, but this term is used in computer science to mean the quality index of a given attribute, characteristic of a measurement system or its specification. The calculated value of the metric is interpretable as the degree of fulfilment of a certain qualitative property of the software unit. A geometric software metric should be related to a specific attribute of the geometry of the measured object in a way that will assess the quality of its mapping through measurement.

Metric development relies on the formulation of an appropriate measuring function [6]. In the geometry inspection of an object, the measurands are the linear and angular quantities. For these values, the metric can be expressed as the following relationship (2).

$$\Delta_m = \left| p - p_{ref} \right|, \tag{2}$$

where: Δ_m – absolute difference between the values attributed to the measured quantity and the corresponding reference value; p – test result; p_{ref} – reference value.

Deviation δ_l^r of variable p_l (1) of a single say r–test is the difference between the value calculated by the software, e.g. the value of the specific feature parameters such as diameter, the position of the point, etc., from the corresponding reference value, included in the general equation (3).

$$\delta_l^r = \left| p_l^r - p_{l,ref} \right|, \qquad (3)$$

where: δ_l^r – calculated deviation, p_l^r – the measured value, $p_{l,ref}$ – reference value, $l=(1,\ldots,m)$.

An easily implemented measuring function (3) is in fact a mathematical operator that requires the entering of two values, and it embodies the traditional approach to validation, which involves the comparison of two deterministic variables (Fig.1.).

The first component of the formula (3) is representing the individual results of the validation tests.

The nature of variables involved in the implementation of the model (1) and affecting the measurement puts into question the results of software evaluation based on a single test result. In real-world work conditions, the tested software may yield different results in each session, as the working conditions of a real application clearly eliminate the possibility of obtaining an identical set of data. Confronting a single test result with the level of user requirements, there is no certainty that the result will be repeated when the application is relaunched or that it is going to be close to the consecutive results in a series, or that it is going to be an outlier, and this distinction is important in the statistical evaluation of the analysed variables.

Simulation experiments require the person constructing the tests to imitate the real working conditions for the software by generating data that would allow for obtaining realistic results.

The second element in operation (3) is a reference quantity, so the value in question is well-known, presumed or agreed upon [7].

However, in most cases the determination of the correct reference value is not an easy task, as this value is a function of an imperfect and partial knowledge of the measurement process. This fact also means that the evaluation of a single test result is not easy.

In simulation validation tests common practice is to adopt the parameters of the modelled shape as reference values. The assumed values describe an ideal form by the nominal parameters and geometric relationships between individual objects of the part modelled during geometry design phase.

3.2. Testing conformity with requirements

Metric values calculated based on tests (2) do not directly tell us whether the values are acceptable until they are verified by comparing them with the assumed numeric criteria, the acceptance criteria [8].

The evaluation of acceptability of the calculated characteristics requires us to define the boundary acceptable error value. The absolute difference (3) will not be significant if it is smaller than or at least equal to the value adopted as the acceptance criterion, equal to, e.g. $\Delta_m(\delta_l)$, (4).

$$\left| \delta_l \right| \le \Delta_m(\delta_l). \qquad (4)$$

The boundaries of this acceptable discrepancy define an interval, which should include the errors in obtained experimental values of the variables, so that the software results can be considered to be in conformity with the requirements. From the metrological point of view, the range characterising variability of the results of measured quantities at a given level of probability may be represented by the uncertainty of measurement, U_Δ [9].

In the case of software for parametric identification of associated features, the acceptance criterion, i.e. the subjective expression describing the end and success of a test, is expressed as intervals that should not be exceeded by the uncertainty of parameters.

The endpoints of this interval should be adjusted to and be valid for a set of levels of factors of a test experiment [8]. They reconstruct the intended use of software included in a test case. Only in such case can we expect a reliable evaluation of the software.

4. ESTIMATION OF UNCERTAINTY

The assessment of the accuracy of measurement results involves the use of different methods of estimation of uncertainty. They are generally based on two approaches: one related to the theory and propagation of uncertainty, the other based on the probabilistic method of distribution propagation. They share a common philosophy that treats the measurement as the process of implementing random variables, with part of them participating in the measurement process, while another part comes from sources outside of the measurement and indirectly influences its result. The use of information concerning these variables, as well as their processing is done according to the probability theory.

Determination of the range, within which the value of the variable will be included according to the probability theory, requires the knowledge of the distribution of this variable. If the measurand is characterised by the probability density function, two parameters characterising the distribution of it, such as the expected value and the standard deviation, describe the best estimate of the measurand and the standard uncertainty related to this estimate, respectively.

The determination of the probability distribution of the measurand based on the input quantity distributions is called the propagation of distributions, regardless of the method of calculation (analytical, numerical). The *Monte Carlo Method* (MCM) is the propagation of distributions which utilises repeated random generation of data by the so-called sampling of probability distributions [9 - 11].

5. RESEARCH OUTLINE

Tests are preceded by the structural decomposition of the software of the coordinate system. So-called partial validations are performed on isolated software components used to verify the geometry of the inspected part. The division into elementary measurement tasks naturally corresponds to a structural division of the system's software module into smallest parts: programs which perform these tasks by calculations. Validation may be based on models

presented in (1), but the "extended" measurement model will also be defined by the selection of factors that form the circumstances for a given use of the program, [12].

The goal of the validation tests was to study the fitness of the programs for their intended use. It is done by determining the uncertainty of the measurement task result in the circumstances defined by the strategy, in order to check if it meets the assumed acceptance criterion.

5.1. Simulation model

The model of quantities $X = [X, Y, Z]^T$ describing the point's coordinates X, Y, Z, is in fact a probabilistic model of the variables of their error. The implementation of each random variable X, Y, Z is the sum of its "real" value and its changing deviation. The real value is considered to be known based on the model assumptions and it reflects to the coordinates of the point lying on an ideal surface or a profile of the associated feature. Deviation is a superposition of a number of implementations of random error variables. In building the model of input quantities, an idea analogous to the vector concept of coordinate device errors can be employed [13], where the cumulative error is the superposition of partial errors generated by certain factors or stemming from certain sources.

Software validation only takes into consideration the factors that may influence its evaluation and, according to the goal of validation, allow for the formulation of a statement, that the software is fit for its intended purpose. It is assumed that the remaining ones, accompanying the real measurement, such as the method of fixing the object, surface soiling or its temperature, all belong to this group, which is subject to the obvious laws of measurement technique, so obeying these laws can significantly limit their influence on the results.

In the measurement model adopted for the study of geometric software using the MCM, we assume the presence of two factors that cannot be eliminated by conducting real measurements, regardless of whether a correct measurement practice was used.

The first factor is related to the influence of the real geometric form of the measured object on the result. Product inspection is usually conducted by analysing the individual components of the structure, but the coordinate measurements of geometric quantities consider the more generalised effect as a composition of the surface structure, shape deviations and other deviations from the ideal geometric form.

Taking into account the periodic nature of the irregularities on machined surfaces, the total error can be modelled by the sum of component variables expressed in a general form of elementary harmonic functions. As a result of the influence of independent groups of technological factors on the process, the amplitudes and phases are considered as random variables in the model.

The second factor determines the influencing quantity, which is a composite of all unidentified sources of uncertainty. The variable models a cumulative direct measurement error, which can be expressed by both random and systematic factors.
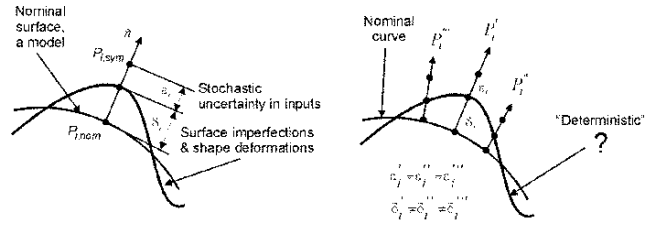

Fig.2. Error simulation model.

By analysing the end result, we can notice that the final value of the error, as a superposition of realizations of various random variables, stems from the location and range of the characteristic of distributions of these variables (Fig.2. – "left" outline).

The study used an universal concept of error modelling by superimposing these two components [14]. The nature of these components may vary depending on which of the groups of factors are considered active and significant.

5.2. The impact of measurement strategy

The strategy significantly affects the uncertainty of the evaluated measuring task.

It is a powerful and underrated means of controlling coordinate measurement uncertainty, superficially ascribed to the area of measurement technique.

According to [15]: "*The accuracy of the results from coordinate measurements depends on the accuracy of the measuring device, workpiece properties, environmental conditions, and especially operator procedures. For the last, and usually most important factor, neither are proven facts known about the combined effects of the various influences nor do any general quantitative statements exist.*"

Coordinate measurement strategy is determined by the quantity and locations of points on the surface being sampled, the size of measurement sections, the direction of measurement, the type of probe, even the sequence of measured points. Changing any of these parameters will change the conditions of the experiment and become a deviation from the repeatability conditions. The evaluation of the measuring task relates to a given level of all factors, within which this task is executed.

The impact of strategy on the form of the variable of the input quantity error is the most visible when generating data encumbered by deviations stemming from the simulation of the real surface condition. The superimposition of the harmonics on ideal form of the modelled surface gives an image of a "deterministic" surface until we obtain a point's location random in relation to this surface while imitating a real measurement (Fig.2. – "right" outline). The superimposition of the harmonics due to the random impact of the factors in the technological process further strengthens the final random character of the deviation of surface structure from the ideal surface and should be reproduced in a simulation experiment.

The determination of the composition of distributions of error variables using analytical methods is quite cumbersome in practice and may lead to overestimating the variability ranges in measurands.

The key issue becomes how to best reconstruct the distribution of input random variables by a properly designed simulation experiment, as the distribution of analysed output variables is probably different from Gaussian or from any other model distribution. The results presented in [16] as well as own studies show that in many practical cases, the resulting distribution of cumulative deviations including few harmonics differs, at given point, essentially from the Gaussian distribution. Estimating coverage intervals for variables not fulfilling the standard model assumptions without the knowledge of the analytical function of distribution is possible if the experimenter acquires the numerical form of the probability density function (*pdf*) or the cumulative distribution function (*cdf*) of these variables.

6. NUMERICAL MONTE CARLO METHOD

6.1. *The essence of MCM in testing geometric software*

The essence of the MCM algorithms used in testing geometric Gaussian programs is the random generation of values of partial errors for a given set of points, and then the calculation of the measurand values using the software entity being tested. The cycles are repeated multiple times, so as to generate a discrete representation of the distribution of these variables.

The simulation model is based on the variables model described in chapter 5.

In order to obtain the coordinates, e.g. for n–points: $\{P_1(x_1,y_1,z_1),...,P_i(x_i,y_i,z_i),...,P_N(x_n,y_n,z_n)\}$, in each cycle of calculations, the generated errors are added to the coordinates of points lying on the modelled theoretically ideal surface. The location of these points is derived from a mathematical description of the feature.

Based on multiple, e.g. M times generated random sample $\{x_i, y_i, z_i\}_{i=1}^n$ the unit being tested calculates the values of an m-dimensional quantity $\boldsymbol{P}$, leading to the creation of an enormous set of these values. This allows the determination of the empirical density function *pdf* or, after sorting the values from the smallest to the largest, the empirical distribution function *cdf* for each component of $\boldsymbol{P}$. They provide a numerical approximation of the distributions of output variables.

The discrete representation of the distribution allows for determining the estimate of the output quantity (measurand) value, standard deviation as standard uncertainty associated with the estimate, and the endpoints of the coverage interval y_{low}, y_{high}, the length of which characterises the variability in the measurement process [10]. It is an interval, which includes, with certain probability, the true value of the measurand.

Traditionally, it is assumed that the point ends of the coverage interval are determined by the quantiles of 0.025 and 0.975, which correspond to the coverage probability p=0.95. The assumed value of probability is symbolic, in relation to the normal distribution, for which it corresponds to the coverage factor $k = 2.0$.

6.2. *Classic algorithm ("crude sampling")*

In a simple, crude Monte Carlo sampling, random variables with any distribution can be generated using the reverse *cdf* method. It is based on a simple proposition that when any variable $\boldsymbol{X}$ has a continuous and strictly increasing *cdf* U=G$_P(\boldsymbol{\eta})$, $U < U(0, 1)$, $\boldsymbol{X}$ can be calculated as its reverse $\boldsymbol{X}$= $G_P^{-1}(\boldsymbol{\eta})$. For an a priori known or empirical probability density function (histogram), $g_P(\eta)$, the *cdf* is then analytically or numerically determined U= $G_P(\boldsymbol{\eta})$. Approximations of an inverse function are determined in independently drawn points according to the uniform distribution from the interval [0, 1], i.e.: $\boldsymbol{X}$= $G(U)^{-1}$.

The classic algorithm for multidimensional input variable $\boldsymbol{X}$ in one simulation cycle is as follows:

1. Selection of M number of repetitions.
2. Randomisation with a probability of 0 to 1 value u_i for each k-dimensional variable $\boldsymbol{X}$ of a single i–point, whereby we obtain a set $\{u_{i,1}, u_{i,2}, u_{i,3}\}$ for three coordinates.
3. Each randomly selected value u_i is assigned to the corresponding variable $\boldsymbol{x}_i$.
4. The drawing is repeated for each n points, creating $n \cdot k$ elements sample of input variable $\boldsymbol{X}$, on which calculations are performed in accordance with model (1).
5. From a set of M–values of m–dimensional variable $\boldsymbol{P}$ calculated in the experiments, an estimate of expected value of output value is deduced, also known as the *empirical median value*.

Due to the randomness of the experiment, each test sequence can give a different estimate, but the series of values will stochastically converge with probability 1 to the expected value of the measurand. The difference between an individual estimate obtained from a single cycle (estimator value) and its expected value can be defined as MCM error. It can be expressed by estimator variance, which is proportional to the variance of the probability distribution, and inversely proportional to the number M of experiment cycles performed [10]. Both factors affect the quality of the MCM results.

Increasing the number of M recurrences is an intuitively chosen means to improve the accuracy of estimates using MCM. The number of cycles recommended in [10] needed to achieve coverage interval accuracy of two significant digits and for 95 % confidence level is 10^6. Accuracy control may be done, among others, using adaptive procedure, recommended by the above mentioned document.

The relationship between the estimation error and the variance of the distribution of output variables shows us, that the correct selection of randomly sampled points from the probability distribution may influence the quality of results, as well as decrease the number of experiment recurrences needed to achieve the necessary accuracy. The improvement of sampling uniformity in relation to the shape of function describing distribution may lead to reduced variance.

Methods for improving the drawing efficiency and reducing the variance include stratified sampling methods and quasi–Monte Carlo methods [18]-[19].

It should be noted that one important benefit of the classic MCM algorithm is the fact that the number of simulations needed does not depend on the number of random variables modelling the input values. This property, along with easy implementation, is the main reason for the popularity of the classic Monte Carlo algorithm.

6.3. Stratified sampling method

LHS (*Latin Hypercube Sampling*) is a *multivariate extension of stratified Monte Carlo method*, which is intended to improve uniformity of sampling and to lead to reduced variance [19]-[20].

In this sampling method, the whole total range of the *cdf* of each component variable is divided into separate sub-areas with equal probability. Each of the created ranges is used to calculate one value for each component X, corresponding to the sampled value of probability density in the chosen sub-area. The name LHS comes from the generated N-values of variables, forming a topologically N–dimensional cube (hypercube).

The algorithm for the method of k-element vector of random input variable X in one simulation cycle for one "sampled" point is as follows [20]-[21]:

1. The division of the probability density curve for say ii– variable ($ii= 1,…, k$) into R equal disjoint ranges forming layers with equal probability, in which the changes of probability function are smaller than in the whole range of values;

2. Sampling R independent random numbers $u_{ii,1}$, $u_{ii,2}$, ...,$u_{ii,R}$, from the interval [0, 1];

3. Random generation of a single number in a single simulation cycle for each of k–components, containing R–repetitions in relation to the LHS algorithm. The number represents the number of the layer, from which $u_{ii,j}$ ($j=1,...,R$) is taken. Randomly sampled numbers form a permutation of the layers, on which the components are grouped.

4. Based on the permutation, a set of sampled ($N= R^k$)·n values of the input variable X is created, (1). Each individual data given in the n–elements sample, defined by X, is a vector of k–components (k=3), where each component (coordinate of datapoint) may come from a different layer.

5. The calculations are performed for the full set of n–points, which result in a set of m-values of P.

According to the MCM concept, after calculations are completed, steps 1−5 are repeated M times.

7. EXPERIMENTS AND RESULTS

The purpose of the comparative simulation experiments was to examine how effective Monte Carlo methods may be in the estimation of uncertainty of parameters calculated by Gaussian software. Two different methods of sampling were used to generate the coordinates of points for the spherical surface: the classic Monte Carlo algorithm and the LHS algorithm.

The tests used model data from an example constructed and published in [22].

The data form a regular grid of n=30 points spaced on a half of a perfect spherical surface (Fig.3.).
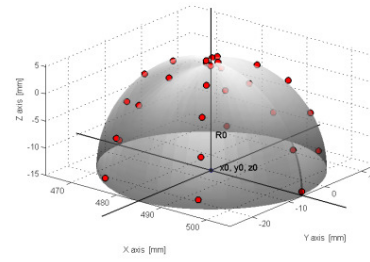

Fig.3. The simulated sphere.

A single data is a vector of three coordinates in a coordinate system that is offset in relation to the centre of the sphere by a given vector. The components of the vector of parameters describing the perfect sphere are: $P_o^{\mathrm{T}} =[$ x_o =484.9407, y_o =−6.1794, z_o =15.1096, r_o =20.000] mm.

Calculated in [22], the values of measured coordinates of the ith –point (P_i) were a sum of values ($x_{i,nom}$, $y_{i,nom}$, $z_{i,nom}$) corresponding to the point on an ideal sphere with parameters of P_o, arbitrarily assigned values of systematic error ($\Delta_{x,s}, \Delta_{y,s}, \Delta_{z,s}$), and one of the M–generated values ($\varepsilon_{i,x}^j$, $\varepsilon_{i,y}^j$, $\varepsilon_{i,z}^j$) of random error.

In the tests presented in this paper, error components are simulated consequently with the simulation model discussed in sub-chapter 5.1. Coordinates of the ith point (P_i) are a sum of reference values ($x_{i,nom}, y_{i,nom}, z_{i,nom}$) as above, the temporary values ($\delta_{i,x}^j, \delta_{i,y}^j, \delta_{i,z}^j$) of form deviation δ_i^j at i^{th} point, and the generated values ($\varepsilon_{i,x}^j, \varepsilon_{i,y}^j, \varepsilon_{i,z}^j$) of random errors in each axis for the j–cycle of simulation and calculations (Fig.2. – "left" outline).

The values of random errors are one of the ($j = 1,…,M$) random combinations generated using a pseudo–random number generator for the assumed normal distributions of these errors, with parameters $N(0, \sigma_{\varepsilon,x})$, $N(0, \sigma_{\varepsilon,x})$, $N(0, \sigma_{\varepsilon,x})$, where: $\sigma_{\varepsilon,x} = \sigma_{\varepsilon,y} = \sigma_{\varepsilon,z} = \sigma_\varepsilon$ =0.002 mm.

Mathematical modelling of sphericity deviation is based on the notation of a chosen surface by a formula using the associated Legendre's functions [23]. The tests included the following: harmonics = 3, amplitude = 0.0050 mm, order = 2, phase = 3. The form deviation values δ_i^j in each variant of the j–cycle of simulation are determined for randomly adopted phases. Deviations vary randomly from cycle to cycle in the range equal to amplitude of the harmonics.

The numerical value of the simulated coordinate (x_i^j) is calculated on the basis of (5).

$$x_i^j = x_{i,nom}^j + \delta_{i,x}^j + \varepsilon_{i,x}^j. \tag{5}$$

The numerical values for the remaining coordinates of point (y_i^j) and (z_i^j) are calculated analogously.

The combined random variable imitating the direct measurement in each of the 3 axes is composed of two independent random variables: one arising from a random distribution of deviations in sphericity at a given point of the spherical surface $\Lambda : \Omega = \left\{ \delta^j \right\} \rightarrow \Re^n$, and the other modelled by Gaussian distribution $E \in N(0, \sigma_\varepsilon)$, and its distribution is a convolution of two distributions, which these variables are subject to. Propagation of uncertainty of the results of geometric software requires knowledge of this distribution. In numerical methods, we can observe its numeric representation as a result of the convolution of the component distributions, which is undoubtedly a great advantage of this concept.
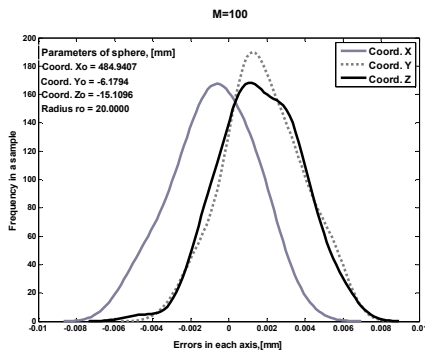

Fig.4. The empirical distributions of the simulated errors for exemplary 15th point on sphere.

In studies according to a predetermined number of simulations, *M* sets of measured coordinate values of *n*–measurement points were generated. Based on these random samples, empirical *pdf* and *cdf* in individual *n*-points on the surface were reconstructed (Fig.4.).

All the necessary simulation programs are written in the form of scripts and performed in MATLAB environment. A fragment of the application performing the geometric calculations comes with the package developed at the National Physical Laboratory, NPL's LSGE MATLAB Software (License Ref: CMSC/X/02/431).

The LHS sampling method employs the division of the variable range into 5 layers with equal probability of occurrence. The number of layers was arbitrarily assumed considering that the number of permutations generated for 5 layers for k=3 variables is a 5^3 dimensional cube.

The influence of the number of replications on the estimation by each method was examined by performing $M=50$, 10^2, and 10^3 measurement cycles. Each of the variants of the experiment was repeated three times, considering the randomness introduced by permutations.

The sizes of coverage intervals calculated using the two methods were compared.

The differences – *skew* values - between the estimates of sphere parameters and their reference values assumed for the experiment were also studied. These deviations of mean values (estimator of expected value) from the reference values may be interpreted as a measure of unsymmetrical property of distribution.

The exemplary results of constructed functions *pdf* and *cdf* for z_o are shown in Fig.5. and Fig.6. They describe the nature of this component of vector ***P***, which is most impacted by the unfavourable limitation of the sampled surface – the sampled fragment is just 50 % of the spherical surface.

The tests results have shown that values of coverage interval from classic sampling and stratified sampling are always smaller than those obtained by the GUM uncertainty framework in both tested sampling methods.

It appears that there are no notable differences in the coverage interval values calculated by two methods, however, intervals from LHS plan are smaller.

Despite this, it should be noted that the differences between estimates of expectation and respective value of model (i.e. reference value) are changing along with the number of replications of classic algorithm, whilst the LHS sample provides the stable *skew* value, independent on number *M*. Because the reference value is constant, the changes of *skew* value are due to the variability of the estimator of expected value, what results in a small value of variance of this estimator.
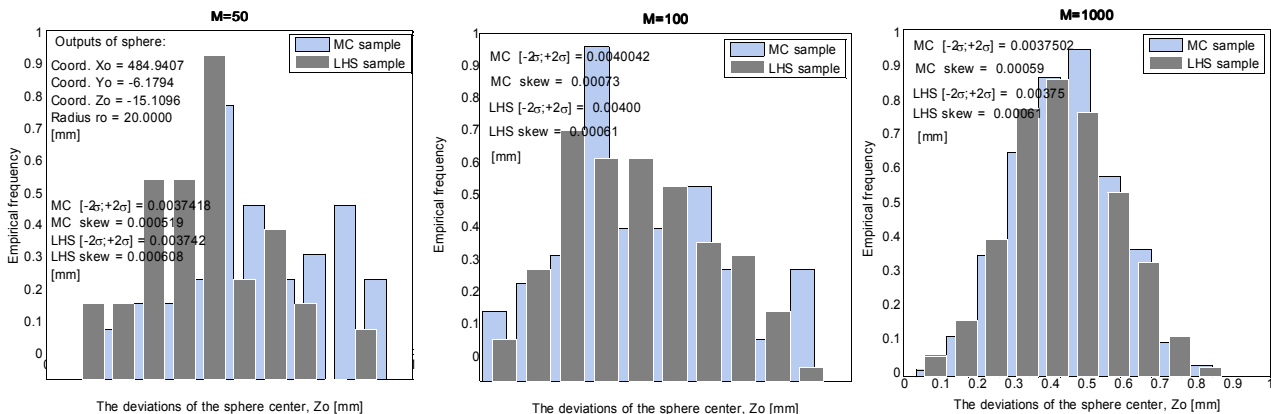

Fig.5. A comparison of *cdf* and *pdf* approximation of z_o coordinate of centre of the sphere for M = 50, M = 100, and M = 1000 replications.
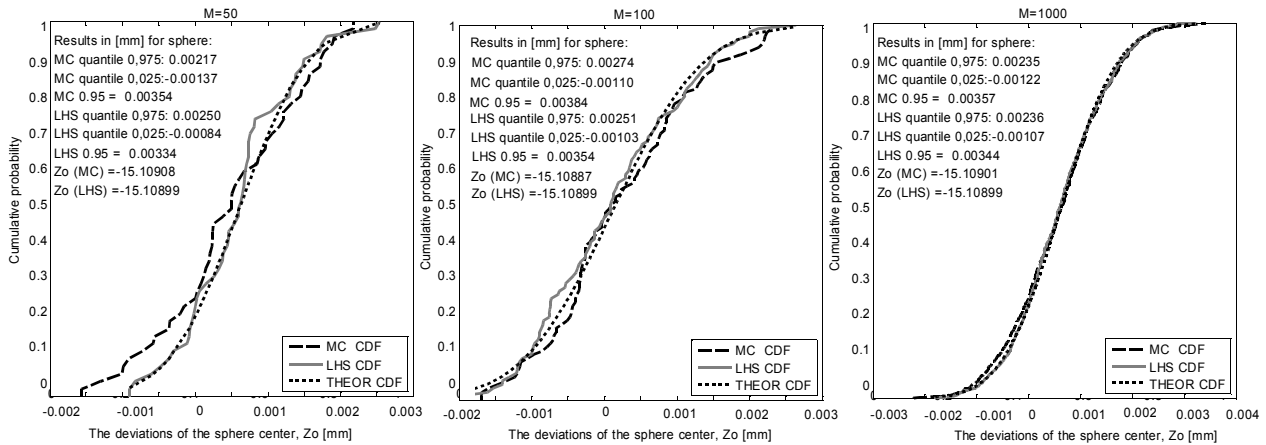
Fig.6. A comparison of *cdf* approximation of z_o coordinate of centre of the sphere for M = 50, M = 100, and M = 1000 replications.

8. COMPARISON OF TWO METHODS

Despite the same number of repetitions, yielding the same values of estimators using two methods seems unrealistic, yet such congruence would allow for considering these methods equivalent. Variation in results between the methods does not necessarily indicate a total divergence in evaluating the values of expected parameters **P** or the measures of their variations. Estimators are also random variables.

A comparative analysis of the results of the goals of the interdependence between the variables observed in both methods was recommended.

Spearman's rank correlation was used for this purpose [24]. The correlation coefficient ρ calculated in the correlation formula allows for measuring the strength of relation between the variables being compared, not only their compatibility.

Unlike the traditionally used Pearson's test, Spearman's test does not require any assumptions considering the distribution of variables being analysed, and allows for assessing non-linear dependencies. The interpretation is similar to the classic ρ – Pearson correlation measure [25].

If the value of the correlation coefficient ρ is close to 1, the relation is strong and positive. ρ values close to zero mean no relationship between the variables or a very weak relationship.

Table 1. contains test results for parameter z_o. Correlation coordinates were verified using a test of significance of the correlation coefficient at confidence level α=0.005.

The following hypotheses were stated: H$_0$ – averages differ significantly, so relationship does not exist: ρ=0, relative to the alternative hypothesis: H$_1$ – averages do not differ, the relationship of characteristics is significant: $\rho \neq 0$, using statistic asymptotic *t* approximation (6), [26].

$$t = \frac{\rho}{\sqrt{1-\rho^2}}\sqrt{M-2}\,,\qquad(6)$$

where: ρ - estimated correlation coefficient, M – sample size, and t is distributed approximately as Student's distribution with $M-2$ degrees of freedom under the H$_0$.

Table 1. Test of lack of relation between the estimate of the expected value of coordinate z_o.

Parameters		M=50			M=100			M=1000		
		run 1	run 2	run 3	run 1	run 2	run 3	run 1	run 2	run 3
$\mu(z_o)$ exp. value estimated, mm	MC	-15.10925	-15.10908	-15.10895	-15.10880	-15.10894	-15.10897	-15.10894	-15.10901	-15.10899
	LHS	-15.10899	-15.10899	-15.10899	-15.10899	-15.10899	-15.10899	-15.10899	-15.10899	-15.10899
$\sigma(z_o)$ std. dev. Estimated, mm	MC	0.00097	0.00094	0.00116	0.00092	0.00100	0.00092	0.00099	0.00094	0.00097
	LHS	0.00092	0.00076	0.00077	0.00090	0.00098	0.000898	0.00089	0.00088	0.00086
Coverage interval estimated at p=0.95, mm	MC	0.00388	0.00354	0.00482	0.00352	0.00384	0.00338	0.00392	0.00357	0.00385
	LHS	0.00364	0.00334	0.00311	0.00345	0.00354	0.00324	0.00346	0.00344	0.00341
Spearman's $p-value$		**0.93327**	0.71415	0.88587	0.51086	0.25328	0.52877	0.63458	**0.16034**	0.45544

For a smaller number of repetitions (M=50), high value p = 0.93327 was yielded, which showed a significant lack of relation between the compared estimates (if the p-value is less than α level, the null hypothesis H_0 should be rejected), [26]. As the number of repetitions increased, the results of both methods came closer, which allowed us to assume, that at M=10^3 repetitions there is a stronger relation (run1: p =0.16034) between the results which come from both methods.

9. SUMMARY

The paper presents the application of the LHS method in testing the geometric software in coordinate systems as an alternative for the traditional approach to Monte Carlo simulation. The basis for this method is the stratification of sampled distribution of input variables, whose representation is available as a result of overlapping the simulated partial errors for simulated surface. Inside each layer, numbers corresponding to the probability are sampled and then used to create the permutations of the coordinates of points.

The tests were designed to examine whether the modification of the MCM results in improved estimates of the statistical parameters of the variables that are analysed by the software. The application of the LHS method was meant to shorten the testing time, as well as to improve the reliability of estimates of chosen characteristics of the model in question.

Summing up the study, it can be seen that both methods have some advantages and disadvantages. One joint advantage of both methods is their simplicity and ease of implementation. However, sampling using the traditional algorithm of MCM is more efficient in calculations. Creating permutations with multi-dimensional variables and high-density stratification in the LHS algorithm leads to a visible increase of testing duration.

No spectacular reduction in coverage interval of z_o – output variable was achieved. In both methods, similar values of estimated coverage intervals were observed, while the samples obtained using the LHS sampling yielded always shorter ones.

The application of LHS algorithm gives more stable estimates of expected value in statistical sense, which may be important especially when the number of replications is small. The differences between these estimates and the respective value of model (i.e. reference value) seem to be not susceptible to replications in LHS algorithm, practically providing the same values irrespective of the number of trials of experiment.

It can thus be concluded that sampling using LHS yields more precise estimation of the expected value of variables calculated by the software being tested.

The statistical model used today for analysing coordinate systems is based on quasi–deterministic metrics. Experiences gained from application of the LHS algorithm allow us to believe that the employment of more efficient MCM tools will enable the design and implementation of metrics for the evaluation of geometric software quality more adequate to the statistical model of variables.

REFERENCES

[1] Ratajczyk, E. (2005). *Coordinate Measuring Technique*. Warsaw University of Technology Publisher.

[2] International Organization for Standardization. (2003). *Measurement management systems - Requirements for measurement processes and measuring equipment*. ISO 10012:2003.

[3] International Organization for Standardization. (2002). *Geometrical Product Specifications (GPS). Coordinate measuring machines (CMM): Acceptance test and reverification tests for coordinate measuring machines (CMM). Part 6: Estimation of errors in computing Gaussian associated features*. PN-EN ISO 10360-6.

[4] International Organization for Standardization. (2002). *Software and System Engineering - Software Product Evaluation - Part 1: General overview*. ISO/IEC 14598-1.

[5] Jaworski, J.M., Morawski, R.Z., Olędzki, J.S. (1995). Measuring as the parametric identification of the mathematical model of the object being measured. *Metrology and Measurement Systems*, 2 (1).

[6] IEEE. (1998). *IEEE standard for a software quality metrics methodology*. IEEE Std. 1061-1998.

[7] Joint Committee for Guides in Metrology. (2008). *International vocabulary of metrology – Basic and general concepts and associated terms* (VIM 3rd edition). JCGM 200:2008.

[8] Nieciąg, H. (2012). The assessment of the criterion in tests of acceptance type of the metrological software in coordinate measuring systems. In *10th International Scientific Conference „Coordinate Measuring Technique"*, 23-25 April, 2012, Bielsko Biała, Poland.

[9] Joint Committee for Guides in Metrology. (2008). *Evaluation of measurement data -- Guide to the expression of uncertainty in measurement*. JCGM 100:2008.

[10] Joint Committee for Guides in Metrology. (2008). *Evaluation of measurement data - Supplement 1 to the „Guide to the expression in measurement" - Propagation of distributions using a Monte Carlo method*. JCGM 101:2008.

[11] Joint Committee for Guides in Metrology. (2011). *Evaluation of measurement data - Supplement 2 to the „Guide to the expression in measurement" - Extension to any number of output quantities*. JCGM 102:2011.

[12] International Organization for Standardization. (2008). *Geometrical Product Specifications (GPS) -- Coordinate measuring machines (CMM): Technique for determining the uncertainty of measurement -- Part 4: Evaluating task-specific measurement uncertainty using simulation*. ISO/TS 15530-4:2008.

[13] Sładek, J. (2011). *The Accuracy of Coordinate Measurements*. Cracow University of Technology.

[14] Nieciąg, H., Tabisz, R.A. (2011). Simulation of influence of systematic and random factors on measurement results of coordinate measuring machines. *Measurement Automation and Monitoring*, 57 (12), 1611-1616.

[15] Weckenmann, A., Knauer, M. (1998). The influence of measurement strategy on the uncertainty of CMM-measurements. *Annals of the CIRP*, 47 (7), 451-455.

[16] Portman, V., Rubenchik, Y., Shuster, V. (2002). Statistical approach to assessments of geometrical accuracy. *Annals of the CIRP*, 51 (1), 463-466.

[17] Cox, M.G., Siebert, B.R.L. (2006). The use of a Monte Carlo method for evaluating uncertainty and expanded uncertainty. *Metrologia*, 43, 178-188.

[18] McKey, M.D., Conover, W.J. (1979). Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21 (2), 239-24.

[19] Helton, J.C., Davis, F.J. (2002). *Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems*. Sandia Report SAND2001-0417, Sandia National Laboratories Albuquerque, California.

[20] Stein, M. (1987). Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29 (2), 143-151.

[21] Owen, A.B. (1998). Latin supercube sampling for very high- dimensional simulations. *ACM Transactions on Modeling and Computer Simulation*, 8 (1), 71-102.

[22] Tabisz, R.A., Nieciąg, H. (2012). Simulation of influence of systematic and random factors on measurement results of coordinate measuring machines. *Measurement Automation and Monitoring*, 58 (12), 1065-1067.

[23] Adamczak, S., Janecki, D., Stępień, K. (2011). Testing of methods of measurement and evaluation of the spherical shape errors of machine parts. *Mechanic*, 12, 958-961.

[24] Domański, Cz., Pruska, K. (2000). *Non-classical Statistical Methods*. Polish Economic Publishing House.

[25] Adamczak, S. (2008). The geometric structure of the surface. Vol. 2. Comparative studies of the instruments. Correlation calculus method. *Mechanic*, 5-6, 514-518.

[26] Wikipedia. *Spearman's rank correlation coefficient*. http://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient.