Optimal Sample Size Selection for Torusity Estimation Using a PSO Based Neural Network

Siwaporn Kunnapapdeelert
School of Manufacturing Systems and Mechanical Engineering, Sirindhorn International Institute of Technology, Thammasat University, Rangsit Campus, Pathum Thani, 12121, Thailand

Chakguy Prakasvudhisarn
School of Technology, Shinawatra University, 15th Floor, Shinawatra Tower III, Chatuchak, Bangkok, 10900 Thailand

Abstract

In a competitive manufacturing environment, the quality, cost, and time to market depend not only on the design and manufacturing but also on the inspection process used. The use of computerized measuring devices has greatly improved the efficacy of geometric tolerance inspection, especially for form measurement. However, they still lack an efficient and effective sampling plan for data collection of a complex form feature like a torus. Factors that could affect plans due to design, manufacturing, and measurement such as size, geometrical tolerance, manufacturing process, and confidence level are studied. Type of manufacturing process, feature size, precision band, and sampling method are identified as impact factors for sampling strategy. A Hammersley sequence based sampling method is extended to cover toroidal shape. A neural network based on the particle swarm optimization (PSO) is then applied to determine sample size for torus feature inspection by taking these impact factors into consideration. The PSO based neural network's algorithm and architecture are described and its predictive ability on unseen test subsets is also presented. An effective and efficient sampling strategy can be achieved by using sampling locations from the Hammersley sampling method and sample size guided by the PSO based neural network.

Keywords: Hammersley sampling methods, torusity, minimum tolerance zone, sample size, neural network, particle swarm optimization

1. Introduction

A basic requirement in manufacturing is to produce a product that meets the specification due to its functional and assembly requirement. Hence, inspection of discrete manufactured parts becomes very critical for conformance of both dimensional and geometrical tolerances. In coordinate metrology, inspection is affected by a variety of data collection and data fitting methods [1]. Many researchers have investigated data fitting issues in geometrical tolerances, especially for basic form tolerances under the assumption that the collected data was a good representative of the inspected shape [2-5]. Data collection thus plays an equally important role as its data fitting counterpart. Theoretically, if the entire surface is measured, all information of that surface can be obtained and the actual error of that part can then be identified. This is very time consuming and hence not suitable for a competitive manufacturing environment. This excessive sample size must be reduced to an acceptable number while maintaining the same high level of accuracy. This undoubtedly can lead to a decrease in the time required for inspection. A sampling plan or strategy considering a minimum sample size and points' locations must be developed to obtain an effective and efficient inspection process. The commonly practiced methods are the uniform sampling, and the simple random sampling methods (SR). Sampling locations based on some mathematical sequences such as Hammersley (HM) and Halton-Zaremba (HZ) have also been studied

with some success [1, 6, 7]. However, the selection of sample size which controls measurement precision is normally conducted by trial and error, experience, and metrology handbooks. This results in a relaxed sample size which gives a trade-off between precision/accuracy of measurement and inspection time. Therefore, a suitable sample size which can represent enough information on the whole population has to be found with a high confidence level.

Interestingly, complex forms such as torusity have been largely ignored and are normally left to be dealt-with by the use of profile tolerance definition, except in a few recent cases [8, 9]. The corresponding equations for toruses are complex. This has partially led to a relative absence of research works dealing with the torusity tolerance in the literature. Since the equations required for torusity calculation have just been found, the study of its data collection issue has not been realized yet. In addition, a sufficient number of industrial parts such as outer and inner races in bearings, and toroidal continuous variable transmission possess toroidal features, must be effectively and efficiently inspected. Considering these many applications, sampling strategies for torusity estimation, especially sample size determination, should be studied more extensively. The need to develop effective and efficient guidelines for sample size used for torusity measurement is the subject of this paper.

The task in doing so is rather complicated because there are many factors involved such as size, dimensional and geometrical tolerances, manufacturing processes, sampling locations, and accuracy and confidence levels. Thus, an analytical approach for their modeling is very difficult due to their unknown nonlinear nature. Feedforward neural networks have been considered a very powerful tool for function approximation and modeling. One of their advantages is the ability to learn from examples. Hence, they can be applied to a model relationship between sample size and its relevant factors. However, their classical training algorithm, back-propagation algorithm (BP), present some disadvantages associated with overfitting, local optimum problems, and sensitivity to the initial values of weights. Particle swarm optimization (PSO), a relatively new population based search technique, demonstrates appealing properties such as simplicity, short computer code, few parameters, fast convergence, consistent results, robustness, and no requirement for gradient information [10]. It can be applied to train neural networks by optimizing their weights in place of the BP. This should relieve some drawbacks posed by the BP algorithm.

The purpose of this work is to propose, for the first time, an efficient and effective method, PSO based neural network, for optimal sample size selection for torusity estimation. To do so, the following steps are investigated; (1) identification of relevant factors influencing sample size of a doughnut-shaped feature inspection; (2) collection of data of these factors and corresponding sample sizes; and (3) development of a PSO based neural network for optimal sample size determination.

2. Literature review

Form tolerance inspection plays a vital role in industrial production since it can guarantee the interchangeability of the parts. Therefore, probe-type coordinate measuring machines (CMMs) have been widely used to accurately measure and analyze parts. However, a main drawback of CMMs is that an entire inspected surface can not practically be measured. CMMs are normally used to measure only a sample of discrete points on the part feature surface and these points are used as a representative of the entire surface. Some other instruments can scan the entire surface but with lower accuracy and precision. Hence, a question follows; how well do the discrete sample points represent the inspected surface?

Dimensional surface measurements have involved the use of deterministic sequences of numbers for determination of sample coordinates to maximize information collected. According to Woo and Liang [6], a two dimensional (2D) sampling strategy based on the Hammersley sequence shows a remarkable improvement of a nearly quadratic reduction in the number of samples when compared with the uniform sampling strategy, while maintaining the same level of accuracy. The HZ based strategy in 2D space was also suggested by Woo et al. [11] without a discernible difference in the performance over the HM strategy. The only differences are that the total number of sample

points in the HZ sequence must be a power of two and the binary representations of the odd bits are inverted. Also, Liang et al. [12, 13] compared the 2D HZ sampling scheme to the uniform scheme and the SR theoretically and experimentally for roughness measurement with similar results. Lee et al. [7] demonstrated a methodology for extending the HM sequence for geometries such as circles, cones, and spheres. Kim and Raman [14] investigated different sampling strategies and different sample sizes for flatness measurements. Their findings were similar to others with regards to accuracy determination. Summerhays et al. [15] proposed new sampling patterns to guide form measurements of internal cylindrical surfaces with some success.

Dowling et al. [16] presented a survey of statistical issues in geometric feature inspection. Fitting and evaluation approaches, sampling design issues, and sources of measurement error were discussed. The incorporation of the knowledge of manufacturing processes was also suggested to improve the accuracy of geometric form inspection. Prakasvudhisarn [1] suggested guidelines for cones and conical frustum inspection by using three sampling sequences, HM, HZ, aligned systematic (AS) with various sample sizes. The sampled points were used to estimate the form error of the feature based on different fitting algorithms.

To help circumvent the adequacy of the data collection problems, Menq et al. [17] suggested a statistical sampling plan to determine a suitable sample size which can represent the entire population of the part surface with sufficient confidence and accuracy. Zhang et al. [18] proposed a feedforward back-propagation neural network approach to estimate sample sizes of holes' measurements from various manufacturing operations. Machining processes, hole diameters, and tolerance bands were considered as influencing factors. Similarly, Lin and Lin [19] developed an algorithm based on the grey theory to predict the number of measuring points on the next workpiece for flatness verification by using data from the last four workpieces. Raghunandan and Rao [20] also reported a method to reduce sample size of flatness estimation by inspecting the first part in detail and using it as the reference for succeeding parts in a batch production.

Conventionally, statistical methods such as multiple regression and partial least squares can be used to determine relationships between inputs and outputs. They normally suffer from assumptions of data distribution. Plus, nonlinear relationships are rather difficult to handle. Without such limitations, neural networks can be alternatively used to model the complex phenomena between factors and outputs of interest in the manufacturing environment. Thus, a neural network would be applied to capture correlation between sample size for torusity inspection and its relevant factors. However, the most widely used algorithm to minimize the sum of squared learning errors, a gradient based BP algorithm, struggles with overfitting, local optimum, and sensitivity to the initial weights. The PSO has been introduced in the framework of an artificial social model that demonstrates appealing properties such as simplicity, short computer code, few parameters, fast convergence, consistency results, robustness, and no requirement for gradient information [10]. Compared to other evolutionary search methods in solving both continuous and discrete optimization problems, the PSO was the second best in terms of processing time while it performed the best in terms of success rate and quality of solutions [21]. In addition, much work has shown the potential of PSO in neural network training while alleviating shortcomings of the BP [22-25].

3. Study of sample size related factors

An artificial Neural Network (ANN) can capture the relationship between input and output by adjusting weights on each link while learning from data. Therefore, selection of data pairs of input and output for training the network is an essential step to ensure sufficiency and integrity of the target function. Determination of suitable sample size is quite complicated since it is affected by various factors such as form fitting criteria, size of the part, type of error on the part's surface, sampling location (position of measured point), and precision band (confidence level on the measurement results). The first factor was not included in this study even though various form-fitting criteria can be used to estimate form tolerances. The most widely used method for form error estimation, the least square method (LSQ), does not guarantee the minimum

tolerance zone defined by ANSI/ASME standard [26]. In other words, it may overestimate the tolerance zone and hence reject some good parts. Therefore, the minimum zone approach, which is consistent with the ANSI standard, was used to evaluate the tolerance zone torusity in this work. The remaining factors are taken into consideration whether or not they really have an impact on sample size. Before considering these factors, some background is discussed in the next two subsections. Effects of these factors are then presented afterward.

3.1 Torus generation

To validate the developed model and also avoid measurement errors from CMMs such as probe orientation, probe angle adjustment, and probe compensation, perfect toruses illustrated in Figure 1 were simulated with details shown in Table 1.

Table 1. Specification of toruses generated.

Area	major radius (c)	Minor radius (a)
2369	10	6
9475	12	20
14804	15	25

Three sizes of perfect toruses were generated by using the following formula:

$$\left(c-\sqrt{x^2+y^2}\right)^2+z^2=a^2 \qquad (1),$$

where c is a major radius of the torus (from the center of hole to the center of the torus tube),
a represents the minor radius of the torus (radius of tube), and
(x, y, z) is the coordinates of the torus' surface.
To imitate the real surface of a manufactured torus-shaped feature, three selected types of error, namely, random pattern, sine coupled with random pattern, and step mixed with random pattern, were each added to the perfect toruses [27]. Three pieces of torus were then generated for each size.
Sine is the sinusoidal oscillation perpendicular to the torus surface as explained below:

$$A\sin(f\hat{u}+p) \qquad (2),$$

where A represents the amplitude of the periodic wave;
f represents frequency;
$\hat{u}=2\pi\left(\frac{i-1}{N-1}\right)$ where N is the total number of simulated points and the position of the point *ith* varies from 1 to N; and
p is the phase angle.
Step is a discontinuity of the radius at each cross section as calculated by:

$$AS(x) \qquad (3),$$

where A is amplitude, and
$S(x)$ is unit step function and $S(x)=\begin{cases}0 & x<0\\ 1 & x\geq 0\end{cases}$.
Random is the random error perpendicular to the torus surface as described below:

$$U(\alpha,\beta) \qquad (4),$$

where $U(\alpha,\beta)$ is a uniformly-distributed random value within the range $[\alpha,\beta]$.

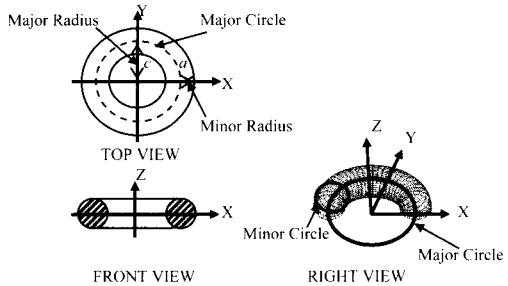

Figure 1. Torus definition.

CMMs provide accuracy of about 0.0001 inch or 2.54 microns for the most basic form, flatness, measurement [28]. A torus, on the other hand, is a complex feature. Their measurement accuracy is hardly the same as that of flatness. Thus, two multiple constants of flatness accuracy were selected to give the tolerance zone errors of 5 and 25 microns for generated toruses. Two groups of nine toruses each were then generated with the controlled torusity of five and twenty five microns. Altogether, there were eighteen toruses included in the experiment.

3.2 Sampling methods

To decrease inspection time while maintaining a high level of accuracy, various sampling techniques such as SR, AS, and mathematical sequence-based, HM and HZ, sampling have been studied. Interestingly, the root mean squares errors of these mathematical sequences are lower than those of commonly practiced procedures, SR, and a type of AS, uniform sampling. Hence, HM, AS, and SR sampling methods were taken into consideration. Even though the performance of HM and HZ based methods are not much different, the HM based method was selected because the number

of sampling points for HZ must be a power of two.

3.2.1 Hammersley based method

HM sequence technique was designed to place N points on the k-dimensional hypercube [11]. In two dimensions, the HM coordinates (x_i, y_i) can be determined as follows:

$$x_i = \frac{i}{N} \quad (5),$$

$$y_i = \sum_{j=0}^{k-1} b_{ij} 2^{-j-1} \quad (6),$$

where N is the total number of sample points,
$i \in I = [0,..., N-1]$,
k is $\lceil \log_2 N \rceil$ = ceiling of $\log_2 N$,
b_i is binary representation of the index i,
b_{ij} denote the j^{th} bit in b_i, and
$j = 0,..., k\text{-}1$.

3.2.2 Aligned systematic sampling method

The systematic sampling sequence is a form of probabilistic sampling which employs a grid of equally spaced location. There are two types of systematic sampling; aligned and unaligned sampling. Aligned sampling is normally called systematic sampling. The sample is first determined by the choice of a pair of random numbers in order to select the coordinates of the upper left unit and the subsequent points are taken according to the predetermined mathematical pattern.

Suppose that a population is suggested in the form of am rows and each row consists of bn units. The basic procedure for arranging the coordinate of aligned systematic sampling can be computed as follows:

1. Determine a pair of random numbers (p,q) where p is less than or equal to m, and q is less than or equal to n. These random numbers would decide the coordinates of the upper left unit by the p^{th} unit column and q^{th} unit row.
2. Locate the subsequent sampling points for x-coordinate as $p + im$ where $i \in [0,..., a-1]$. Therefore, the row consists of $p, p+m, p+2m,..., p+(a-1)m$.
3. Locate the subsequent sampling points for y-coordinate as $q + jn$ where $j \in [0,..., b-1]$. Therefore, the column consists of $q, q+n, q+2n,..., q+(b-1)n$.

3.2.3 Simple random sampling method

Simple random sampling is the sampling procedure that each search element in the population has an equal chance of being selected.

The above sampling strategies are normally described for a 2 dimensional (2D) rectangle but the toroidal feature is a 3D problem. Extension from 2D space to 3D space is required. The concept of torus generation as shown in Figure 2 was applied to all three sampling methods. The generated coordinates via the three sampling methods were transformed from the first picture (upper left corner) to the last picture (lower right corner). Therefore, coordinates generated by each sampling method for toruses would result.

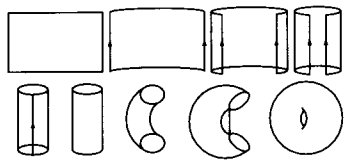

Figure 2. Torus surface generation [29].

Note that the sample size attempted for each transformed 3D sampling method was varied from 8 to 256 points to measure each group of nine toruses. Then, the torusity tolerance zone would be calculated from such points by [8]:

$$d_i = \sqrt{\left[\sqrt{\frac{[(x_i - x_0)v - (y_i - y_0)u]^2 + (x_i - x_0 - z_i u)^2 + (y_i - y_0 - z_i v)^2}{u^2 + v^2 + 1}} - c\right]^2 + z_i^2} - r_0 \quad (7),$$

where d_i is the normal deviation from the measurement (x_i, y_i, z_i) to the ideal torus surface; and x_0, y_0, u, v, c, and r_0 are searched parameters for establishing the ideal torus surface by using the following minimax criterion:

$$\text{minimum zone torusity} = 2 \times \min(\max d_i) \quad (8).$$

Clearly, the torusity zone obtained depends on the measurements (x_i, y_i, z_i) and hence sampling strategies used. Different strategies may give different torusity tolerance zones. Five levels of quantitative precision (precision band) were chosen as 0.3, 0.9, 1.5, 2.1, and 2.7 μm to reflex various sampling strategies employed for each precision.

3.3 Effect of surface area

Intuitively, a larger surface area requires more measurement points than a smaller one to reach the same accuracy and/or precision. To verify such a statement, the impact of changing surface area on the sample size required for torusity inspection was studied by using three different sampling methods, HM, AS, and SR. Each surface had different surface error patterns: random, sine+random, and step+random; with a fixed torusity of 25 microns and precision band of 1.5 microns as shown in Figures 3-5.

Clearly, Figures 3-5 illustrate that if surface area (feature size) of inspected workpiece increases, the number of points required also increases to obtain the same accuracy and precision results. Hence, size or surface area of doughnut-shaped feature must be selected as one of the relevant factors for sample size prediction.

3.4 Effect of surface error patterns

Obviously, the manufacturing process used to produce the part affects its entire surface [20]. Identification of the characteristics of a surface resulting from a manufacturing process can be very complicated and it is very challenging to obtain proven models due to many factors involved such as characteristics of the process, vibration, tool wear, workpiece deformation, and temperature. In the absence of such models, some selected error patterns such as random (noise), sine (systematic), and step (systematic) were added to the surface of those generated perfect toruses to imitate the actual surface of manufactured parts as if they would be produced from different manufacturing processes.

Figures 3-5 show that with the same size of workpiece, torusity value, and precision band, the (sine+random) pattern required the highest number of points. The random pattern required more measurements than the (step+random) pattern for all sampling methods. Therefore, error patterns (or manufacturing processes) must be included as a relevant factor for sample size prediction.

3.5 Effect of sampling locations

As discussed above, the quality of information from measurement depends on the number of points collected and their locations. Three sampling methods, namely, HM, AS, and SR, were taken into consideration as depicted in Figures 6-8. They show that the HM sampling sequence required a smaller sample size than the other two sampling methods; and AS outperformed SR in almost every case. Obviously, different sampling methods extract different information from inspected surfaces. Hence, a sampling method should be selected as an input for sample size estimation as well.

3.6 Effect of precision band

Precision represents the degree of repeatability in a measurement whereas accuracy is the degree which the measured value agrees with the true value. The true value is not possible to obtain due to the time, and hence, cost incurred. Intuitively, if the number of sample points is finite, the measurement result does not converge to a single value, but it does vary within a certain range, the so-called precision band. Obviously, a tighter precision band implies that all measurement results are closer to one another and would give a higher

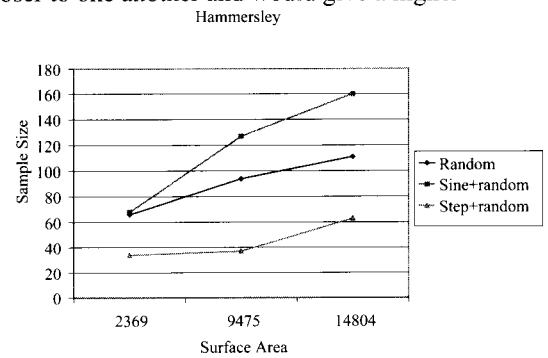

Figure 3. Sample size versus surface area when measured by using HM method with tolerance zone = 25 μm and precision band = 1.5 μm.

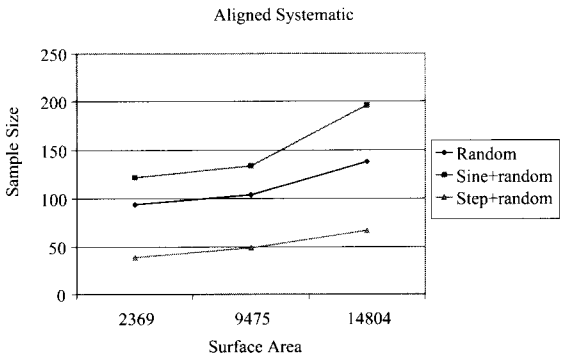

Figure 4. Sample size versus surface area when measured by using AS method with tolerance zone = 25 μm and precision band = 1.5 μm.

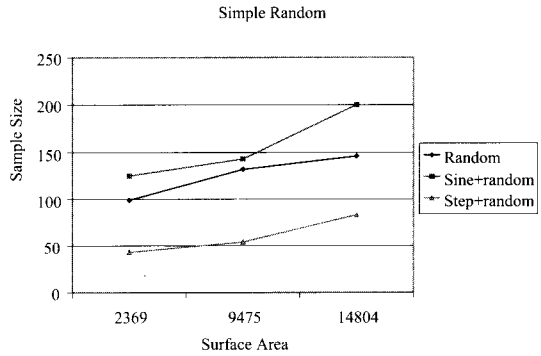

Figure 5. Sample size versus surface area when measured by using SR method with tolerance zone = 25 μm and precision band = 1.5 μm.

confidence level of the outcomes obtained. Figures 6-8 show that the tighter the precision band the greater the sample size and vice versa. Thus, the band of variation should be taken into account as a factor for sample size determination. Note that five experiments for determining suitable number of points for each precision band were conducted to ensure that the sample sizes, an average value of these five experiments, for each precision band were reliable.

In summary, various levels of relevant factors taken into account in this work are depicted in Table 2.

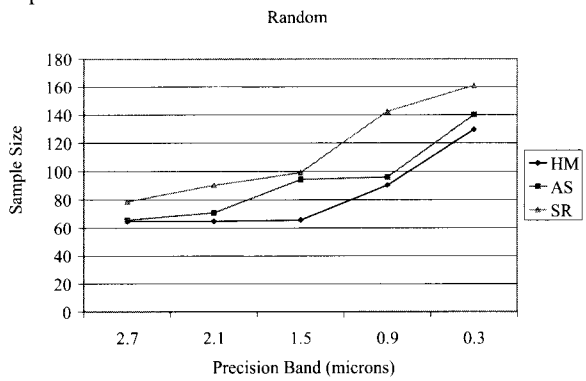

Figure 6. Variations of measurement on random error surface with torusity of 25 μm and surface area of 2369 mm^2.

As clearly seen in this section, the qualitative correlation between sample size and relevant factors such as surface area, surface error pattern, sampling location, and precision band was clearly identified. The next step is to determine the quantitative correlation between them by using artificial neural networks.

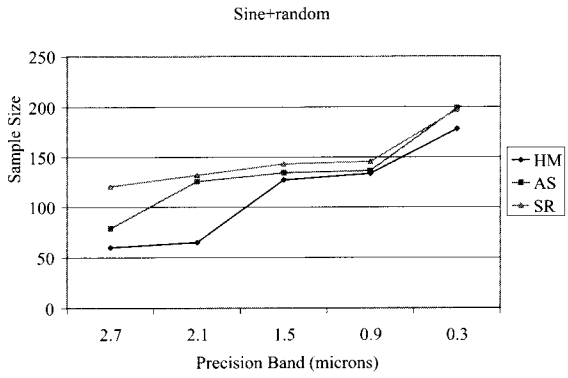

Figure 7. Variations of measurement on sine+random error surface with torusity of 25 μm and surface area of 9475 mm^2.

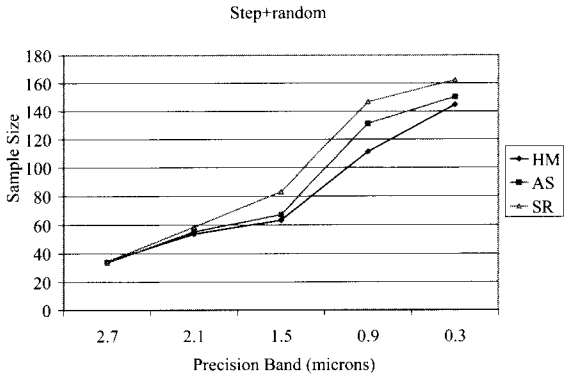

Figure 8. Variations of measurement on step+random error surface with torusity of 25 μm and surface area of 14804 mm^2.

Table 2. Summary of relevant factors for sample size prediction.

Parameters	Details of each parameter
Part dimension (mm^2)	2369 9475 14804
Error pattern	Random Sine + random Step + random
Sampling sequence	Hammersley Aligned systematic Simple random
Precision band (μm)	0.3 0.9 1.5 2.1 2.7

4. Artificial neural networks

The implementation process of feedforward neural networks to model relationships between samples size and its relevant factors can be roughly divided into four main steps, (1) assembling the data, (2) creating the network, (3) training the network, and (4) simulating the network.

In step (1), measurement data were collected from simulated toruses as described in Section 3. Obviously, not all input factors were represented by numerical data. Some were categorical variables such as sampling method and type of surface error. Therefore, these variables must be encoded to numerical numbers between -1 and 1. The three surface error types, namely, random, sine+random, and step+random, were encoded as 0.3, 0.6, and 0.9, respectively. Similarly, sampling strategies were also encoded as 0.3, 0.6, and 0.9 for HM, AS, and SR, respectively. Normally, input parameters of the target function are composed of various magnitudes. The one with higher magnitude may dominate the one with lower magnitude. Therefore, preprocessing should be applied to raw data before training. Thus, the raw data were normalized to [-1, 1] for every factor. Since a large data set of 135 points was collected, the holdout method was chosen as a validation technique for model selection and performance estimation of the constructed model. This data set was thus randomly divided into three subsets for training, validating, and testing. Training the network was performed by using about 70% of the original data (95 data points) whereas the remaining 40 data points was split equally for validating and testing. All of these 135 data are shown in Appendix A.

In step (2), a neural network was created with 4 inputs and 1 output. Four inputs were composed of surface area, error pattern, sampling sequence, and precision band while the only output was the target, sample size. Trial and error was used to determine the network architecture including the number of hidden nodes for each layer and the number of hidden layers by choosing the highest accuracy combined from both training and validating subsets. In one hidden layer architecture, the number of hidden nodes was varied from 4 to 11 to find the best combined accuracies between both subsets. Table 3 illustrates averages percentage of accuracy of different hidden layer nodes conducted for ten runs per structure. As a result, the structure of 8 hidden nodes was selected to alleviate the overfitting problem. Two hidden layers were also attempted and discarded due to high overfitting results and longer computational time.

Table 3. Averages percentage of accuracy of different structures for PSONN.

Number of hidden nodes	% accuracy	
	Training set	Validation set
4	88.58056238	86.06696142
5	90.15994952	87.42250469
6	90.99489026	87.58054576
7	91.02564456	87.65348044
8	**91.1597605**	**87.69065186**
9	91.29307204	87.65102572
10	91.33166735	87.55392991
11	91.51976906	87.4947543

The percentage of accuracy was calculated by

$$\%accuracy = \left(\frac{1 - sum|predicted - actual|}{\sum_{i=1}^{N} actual} \right) \times 100 \quad (9),$$

where N is the total number of data.

The same procedure was also conducted for BPNN. The best generalization performance obtained for both BPNN and PSONN were reached by using 8 hidden neurons. Thus, the architecture of 4-8-1 (four input nodes, eight hidden nodes, and 1 output node) was selected for sample size prediction of torusity verification. Note that the standard *hyperbolic tangent sigmoid* function or *tansig* function was used in the hidden layer to limit its output to small range (-1, 1) whereas the linear *purelin* function was used in the output layer to allow the network output to be a real number.

In step (3), training the network is an attempt to minimize the sum of squared error (difference between actual output and desired output) by adjusting the weights on each link. The bases of feedforward back-propagation neural network and PSO are well documented in the literature and are not repeated here. The weaknesses of the BP such as slow convergence during training, possible divergence for certain conditions, extensive computations (its performance decreases when the size of problem increases), and trap in local minima are alleviated by training the NN with a more

effective and efficient optimization technique like the PSO. Therefore, the PSO was proposed to train the neural network instead of the BP.

4.1 PSO based neural network

The following steps illustrate neural network training by using the PSO:

1. Initialize a population of particles with small random positions, $presentx[i][d]$, and velocities, $v[i][d]$, of the i^{th} particle in the d^{th} dimension on problem space D dimensions (number of weights searched). The position of each particle corresponds to weights in the neural network whereas velocity represents the rate of position change. Also, initialize NN's parameters.
2. Evaluate the desired optimization function, minimization of sum-squared error, in D dimensions for each particle. This is done for every training pair by computing the actual output via analyzing the network from input layer to output layer.
3. Compare evaluation with particle's previous best value, $pbest[i]$. If current value is better than $pbest[i]$, then $pbest[i]$ = current value and $pbest$ position, $pbestx[i][d]$, is set to the current position (or weight).
4. Compare evaluation with swarm's previous best value, $pbest[gbest]$. If current value is better than ($pbest[gbest]$), then $gbest$ = particle's array index.
5. Update velocity and position of each particle by using Equations (10) and (11), respectively:

$$v[i][d] = w \times v[i][d] + C_1 \times rand() \times (pbestx[i][d] - presentx[i][d]) + C_2 \times rand() \times (pbestx[gbest][d] - presentx[i][d]) \quad (10),$$

$$presentx[i][d] = presentx[i][d] + v[i][d] \quad (11).$$

A linearly decreasing inertia weight was implemented by starting at 0.9 and ending at 0.4. This helps expand the search space in the beginning so that the particles can explore new areas, which implies a global search. This statistically shrinks the search space through iterations, which resembles a local search. The acceleration constants C_1 and C_2 represent the weighing of the stochastic terms that pull each particle toward $pbest$ and $gbest$ positions. They are normally set to 2.0 to give it a mean of 1 for the cognition (2^{nd} term) and social parts (3^{rd} term), so that the particles would thoroughly search the settled regions [10]. $rand()$ is a uniformly random number generator within the (0,1) range. This makes the system less predictable and more flexible.

6. Loop to step 2 until a stopping criterion, either a sufficiently good evaluation function value or a maximum number of iterations, is met.

In step (4), both trained networks would then be simulated with all data sets to check their predictive abilities.

5. Results and analyses

The discussed PSONN was implemented in MATLAB 7 running on a Pentium IV 2.4 GHz with Microsoft Windows XP operating system. The computation of PSO depends on a few parameters such as population size, inertia weight, maximum velocity, maximum and minimum positions on each dimension, and maximum number of iterations. Population size and maximum iterations of 20 and 600 were selected, respectively. The inertia weight gradually decreased from 0.9 to 0.4 so as to balance the global and local exploration. Particles' velocities in each dimension were clamped to a maximum velocity, v_{max}, to control the exploration ability of particles. If v_{max} is too high, the PSO facilitates a global search; and particles might pass good solutions. However, if v_{max} is too small, the PSO facilitates a local search; and particles might not explore beyond locally good regions. v_{max} is a problem-oriented parameter and should be set at about 10-20% of the dynamic range of the variable in each dimension. In this experiment, maximum velocity (v_{max}), was set at 12%. The maximum and minimum positions of each variable were chosen to be 0.5 and -0.5, respectively, so that they would give small, around zero, initial weights.

The BPNN was also created by using the neural network toolbox in MATLAB 7 to predict the sample size of torusity verification. Parameter selection in BPNN depends on a few factors, learning rate (η) and momentum (λ). Both are used to control weight adjustment along the gradient direction. The learning rate is used to adjust step size of the weight whereas the momentum factor is used to accelerate convergence of the network. η (learning rate) and λ (momentum factor) were selected as 0.5 and 1, respectively. In addition, the maximum number of epochs was 12000 (number of epochs in BPNN = maximum iterations × size of swarm

in PSONN). All other things for training both networks were kept the same.

PSONN can function very well for prediction of sample size of torusity estimation with the training subsets and just slightly poorer for the validating and unseen test subsets as illustrated in Table 4. Recall that the BP algorithm has some serious limitations associated with overfitting and local optimum problems. The outcomes comparison shows that the PSONN can avoid local optimum trap and reach near-optimal results better than the BPNN can. For both training subsets of torusity errors, their accuracies were much lower than those of PSONN. To avoid overfitting for both PSONN and BPNN, the proper architecture of 8 hidden layer nodes was selected based on the best combined results between training and validating subsets. When applied to both unseen test subsets, their performance showed a slight drop from those of training subsets and the results from PSONN were still much higher than those of BPNN. This shows good generalization performance. It can be concluded that the performance of the trained PSONN is remarkable and consistent for both training samples and testing samples.

Classical BP updates weights and biases based on the gradient descent concept so the solution obtained might get stuck in a local minimum easily without any mechanism to avoid it. Particles in PSO explore new areas in the beginning and refine the search later on, while keeping personal best and group best values. Better solutions can be found based on this concept. In addition, to avoid local traps, some mechanisms such as inertia weight and *rand*() are incorporated in velocity adjustment. This should result in near-optimal performance. Coupled with the model selection technique, the overfitting issue can be avoided to some extent. Consequently, the results obtained confirm that improvement of NN can be accomplished when trained by the PSO.

Table 4. Results of % accuracy comparison between the two training methods.

Torusity error	PSONN		BPNN	
	Training Set	Test Set	Training Set	Test Set
5	91.1597	85.8066	86.2648	81.3906
25	83.9762	79.6038	78.2935	74.2561

Since the test subsets were not used for training, it can be concluded that the neural network can perform well in determining the required sample size for torusity inspection with a certain confidence level. Its performance can be enhanced by utilizing a better training algorithm (optimizer). Data distribution assumptions required in traditional statistical approaches can be discarded in this approach. In addition, more relevant factors, if any, could be included rather easily by retraining the network to obtain a more realistic and comprehensive model with added factors. For inspection of other forms, this PSONN could also be applied, but new training with corresponding data must be conducted first to capture new characteristics of that particular form and its relevant factors. This learning ability, simplicity, and effectiveness are important advantages of the neural network approach.

6. Conclusions and recommendations

In coordinate metrology, an effective and efficient sampling plan for data collection of a given feature is difficult to determine since it can be affected by many factors such as geometric tolerances, manufacturing processes, size, and confidence level on the measurement results. Establishing the correlation between them is the key leading to such a sampling strategy. Experimental studies on torusity produced by different processes (different error types) were carried out to identify key factors that affect the sample size. Surface area, error pattern, sampling methods, and precision band were found as relevant factors. An improved neural network, PSONN, was proposed to model this relationship with significant improvement over the original BPNN due to the appealing properties of PSO such as fast convergence, consistency results, robustness, and local trap avoidance. The results from both training subsets and unseen test subsets demonstrate that the PSONN has the potential for sample size selection of a given form feature measurement, especially when the explicit relationship model is hard to find or does not exist. The PSONN can also be easily expanded to cover more factors so that the correlation model would be more comprehensive and realistic. Moreover, the PSONN could be used to handle other form features by retraining the network with new

corresponding data sets. Therefore, an effective and efficient sampling strategy can be devised by selecting a low discrepancy Hammersley sampling method and a sample size guided by the PSONN. This should give a good representative of the inspected shape for data collection in coordinate metrology.

Just like a black-box, the PSONN still lacks clear interpretability in expressing and explaining relationships between sample size and its relevant factors. This issue should be investigated in the future. In addition, systematic parameters selection of the PSO will certainly enhance its ease of use and should be investigated as well.

7. Acknowledgment

The authors were partially supported by the Thailand Research Fund (TRF) grant MRG4980170.

8. References

[1] Prakasvudhisarn, C., Raman, S., Framework for Cone Feature Measurement Using Coordinate Measuring Machines, J. Manu. Sci. Eng., Vol.126, pp. 169-177, 2004.

[2] Traband, M.T., Joshi, S., Wysk, R.A., and Cavalier, T.M., Evaluation of Straightness and Flatness Tolerances Using the Minimum Zone, Manu. Rev., Vol.2, No.3, pp. 189-195, 1989.

[3] Prakasvudhisarn, C., Trafalis, T.B., and Raman, S., Support Vector Regression for Determination of Minimum Zone, J. Manu. Sci. Eng., Vol.125, pp. 736-739, 2003.

[4] Shunmugam, M.S., On Assessment of Geometric Errors, Int. J. Prod. Res., Vol.24, No.2, pp. 413-425, 1986.

[5] Shunmugam, M.S., Comparison of Linear and Normal Deviations of Forms of Engineering Surfaces, Prec. Eng., Vol.9, No.2, pp. 96-102, 1987.

[6] Woo, T.C. and Liang, R., Dimensional measurement of surfaces and their sampling, Computer Aided Design, Vol.25, No.4, pp. 233-239, 1993.

[7] Lee G., Mou J., and Shen, Y., Sampling Strategy Design for Dimensional Measurement of Geometric Features Using Coordinate Measuring Machine, Int. J. Mach. Tools and Manu., Vol.37, No.7, pp. 917-934, 1997.

[8] Prakasvudhisarn, C., and Kunnapapdeelert, S., Torusity Tolerance Verification Using Swarm Intelligence, Internal Report IE0601, School of Manufacturing Systems and Mechanical Engineering, SIIT, TU, 2006.

[9] Aguirre-Cruz, J.A., and Raman, S., Torus Form Inspection Using Coordinate Sampling, J. Manu. Sci. Eng., Vol.127, pp. 84-95, 2005.

[10] Kennedy, J., and Eberhart, R.C., Particle Swarm Optimization, http://www.engr.iupui.edu/~shi/Coference/psopap4.htmlProc. IEEE Int. Conf. on Neural Networks, IEEE Service Center, Piscataway, NJ, Vol.4, pp. 1942-1948, 1995.

[11] Woo, T.C., Liang, R., Hsieh, C.C., and Lee, N.K., Efficient Sampling for Surface Measurements, J. Manu. Sys., Vol.14, No.5, pp. 345-354, 1995.

[12] Liang, R., Woo, T.C., and Hsieh, C.C., Accuracy and Time in Surface Measurement, Part 1: Mathematical Foundations, J. Manu. Sci. Eng., Vol.120, No.1, pp. 141-149, 1998a.

[13] Liang, R., Woo, T.C., and Hsieh, C.C., Accuracy and Time in Surface Measurement, Part 2: Optimal Sampling Sequence, J. Manu. Sci. Eng, Vol.120, No.1, pp. 150-155, 1998b.

[14] Kim, W.S. and Raman, S., On the Selection of Flatness Measurement Points in Coordinate Measuring Machine Inspection, Int. J. Mach. Tools Manu., Vol.40, No.3, pp. 427-443, 2000.

[15] Summerhays K.D., Henke R.P., Baldwin J.M., Cassou R.M., Brown C.W., Optimization Discrete Point Sample Patterns and Measurement Data Analysis on Internal Cylindrical Surfaces with Systematic Form Deviations, Prec. Eng., Vol.26, pp. 105-121, 2002.

[16] Dowling, M.M., Griffin, P.M., Tsui, K.L., and Zhou, C., Statistical Issues in Geometric Feature Inspection Using Coordinate Measuring Machines, Technometrics, Vol.39, No.1, pp. 3-17, 1997.

[17] Menq, C.H., Yau, H.T., Lai, G.Y., and Miller, R.A., Statistical Evaluation of Form Tolerances Using Discrete Measurement Data, American Society of Mechanical Engineers, Production Engineering

Division (Publication) PED, 47, pp. 135-149, 1990.

[18] Zhang Y.F., Nee A.Y.C., Fuh J.Y.H., Neo K.S., and Loy H.K., A Neural Network Approach to Determining Optimal Inspection Sampling Size for CMM, Computer-Integrated Manufacturing Systems, Vol.9, No.3, pp. 161-169, 1996.

[19] Lin Z.C., and Lin W.S., Measurement Point Prediction of Flatness Geometric Tolerance by Using Grey Theory, Prec. Eng., Vol.25, No.3, pp. 171-184, 2001.

[20] Raghunandan, R., and Rao, P.V., Selection of An Optimum Sample Size for Flatness Error Estimation While Using Coordinate Measuring Machine, Int. J. Mach. Tools Manu., Vol.47, No.3-4, pp. 477-482, 2007.

[21] Elbeltagi, E., Hegazy, T., and Grierson, D., Comparison Among Five Evolutionary-based Optimization Algorithms, Advanced Engineering Informatics, Vol.19, pp.43-53, 2005.

[22] Clow, B., and White, T., An Evolutionary Race: A Comparison of Genetic Algorithms and Particle Swarm Optimization Used for Training Neural Networks, Proc. of the Int. Conf. on Artificial Intelligence, IC-AI'04, 2, pp. 582-588, 2004.

[23] Zhang, C., Shao, H., and Li, Y., Particle Swarm Optimisation for Evolving Artificial Neural Network, Proc. IEEE Int. Conf. on Sys., Man, and Cyb., pp. 2487-2490, 2000.

[24] Mendes, R., Cortez, P., Rocha, M., and Neves, J., Particle Swarms for Feedforward Neural Network Training, Proc. Int. Conf. on Neural Networks (IJCNN 2002), pp. 1895-1899, 2002.

[25] Gudise, V. G. and Venayagamoorthy, G. K., Comparison of Particle Swarm Optimization and Backpropagation as Training Algorithms for Neural Networks, Proc. IEEE Swarm Intelligence Symposium 2003 (SIS 2003), Indianapolis, Indiana, USA, pp. 110-117, 2003.

[26] ASME Y14.5M-1994, Dimensioning and Tolerancing, The American Society of Mechanical Engineers, New York, 1995.

[27] Algeo, M.E. and Hopp, T.H., Form Error Models of the NIST Algorithm Testing System, NISTIR 4740, National Institute of Standards and Technology, Gaithersburg, MD, 1992.

[28] http://en.wikipedia.org/wiki/Lapping, June 2006.

[29] http://abyss.uoregon.edu/~js/glossary/torus.html, August 2006.

Appendix A

Surface area (mm^2)	Sampling strategy	Error pattern	Precision band (μm)	Sample size
2369	Hammersley	random	2.7	58
2369	Hammersley	random	2.1	84
2369	Hammersley	random	1.5	156
2369	Hammersley	random	0.9	156
2369	Hammersley	random	0.3	184
2369	Aligned Systematic	random	2.7	50
2369	Aligned Systematic	random	2.1	140
2369	Aligned Systematic	random	1.5	140
2369	Aligned Systematic	random	0.9	144
2369	Aligned Systematic	random	0.3	184
2369	Simple Random	random	2.7	64
2369	Simple Random	random	2.1	96
2369	Simple Random	random	1.5	116
2369	Simple Random	random	0.9	116
2369	Simple Random	random	0.3	184
9475	Hammersley	random	2.7	62
9475	Hammersley	random	2.1	92
9475	Hammersley	random	1.5	116
9475	Hammersley	random	0.9	164
9475	Hammersley	random	0.3	200
9475	Aligned Systematic	random	2.7	52
9475	Aligned Systematic	random	2.1	140
9475	Aligned Systematic	random	1.5	144
9475	Aligned Systematic	random	0.9	164

Surface area (mm^2)	Sampling strategy	Error pattern	Precision band (μm)	Sample size
9475	Aligned Systematic	random	0.3	200
9475	Simple Random	random	2.7	80
9475	Simple Random	random	2.1	98
9475	Simple Random	random	1.5	128
9475	Simple Random	random	0.9	132
9475	Simple Random	random	0.3	196
14804	Hammersley	random	2.7	76
14804	Hammersley	random	2.1	108
14804	Hammersley	random	1.5	180
14804	Hammersley	random	0.9	180
14804	Hammersley	random	0.3	200
14804	Aligned Systematic	random	2.7	52
14804	Aligned Systematic	random	2.1	148
14804	Aligned Systematic	random	1.5	196
14804	Aligned Systematic	random	0.9	196
14804	Aligned Systematic	random	0.3	200
14804	Simple Random	random	2.7	92
14804	Simple Random	random	2.1	132
14804	Simple Random	random	1.5	148
14804	Simple Random	random	0.9	148
14804	Simple Random	random	0.3	200
2369	Hammersley	sine+random	2.7	76
2369	Hammersley	sine+random	2.1	96
2369	Hammersley	sine+random	1.5	120
2369	Hammersley	sine+random	0.9	120
2369	Hammersley	sine+random	0.3	132
2369	Aligned Systematic	sine+random	2.7	64
2369	Aligned Systematic	sine+random	2.1	84
2369	Aligned Systematic	sine+random	1.5	116
2369	Aligned Systematic	sine+random	0.9	116
2369	Aligned Systematic	sine+random	0.3	160
2369	Simple Random	sine+random	2.7	72
2369	Simple Random	sine+random	2.1	72
2369	Simple Random	sine+random	1.5	84
2369	Simple Random	sine+random	0.9	84
2369	Simple Random	sine+random	0.3	116
9475	Hammersley	sine+random	2.7	88
9475	Hammersley	sine+random	2.1	120
9475	Hammersley	sine+random	1.5	128
9475	Hammersley	sine+random	0.9	128
9475	Hammersley	sine+random	0.3	180
9475	Aligned Systematic	sine+random	2.7	104
9475	Aligned Systematic	sine+random	2.1	128
9475	Aligned Systematic	sine+random	1.5	128
9475	Aligned Systematic	sine+random	0.9	128
9475	Aligned Systematic	sine+random	0.3	170
9475	Simple Random	sine+random	2.7	84
9475	Simple Random	sine+random	2.1	84
9475	Simple Random	sine+random	1.5	120
9475	Simple Random	sine+random	0.9	120
9475	Simple Random	sine+random	0.3	120
14804	Hammersley	sine+random	2.7	92
14804	Hammersley	sine+random	2.1	124
14804	Hammersley	sine+random	1.5	144
14804	Hammersley	sine+random	0.9	144
14804	Hammersley	sine+random	0.3	184
14804	Aligned Systematic	sine+random	2.7	104
14804	Aligned Systematic	sine+random	2.1	164
14804	Aligned Systematic	sine+random	1.5	164
14804	Aligned Systematic	sine+random	0.9	164

Surface area (mm^2)	Sampling strategy	Error pattern	Precision band (μm)	Sample size
14804	Aligned Systematic	sine+random	0.3	180
14804	Simple Random	sine+random	2.7	104
14804	Simple Random	sine+random	2.1	132
14804	Simple Random	sine+random	1.5	132
14804	Simple Random	sine+random	0.9	132
14804	Simple Random	sine+random	0.3	132
2369	Hammersley	step+random	2.7	56
2369	Hammersley	step+random	2.1	56
2369	Hammersley	step+random	1.5	56
2369	Hammersley	step+random	0.9	56
2369	Hammersley	step+random	0.3	88
2369	Aligned Systematic	step+random	2.7	64
2369	Aligned Systematic	step+random	2.1	64
2369	Aligned Systematic	step+random	1.5	64
2369	Aligned Systematic	step+random	0.9	64
2369	Aligned Systematic	step+random	0.3	76
2369	Simple Random	step+random	2.7	60
2369	Simple Random	step+random	2.1	60
2369	Simple Random	step+random	1.5	60
2369	Simple Random	step+random	0.9	60
2369	Simple Random	step+random	0.3	84
9475	Hammersley	step+random	2.7	64
9475	Hammersley	step+random	2.1	64
9475	Hammersley	step+random	1.5	64
9475	Hammersley	step+random	0.9	64
9475	Hammersley	step+random	0.3	92
9475	Aligned Systematic	step+random	2.7	76
9475	Aligned Systematic	step+random	2.1	76
9475	Aligned Systematic	step+random	1.5	76
9475	Aligned Systematic	step+random	0.9	76
9475	Aligned Systematic	step+random	0.3	80
9475	Simple Random	step+random	2.7	76
9475	Simple Random	step+random	2.1	76
9475	Simple Random	step+random	1.5	76
9475	Simple Random	step+random	0.9	76
9475	Simple Random	step+random	0.3	92
14804	Hammersley	step+random	2.7	72
14804	Hammersley	step+random	2.1	72
14804	Hammersley	step+random	1.5	72
14804	Hammersley	step+random	0.9	72
14804	Hammersley	step+random	0.3	108
14804	Aligned Systematic	step+random	2.7	84
14804	Aligned Systematic	step+random	2.1	84
14804	Aligned Systematic	step+random	1.5	84
14804	Aligned Systematic	step+random	0.9	84
14804	Aligned Systematic	step+random	0.3	120
14804	Simple Random	step+random	2.7	80
14804	Simple Random	step+random	2.1	80
14804	Simple Random	step+random	1.5	80
14804	Simple Random	step+random	0.9	80
14804	Simple Random	step+random	0.3	100