IEMS Vol. 6, No. 2, pp. 94-105, December 2007.

Torusity Tolerance Verification using Swarm Intelligence

Chakguy Prakasvudhisarn †
Industrial Engineering Program, Sirindhorn International Institute of Technology
Thammasat University, Pathumthani, 12121, THAILAND
+662-986-9009 Ext. 2108, Email: chakguy@siit.tu.ac.th

Siwaporn Kunnapapdeelert
Industrial Engineering Program, Sirindhorn International Institute of Technology
Thammasat University, Pathumthani, 12121, THAILAND
+662-986-9009 Ext. 2108, Email: chakguy@siit.tu.ac.th

Received Date, October 2006; Accepted Date, April 2007

Abstract. Measurement technology plays an important role in discrete manufacturing industry. Probe-type coordinate measuring machines (CMMs) are normally used to capture the geometry of part features. The measured points are then fit to verify a specified geometry by using the least squares method (LSQ). However, it occasionally overestimates the tolerance zone, which leads to the rejection of some good parts. To overcome this drawback, minimum zone approaches defined by the ANSI Y14.5M-1994 standard have been extensively pursued for zone fitting in coordinate form literature for such basic features as plane, circle, cylinder and sphere. Meanwhile, complex features such as torus have been left to be dealt-with by the use of profile tolerance definition. This may be impractical when accuracy of the whole profile is desired. Hence, the true deviation model of torus is developed and then formulated as a minimax problem. Next, a relatively new and simple population based evolutionary approach, particle swarm optimization (PSO), is applied by imitating the social behavior of animals to find the minimum tolerance zone torusity. Simulated data with specified torusity zones are used to validate the deviation model. The torusity results are in close agreement with the actual torusity zones and also confirm the effectiveness of the proposed PSO when compared to those of the LSQ.

Keywords: Torusity, Tolerance Verification, Coordinate Metrology, Minimum Zone Estimation, Particle Swarm Optimization.

1. INTRODUCTION

Coordinate measuring machines (CMMs) are an extremely powerful metrological instrument. Coupled with the aid of computer and CMM software, they can automatically perform complex analysis to verify manufactured parts' conformance to size and geometric tolerances such as form, orientation, and runout. The most widely used technique for form tolerances analyses in practice is the least squares method (LSQ) due to its simplicity and robustness (Traband *et al.*, 1989; Shunmugam, 1987; Prakasvudhisarn *et al.*, 2003). Also, it can be applied to most geometries quite easily provided that their respective discrepancy models are known. However, the results obtained by the LSQ do not guarantee the minimum zones as specified by the ANSI standard (ASME Y14.5M-1994, 1995). It occasionally overestimates the tolerance zone. Consequently, this leads to the economical disadvantages of rejecting or reworking some good parts. Therefore, minimum zone approach has been pursued for zone fitting instead.

Although all basic form tolerances such as straightness, flatness, circularity, cylindricity, and sphericity have been defined and investigated based on minimum zone approach in coordinate form literature, interestingly the form tolerances for torus and other complex shapes have been largely ignored due to the lack of their geometrical deviation models. They are normally left to be dealt-with by the use of profile tolerance definition. Profiles such as straight lines, arcs, and other curved lines may be applied and the tolerance estimations of these elements are verified individually. Such a procedure may be impractical in case where accuracy of the

† : Corresponding Author

entire profile is a requirement. This implies that the common practice, the use of profile, for torus is far from optimal in spite of the sufficient need to inspect them in parts such as outer and inner races in bearings and toroidal continuous variable transmission. Hence, to inspect doughnut-shaped feature, a set of mathematical models for torusity error verification consisting of the true deviation model and its nonlinear optimization counterpart is clearly desired.

The true nonlinear deviation model of torus-shaped must first be developed and then used to establish the ideal feature of torus from actual measurements. The main idea of this step is similar to the procedure to obtain the discrepancy model in the orthogonal least-squares regression. Based on minimum tolerance zone approach, this ideal torus is determined to split the measured data into two parts equidistantly, inside and outside the assessment torus. In other words, it is utilized to set up two imaginary tori, the outer and the inner tori respect to the ideal, to form a tolerance zone torusity. This is where the minimax criterion comes into play. That is, the chosen ideal torus minimizes the maximum distance that an individual point falls from the ideal. Therefore, the normal discrepancy model and the minimax criterion are combined to verify form tolerance of the toroidal object.

Besides the need for the set of mathematical models, an effective and efficient optimization technique should be taken into consideration to solve the formulated mo-del for minimum zone torusity estimation. A relatively new algorithm, the particle swarm optimization (PSO), has been introduced in the framework of an artificial social model. It is a population based stochastic optimization method that demonstrates appealing properties such as simplicity, short computer code, fast convergence, consistency results, robustness, and no requirement for gradient information (Kennedy and Eberhart, 1995). Recently, the PSO has been successfully applied to solve a wide range of applications (Allahverdi and Al-Anzi, 2006; Liu *et al*., 2006; Lawtrakul and Prakasvudhisarn, 2005; Chuanwen and Bompard, 2005; Elbeltagi *et al*., 2005). However, its application to form errors evaluation has not yet been fully realized. Hence, the PSO is selected to verify the form conformance of the manufactured toroidal object by optimizing the developed minimax function.

Therefore, to effectively and efficiently inspect doughnut-shaped features, this research attempts to 1) determine the *first* true nonlinear torusity model through an integrative investigation of torusity definition, its deviation model, and its optimization formulation and 2) investigate an application of the PSO for torusity form error evaluation.

2. LITERATURE REVIEW

Various techniques have been discussed for various form tolerances evaluation based on the minimum zone concept. They can be roughly classified into two categories, computational geometry approach and numerical approach. The former approach deals with algorithms and data structures (Traband *et al*., 1989; Hong *et al*., 1991; Le and Lee, 1991; Roy and Zhang, 1992, 1994; Roy, 1995). The information of the problems is organized in such a way that would permit the algorithms to run in the most effective manner since they exploit the problems' structures. However, each algorithm is limited to a particular form tolerance and difficultly expanded to cover other forms, especially when they possess totally different geometrical characteristics such as linearity nature of line or plane and nonlinearity nature of circle or cylinder. Some computational geometry based methods are convex hull, eigenpolygon, and Voronoi diagram. The latter approach consists of using linear and nonlinear optimization methods such as simplex search, genetic algorithms (GAs), and simulated annealing (SA) with the corresponding deviation model of each form feature (Shunmugam, 1987; Dhanish and Shunmugam, 1991; Wang, 1992; Kanada and Suzuki, 1993; Carr and Ferreira, 1995a, 1995b; Lai *et al*., 2000; Sharma *et al*., 2000; Liu *et al*., 2001; Hong *et al*., 2001; Wen and Song, 2004). That is, the formulated optimization model remains the same for every form feature since it is based on the same criterion for tolerance zone evaluation. For example, the minimax criterion is one of a few variants applied to every basic form with the respective discrepancy model. Hence, this approach is quite flexible because it can simply be extended to cover various form tolerances if those forms' deviation models are available. Its computational speed is rather fast even though it is not as computationally efficient as that of the computational geometry based approach.

Traband *et al*. (1989) proposed a methodology based on the convex hull principle to evaluate form tolerances. Straightness and flatness were evaluated by adopting 2 dimensional (2D) and 3 dimensional (3D) convex hulls, respectively. Hong *et al*. (1991) verified minimum zone straightness by using the concept of geometrical eigen-polygon (EPG). Le and Lee (1991) introduced a standard called the minimum area difference (MAD) center to measure the tolerance of circular profile. The MAD center relied on the information of the farthest and nearest neighbor Voronoi diagrams and the convex hull concept. Roy and Zhang (1992, 1994) proposed a hybrid assessment technique combining convex hull and Voronoi diagrams for determining the roundness error. Roy (1995) addressed the concepts of the tolerance zones (TZs) and minimum zones (MZs) for evaluating form and positional tolerances. Minimum zone straightness and roundness were calculated based on 2D convex hull whereas flatness and cylindricity were estimated based on 3D convex hull.

In the numerical approach, before the optimization model can be formulated, the relationship function of relevant form parameters must geometrically be determined. Then, various techniques can be used to search the formulated function for optimal solutions of basic form features such as straightness, flatness, circularity, cylindricity, and sphericity. Shunmugam (1987) compared the linear and normal deviations of form tolerances using the LSQ and minimum deviation methods. A simplex search algorithm was used in a search procedure for both approaches. Dhanish and Shunmugam (1991) presented an algorithm based on the theory of discrete and linear Chebyshev approximation to evaluate the form errors. Wang (1992) proposed a nonlinear optimization method for minimum zone evaluation of common form features. The form errors were conceptually determined by minimizing the maximum deviation of the sampled points, which led to minimax problems. They were reformulated into nonlinearly constrained problems by introducing an additional variable. Some simple mechanisms were suggested to improve the stability of the algorithm. Other nonlinear optimization techniques such as downhill simplex and repetitive bracketing methods were also used to evaluate flatness (Kanada and Suzuki, 1993). Carr and Ferreira (1995a) developed a single verification methodology that can be applied to the cylindricity and straightness of median line problems by using a successive linear programming, which was an extension of their straightness and flatness evaluations (Carr and Ferreira, 1995b). The form tolerances for complex shapes such as cone and torus were also investigated. Prakasvudhisarn and Raman (2004) developed the linear and nonlinear deviation models for conicity and used the LSQ and the generalized reduced gradient algorithm to find the conicity zones. Subsequently, the linear deviation models of torus were proposed and fitted by using the LSQ for torusity evaluation (Aguirre-Cruz and Raman, 2005).

Some heuristic techniques such as GA and SA have also been applied to verify form tolerances with good results. GAs were chosen to evaluate the cylindricity (Lai *et al*., 2000) and also other basic form tolerances (Sharma *et al*., 2000). Liu *et al*. (2001) proposed a hybrid approach between GA and geometric characterization method for assessing tolerance specifications of straightness and flatness. Another hybridization of computational geometry and SA was used to evaluate straightness and flatness (Hong *et al*., 2001). Prakasvudhisarn *et al*. (2003) adopted an approach based on the support vector machine algorithm, v support vector regression (v-SVR), to estimate the minimum enclosing zones straightness and flatness. The presented algorithm attempted to minimize the ε-insensitive tube which was modeled as the tolerance zone of the inspected linear form features. An algorithm inspired by the immune system and the evolutionary biology was proposed to evaluate sphericity error (Wen and Song, 2004). Another evolutionary method, the PSO, has gained more interest due to its appealing performance in many applications in term of convergence rate, consistency results, and differentiable requirement of the evaluation function.

The PSO has been introduced in the framework of an artificial social model to evaluate continuous nonlinear functions (Eberhart and Kennedy, 1995). It is based on a very simple concept of bird flocking, fish schooling, and swarming theory, exhibiting some popula-tion based stochastic evolutionary computation. Each particle flies over the solution space and adjusts its trajectory toward its current velocity, own experience, and swarm's experience. To avoid local optima, randomness is also incorporated into the computation of a new velocity. Furthermore, this philosophical framework can be implemented in a few lines of computer code. The PSO has been applied to solve a wide range of applications. It was used to evolve artificial neural networks for human tremor analysis (Eberhart and Hu, 1999). It was expanded to handle a mixed-integer nonlinear optimization problem for reactive power and voltage control considering voltage security assessment (Yoshida *et al*., 1999). Furthermore, the PSO was applied to solve a set of well-known test minimax problems with promising results (Laskari *et al*., 2002). To assure the PSO's performance, five evolutionary-based optimization algorithms such as genetic algorithm (GA), memetic algorithms (MA), particle swarm (PSO), ant-colony optimization (ACO), and shuffled frog leaping (SFL) were tested with both continuous and discrete optimization problems and then compared in terms of processing time, success rate, and solution quality. The results showed that the PSO algorithm was the second best in terms of processing time while performed the best in terms of success rate and quality of solutions (Elbeltagi *et al*., 2005).

In summary, the inspection of toroidal object has rarely truly been investigated due to its complexity and hence needs to be tackled to improve its effectiveness and efficiency. Therefore, the purpose of this study is to develop the currently nonexistent set of true nonlinear mathematical model for the doughnut-shaped fitting with minimax criterion and to apply the promising algorithm like the PSO to torusity determination.

3. TORUS FORM TOLERANCE DEFINITION AND DEVIATION

Tolerances are the total amount from which a specified dimension is permitted to vary. This concept is applied not only to size tolerance but also to geometric tolerances such as location, orientation, runout, and form. Form tolerances are most frequently applied to single feature or portion of a feature. To evaluate form feature, an ideal feature is established from the actual measurements while simultaneously constructing a minimum tolerance zone within which all measurement values must lie. The obtained zone or the deviations of the feature from the ideal must be within the specified tolerance. In other words, form tolerances state how far the actual features are permitted to vary from the designed nominal form. Common types of form tolerances such as straightness, flatness, circularity, sphericity, and cylindricity are defined by the ANSI standard based on the minimum zone concept. For instance, cylindricity is the entire feature surface during one revolution in which all points are on equal distance from a common axis as shown in Figure 1. This section only presents the derivation process of the torusity deviation model. Its minimum tolerance zone formulation is discussed in the subsequent section.

A torus is formed by rotating a circle, minor circle, about a line that is in the plane of the circle, but not intersecting the circle as illustrated in Figure 2. All center points of all revolved circles form a common core circle, major circle, of torus. Torusity is then defined as the entire feature surface during one revolution in which all points are on equal distance from the center points on the major circle. Thus, torusity can be determined by calculating the normal distances between measurement points and surface of the assessment torus as illustrated in Figure 2 to Figure 5.

In Figure 3, given that $L = (x_i - x_0, y_i - y_0, z_i - z_0)$ is the vector between point $O(x_0, y_0, z_0)$ and a measurement point r_i, hence,

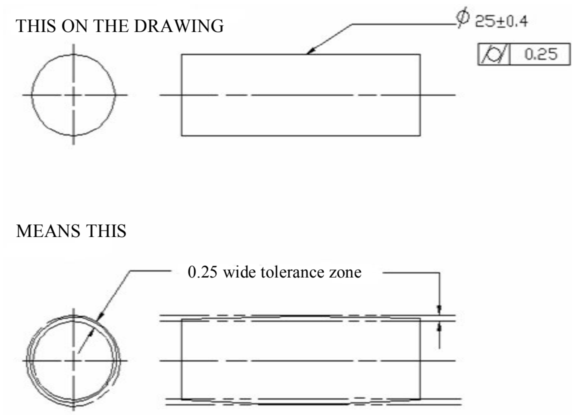

Figure 1. Specifying cylindricity of surface elements (ASME Y14.5M-1994, 1995)

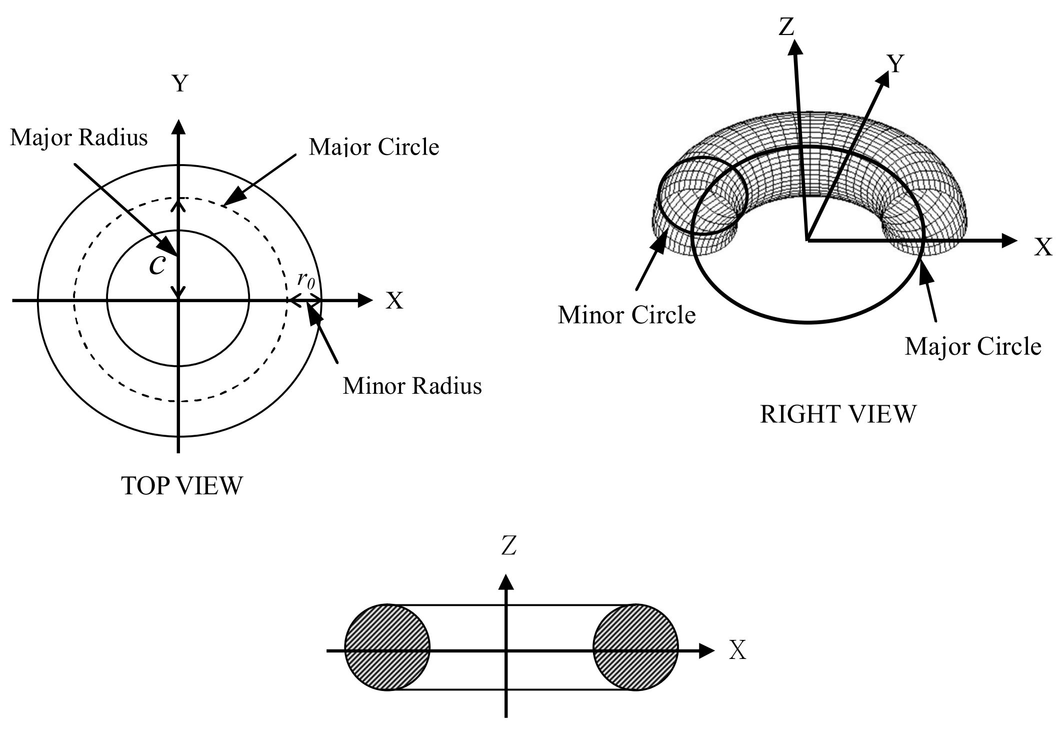

Figure 2. Torus definition

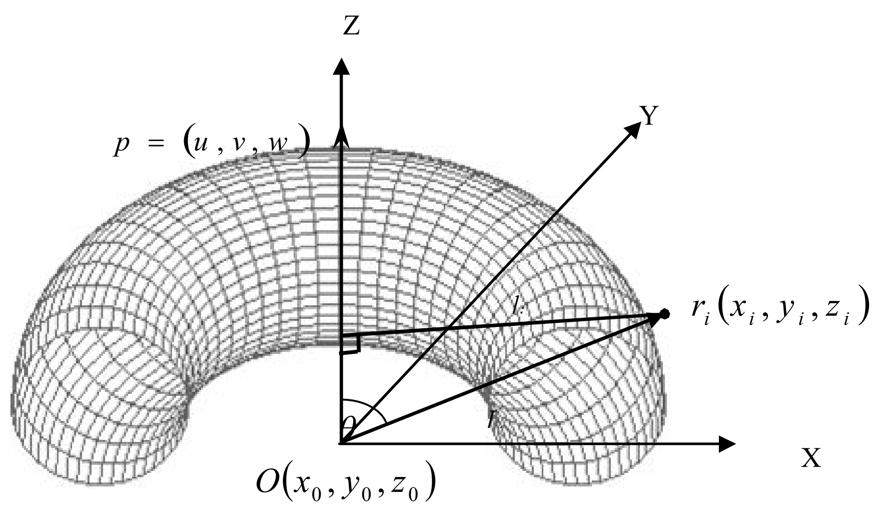

Figure 3. Assessment of torusity error showing calculation of the distance l

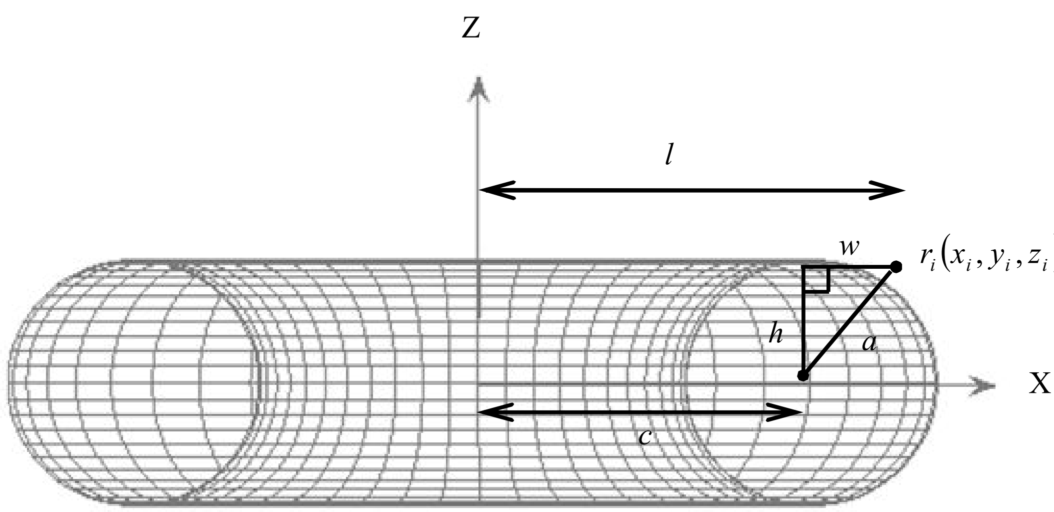

Figure 4. Assessment of torusity error showing minor radius calculation

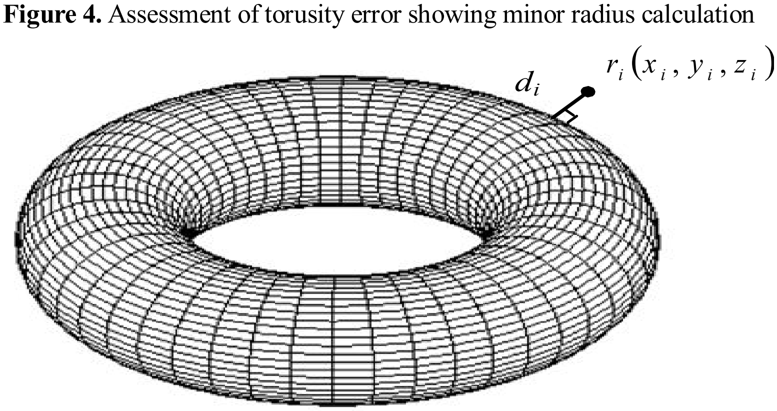

Figure 5. The orthogonal distance from a measurement value $r_i(x_i, y_i, z_i)$ to the surface of the assessment torus

$$l^2 = \|L\|^2 - \left(\frac{L \cdot p}{\|p\|}\right)^2 \quad \text{and}$$

$$l^2 = \left[(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2\right] - \left(\frac{1}{\sqrt{u^2+v^2+w^2}}\right)^2 \left[(x_i - x_0)u + (y_i - y_0)v + (z_i - z_0)w\right]^2 \tag{1}$$

$$l^2 = \left(\frac{1}{u^2+v^2+w^2}\right)\begin{Bmatrix} \left[(x_i - x_0)v - (y_i - y_0)u\right]^2 \\ + \left[(x_i - x_0)w - (z_i - z_0)u\right]^2 \\ + \left[(y_i - y_0)w - (z_i - z_0)v\right]^2 \end{Bmatrix} \tag{2}$$

$$l = \sqrt{\frac{\begin{Bmatrix} \left[(x_i - x_0)v - (y_i - y_0)u\right]^2 + \left[(x_i - x_0)w - (z_i - z_0)u\right]^2 \\ + \left[(y_i - y_0)w - (z_i - z_0)v\right]^2 \end{Bmatrix}}{u^2+v^2+w^2}} \tag{3}$$

where r_i is the coordinates of a sampled point; $O(x_0, y_0, z_0)$ is the origin of a local frame (about the center of torus); (u, v, w) is a normalized direction vector of a torus and its axes. In Figure 4, let a represent the normal distance from a measurement point to a center point of a minor circle (torus tube); c be the distance between center points of major and minor circles; and h or $(z_i - z_0)$ be the linear distance along z-axis between a measured point and xy-plane of torus' base. Thus, $a^2 = w^2 + h^2$ and $w = l - c$. Consequently, $a^2 = (l-c)^2 + h^2$ and substitute l from Equation (3), then

$$a^2 = \left[\sqrt{\frac{\begin{Bmatrix} [(x_i - x_0)v - (y_i - y_0)u]^2 + [(x_i - x_0)w - (z_i - z_0)u]^2 \\ + [(y_i - y_0)w - (z_i - z_0)v]^2 \end{Bmatrix}}{u^2+v^2+w^2}} - c\right]^2 + (z_i - z_0)^2 \tag{4}$$

Therefore, an error which is a normal distance from a measurement point $r_i(x_i, y_i, z_i)$ to the surface of the fitted ideal torus is $d_i = a - r_0$ as illustrated in Figure 2 to Figure 5. Thus,

$$d_i = \sqrt{\left[\sqrt{\frac{\begin{Bmatrix} [(x_i - x_0)v - (y_i - y_0)u]^2 + [(x_i - x_0)w - (z_i - z_0)u]^2 \\ + [(y_i - y_0)w - (z_i - z_0)v]^2 \end{Bmatrix}}{u^2+v^2+w^2}} - c\right]^2 + (z_i - z_0)^2} - r_0 \tag{5}$$

where r_0 is the minor radius of the minor circle of the ideal torus.

Without loss of generality, z_0 and w are simplified by equating to 0 and 1, respectively. Then, the normal distance d_i becomes

$$d_i = \sqrt{\left[\sqrt{\frac{[(x_i - x_0)v - (y_i - y_0)u]^2 + (x_i - x_0 - z_i u)^2 + (y_i - y_0 - z_i v)^2}{u^2+v^2+1}} - c\right]^2 + z_i^2} - r_0 \tag{6}$$

The ideal torus can be established by determining values of these relevant parameters, x_0, y_0, u, v, c , and r_0, of the derived deviation model above by using a criterion for best fit. In this case, the minimax criterion was selected since it conformed to minimum tolerance zone approach set by the ANSI standard (ASME Y14.5M-1994, 1995). This is a major difference why the LSQ which is based on a different criterion, minimization of the sum of the squares of the errors, tends to overestimate the computed tolerance zone.

4. FORM TOLERANCES FORMULATION

A form tolerance can be established by searching for the ideal form of the inspected feature from collected data points while simultaneously minimizing the maximum deviation between measurement values and the fitted form (Wang, 1992). As a result, the upper and lower limits that contain all measured points would provide a geometrical tolerance zone. To follow this criterion, a minimax problem can be formulated. For instance, the basis for minimum zone straightness is described as follows:

minimum zone = 2* min (max d_i)
or h = min (max d_i) (7)

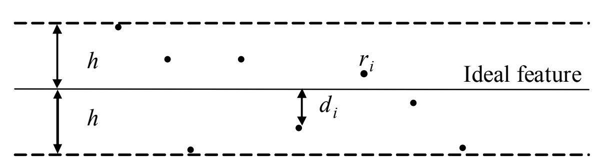

Figure 6. Minimum zone straightness

where d_i is a normal distance from a measurement point to the ideal feature. Figure 6 illustrates such zone; where dots represent measured data points i or r_i, h is a half width of zone, and d_i is the deviation of point i from the ideal feature. The minimax formulation would result in an ideal feature having equi-distance to the farthest data points on both sides of this ideal. The minimum zone obtained is then compared with the specified tolerance limit for conformance. The above formulation model can also be applied to cover other form features by using a proper deviation model for each particular form.

In other words, the discrepancy model derived for toroidal object should be combined with this minimax criterion and Equation (6) was then used as an error model, d_i, in Equation (7) for torusity verification.

Normally, sequential quadratic programming is a common gradient-based approach for solving minimax problems. The quality of the obtained solution is very much dependent upon the initial solution and continuity of the objective function (Laskari *et al.*, 2002). Further, the derivative information of the objective function is required analytically or approximately. Consequently, the gradient-based methods may experience some difficulties in trying to reach acceptable zone solutions. An algorithm such as PSO that does not require the gradient information could alleviate these shortcomings encountered by gradient-based methods. It was also reported to tackle minimax problems effectively on a set of well-known test functions (Laskari *et al.*, 2002).

5. PARTICLE SWARM OPTIMIZATION (PSO)

A relatively new evolutionary computational technique called particle swarm optimization (PSO) was introduced by Kennedy and Eberhart (1995). It is inspired by the social behavior of animals such as bird flocking in searching for food. Each particle flies in hyperspace searching for the best solution by adjusting position and velocity based on its own flying experience (*pbest*) and its companions' experience (*gbest*). The inertia weight *w* was later introduced to reportedly improve the PSO optimizer. The PSO has been applied to many applications such as optimization problems, neural network training, traveling salesman problems, and job scheduling. It is very attractive because requirement of gradient information is not needed. Hence, it is unaffected by discontinuities of the objective function. The equations used consist of flexible and well-balanced mechanisms to enhance the global and local exploration abilities. These allow a thorough search and simultaneously avoid the premature convergence. In addition, PSO uses probabilistic rules for particle's movements. Therefore, it is quite robust to local optima. Plus, it can be implemented easily with a few lines of computer code. The steps of the PSO are illustrated below (Eberhart and Shi, 2001):

1. Initialize a population of particles with random positions and velocities on *D* dimensions.
2. Evaluate the desired optimization function in *D* variables for each particle.
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* location, *pbestx*[*i*][*d*], is set to the current location in *d*-dimensional space.
4. Compare evaluation with the swarm's previous best value, (*pbest*[*gbest*]). If current value is better than (*pbest*[*gbest*]), then *gbest* = current particle's array index.
5. Change velocity and position of the particle according to the following equations, respectively:

$$v[i][d] = w*v[i][d] + c_1*rand()*(pbestx[i][d] - presentx[i][d]) + c_2*rand()*(pbestx[gbest][d] - presentx[i][d]) \quad (8)$$

$$presentx[i][d] = presentx[i][d] + v[i][d] \quad (9)$$

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

From Equation (8) and Equation (9), *v*[*i*][*d*] is a velocity of the i^{th} particle in the d^{th} dimension; *w* is inertia weight, *pbestx*[*i*][*d*] and *pbestx*[*gbest*][*d*] represent the best previous position (the position giving the particle's best fitness value) of the i^{th} particle in the d^{th} dimension and the best previous position (the position giving the swarm's best fitness value) of the $gbest^{th}$ particle in the d^{th} dimension, respectively. The current location of the i^{th} particle in the d^{th} dimension is represented as *presentx*[*i*][*d*].

As shown in Equation (8), *v*[*i*][*d*] consists of three terms. The first term is the momentum of the particle. It is computed by multiplying the inertia weight with particle's previous velocity. *w* is a control parameter, which is used to influence the current velocity from previous velocity. The larger weight implies a global exploration because the particle can fly in large area for finding a good region. On the contrary, the smaller weight results in refining the search within it. The suitable selection of the inertia weight should provide the balance between global and local search area. Therefore, the inertia weight should be initialized to a large value and then gradually decreased toward the end of the search process (Eberhart and Shi, 1998). More detail of inertia weight setting is discussed in the next section. The second term is the cognition part because the particle consults with its own best experience, *pbestx*[*i*][*d*] – *presentx*[*i*][*d*]. The third term is the social part since each particle considers the shared swarm's best experience, *pbestx*[*gbest*][*d*]–*presentx*[*i*][*d*]. c_1 and c_2 are the positive constants called "acceleration coefficients." They indicate how much the particle trusts its own and companions' experiences. The higher the constant is, the greater the acceleration of the particle will be. In general, to balance the impact of the cognition and the social parts, these two parameters are set to two to give it a mean of one (Eberhart and Shi, 1998). *rand*() is a uniformly random number generator within the (0,1) range.

These terms, their parameters, as well as the sharing information mechanism make the PSO less predictable and more flexible to avoid local optima, to improve convergence rate, and to give consistency results. The system tackles the objective function directly without the need for its gradients. This makes it more practical when complex function that is difficult to obtain gradients is encountered. Moreover, the PSO is implicitly built for speed since only primitive mathematical operators are computed. Hence, the PSO is very attractive especially for tolerance zone estimation that must deal with complex function, d_i, and minimax criterion.

6. DATA COLLECTION

The simulated data were used to verify the tolerance zone torusity so that the developed deviation model could be validated and the effectiveness of the PSO and the LSQ could be tested without concerns of measurement errors such as probe orientation, probe angle adjustment, and probe compensation. Table 1 depicts the details of the five perfect tori simulated. They assumably represented the preset nominal values of tori as specified on engineering drawing. An arbitrarily specified error zone was then added to each generated perfect torus. In doing so, two imaginary tori were implemented to cover the perfect torus with the gap of the specified error. One of those two tori would lie outside and the other one would lie inside. The measurements were taken by selecting some data points from the outer tori and some from the inner one. Together, these taken samples would represent a real manufactured toroidal object having up and down surface around the perfect torus with theoretical nominal values.

Table 1. Details of five perfect tori tested and the selected actual zones

Data set	1	2	3	4	5
x_0	0	0	0	0	0
y_0	0	0	0	0	0
u	0	0	0	0	0
v	0	0	0	0	0
c	9	15	20	30	33
r_0	1	3	4	6	7
Actual zone	0.0050	0.0062	0.0093	0.0142	0.2330

A widely used sampling method in practice, a uniform sampling, was applied to collect 64 data points from each and every simulated torus. The assumption that these points accurately represented the part surface was held by sectioning each torus normal to its circular core for 16 partitions and taking 4 points from each section. Out of these four points, two points were taken from the surface of the outer torus and the other 2 points were chosen from the surface of the inner one. Altogether there would be a total of 64 sampled points for each torus. This controlled data collection procedure should ease the computation of torusity fitting so that the specified actual zone could be reached.

7. RESULTS AND ANALYSES

To establish ideal torus from all measurement values (data points collected), four torusity-controlled factors were taken into consideration for determination of the normal distances between measurement points and surface of the ideal torus as illustrated in Figure 2 to Figure 5. The first one was the absolute distance between major radii of both actual and ideal surfaces. The second was the absolute distance between minor radii of both surfaces. The third was the distance between the center points of both tori. Finally, the last factor was the different angle between the direction vectors of both tori depicted as $p(u,v,w)$ in Figure 3. Together, these factors contributed to the discrepancy model obtained. Hence, Equation (6) was resulted and used as an error model in the minimax problem formulated. The PSO was then applied to solve the obtained formulation (Equation (7) with d_i from Equation (6)) for minimum tolerance zone torusity. It iteratively attempted to minimize the desired function that was the maximum deviation between data points and the searched ideal feature. In each iteration, every particle would search for the values of those six relevant parameters, x_0, y_0, u, v, c, and r_0, that would contribute to the deviation of each data point from the imaginary ideal torus. The maximum deviation would then be the evaluation result in Step 2 of the PSO algorithm and was minimized by the PSO.

The computation of the PSO normally depends on population size, inertia weight, maximum velocity, maximum and minimum positions and maximum number of iteration. The initial population size was chosen such that it was large enough to cover the search space within the iteration limit based on the trial runs and literature. The population size of twenty was then selected in this work. Inertia weight started from 0.9 and gradually decreased to 0.4 to balance the global and local exploration based on a linear function of time (iteration). This also contributed to improve convergence rate (Kennedy, 1997). Particles' velocities on 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 global search, and particles may fly pass good solutions. However, if v_{max} is too small, the PSO facilitates local search, and particles may not explore beyond locally good regions (Kennedy, 1997). Thus, if $v[i][d]$ is greater than v_{max}, then $v[i][d]$ is equated to v_{max}. Similarly, if $v[i][d]$ is less than $-v_{max}$, then $v[i][d]$ is equated to $-v_{max}$. In this study, v_{max} was set at 12% of the dynamic range of the variable in each dimension. In case of maximum and minimum positions of the variables in each dimension, they were chosen to represent the suitable search space, which was problem dependence. The selection of these values could be justified by considering the inspected part's relevant specification, x_0, y_0, u, v, c, and

r_0. The real manufactured torus would vary these variables to some extent. Therefore, the settings of maximum and minimum positions of those variables should not deviate from the preset nominal values (on engineering drawing) too much in specifying the boundary of the search space because torusity verification was a geometrical inspection of the manufactured part that was always made not very far from the nominal values. Thus, this rationale was realized and could save some computational time by the PSO. To illustrate the selection of these settings, Dataset 5 was used as an example. Its preset nominal values of x_0's, y_0's, u's and v's were all zeroes; c's and r_0's were set to 33 and 7, respectively. The maximum and minimum positions of x_0's, y_0's, u's and v's for all particles were then specified as (0.1, -0.1) whereas those of c's and r_0's were (33.5, 32.5) and (7.2, 6.8), respectively. This procedure was also repeated with all other datasets. They should ensure that the search spaces were never violated and the solutions obtained were always valid. The last parameter was the maximum iteration number, which was set at 300 based on trial runs. Figure 7 illustrates that the PSO converged very fast. The zone found decreased extremely quickly about 60-70 iterations. Afterward, the curve appears almost flat. This implies that the near-optimal zone solutions were reached quickly. The graph of torusity zone obtained remains flat after 200 iterations. Thus, the maximum iteration number of 300 was sufficient. Note that the same parameters settings of the PSO were used for all datasets except maximum and minimum positions that were changed to reflect various dimensions of tori tested.

The PSO based torusity fitting algorithm and the LSQ were implemented in MATLAB 6.5. Every numerical computation was performed on a PC with a Pentium IV 2.4 GHz. The PSO and the LSQ were both tested with five sets of simulated data and their results are tabulated in Table 2.

Table 2. Torusity tolerance zone obtained by the PSO and the LSQ

Dataset	1	2	3	4	5
x_0	-0.0174	0.0394	-0.0352	0.0093	0.0194
y_0	-0.0011	-0.0399	0.0185	-0.0054	-0.0672
u	-0.0026	0.0012	0.0011	-0.0010	-0.0142
v	0.0030	-0.0590	-0.0060	-0.0172	-0.0069
c	9.0190	14.9591	20.0322	29.9963	33.0692
r_0	1.0000	3.0000	4.0000	6.0000	7.0013
Actual zone	0.0050	0.0062	0.0093	0.0142	0.2330
PSO	**0.0050**	**0.0062**	**0.0093**	**0.0142**	**0.2330**
LSQ	0.0076	0.0094	0.0139	0.0215	0.3522

Clearly, the results obtained by the PSO were equal to the specified actual zones whereas those obtained by the LSQ indicated overestimations. The optimal tolerance zones were obtained for every dataset by the PSO based geometrical fitting algorithm. This shows that the proposed deviation model was very effective. It did not overestimate or underestimate the tolerance zones with every tested dataset. The PSO also performed very well. It could find the minimum zone torusity from the developed deviation model with the minimax criterion. Under the controlled environment, the outcomes should validate and verify the deviation model presented and also demonstrated that the PSO was very attractive for torusity verification. The LSQ method is generally used to find the trend of data under normal distribution condition. This assumption requirement is quite cumbersome to verify and may not hold in many problems. Moreover, even though the discrepancy function of the LSQ was quite sensitive to outliers, it still could not guarantee if the tolerance zone obtained was, in fact, minimum. This was obviously the case in this study as well. Consequently, some good parts conforming to the specification would be rejected or reworked.

Generally, the gradient-based algorithms are most suitable for a problem with a smooth objective function (continuous first and second derivatives). However, many minimax fit models do not have smooth objective functions. They may suffer from numerical instabilities with respect to convergence because the first or second derivatives of the objective functions are not continuous. The PSO could overcome this issue fundamentally. Furthermore, the global mechanism in PSO and global strategy used (making several runs from several initial solutions to avoid local optima traps) should help improve the solutions obtained. In addition, specific requirements on the mode of data collection for any of these forms were not needed. That is, the location of the measurements could be anywhere on the surface. This implies that the data points were not necessarily collected at sections perpendicular to the circular core of a torus. They could be spread around covering the entire surface of inspected torus so that they would assumedly represent this part. The form tolerances for complex shapes like this are typically left to be dealt-with by the use of profile tolerance definition. This really is the solution of two 2D problems rather than the 3D solution. This procedure results in significant inconsistencies and may be impractical in cases where accuracy of the whole profile is a requirement.

The PSO theoretically may require quite a computational time when the size of dataset is quite large (in thousands or more). However, in discrete measurement where dataset is usually in tens or hundreds, the time taken does not have much effect. Furthermore, the PSO requires only primitive mathematical operators and uses memory array to handle variables and solutions. These make it very fast to determine the near-optimal torusity

Table 3. Average computational time in seconds of both methods for ten runs

Dataset	1	2	3	4	5
PSO's time	0.5439092	0.5265912	0.5149398	0.5185026	0.5439428
LSQ's time	0.0760935	0.0781165	0.0774667	0.0842668	0.0821641

tolerance zone. The average processing time of both methods for ten runs is depicted in Table 3. The time taken by the PSO was about 0.5 second to reach the preset maximum iteration of 300 and the LSQ's computational time was around 0.08 second. Clearly, the LSQ was more efficient since theoretically it was an analytical method that required only substitution of relevant values for the respective variables into its closed form solution. Even though the PSO's execution time was longer, it was still considered very fast, about a half of a second for computing 64 collected data points. In fact, this time could be further saved by decreasing the preset maximum iteration number since the PSO could converge very rapidly as shown in Figure 7.

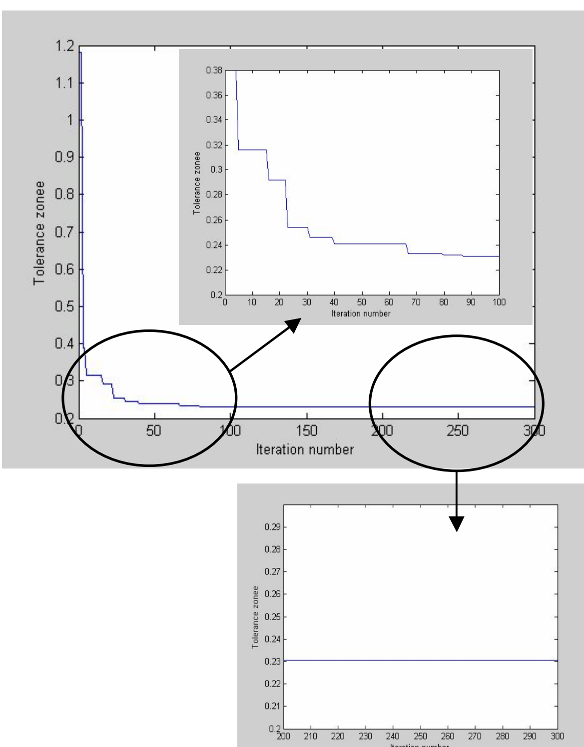

Figure 7. Convergence of PSO for Dataset 5 with the closer looks of decreasing torusity from 0-100 and from 200-300 iterations

Therefore, the PSO based zone estimation could effectively and efficiently solve two-sided minimax torusity fitting leading to minimum tolerance zone specified by the ANSI standard (ASME Y14.5M-1994, 1995). With accuracy of torusity zones, ease of algorithm and programming, and fast processing time, the models and algorithm proposed were very attractive.

8. CONCLUSION

Form tolerances inspection plays an important role in industry. The complex forms such as torus are normally dealt-with by the use of profiles of individual features which may not be very accurate when combined. Hence, a new method for finding the minimum enclosing zone of torus was proposed in this work. The presented method addressed the deviation model of torus and its zone estimation using the PSO under the assumption that the data points collected accurately represented the manufactured part surface. The true nonlinear deviation model of torusity was derived for the first time. The torusity zones obtained clearly indicate that the true nonlinear deviation model served its purpose very well and it should be used, instead of profiles or approximated linear model, for torusity verification without any specific requirements during data collection process. The PSO was next applied for determination of minimum zone torusity due to its simplicity in concept and programming, short computer code, and no requirement of gradient information. Hence, it was unaffected by discontinuity of objective functions presented in the minimax problem. It also required only primitive mathematical operators. Coupled with the PSO's global mechanism and global strategy used, the obtained results showed that the PSO algorithm provided very good solutions, especially when compared with those of the LSQ. Therefore, the PSO demonstrated much potential in finding minimum enclosing zone and consequently was very attractive for adoption in practice.

The time taken to collect data and information obtained from these data should have significant impact for torus inspection in terms of accuracy and cost. Ideally, this verification requires information of entire feature. Normally, large sample size is preferred but the inspection time would also increase. Therefore, efficient data collection consisting of sampling strategies; sample size and sampling location, should be investigated to minimize the inspection time and hence cost while maintaining the high level of accuracy of torusity inspection. In addition, systematic parameters selection of the PSO for form tolerance analysis will certainly enhance its ease of use and should be investigated further.

ACKNOWLEDGEMENT

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

REFERENCES

Traband, M. T., Joshi, S., Wysk, R. A., and Cavalier, T. M. (1989), Evaluation of Straightness and Flatness Tolerances Using the Minimum Zone, *Manufacturing Review*, **2**(3), 189-195.

Shunmugam, M. S. (1987), Comparison of Linear and Normal Deviations of Forms of Engineering Surfaces, *Precision Engineering*, **9**(2), 96-102.

Prakasvudhisarn, C., Trafalis, T. B., and Raman, S. (2003), Support Vector Regression for Determination of Minimum Zone, *ASME Journal of Manufacturing Science and Engineering*, **125**(4), 736-739.

ASME Y14.5M-1994 (1995), *Dimensioning and Tolerancing*, The American Society of Mechanical Engineers, New York.

Kennedy, J. and Eberhart, R. C. (1995), Particle Swarm Optimization, *Proceedings of the IEEE International Conference on Neural Networks (Perth, Australia)*, IEEE Service Center, Piscataway, NJ, **4**, 1942-1948.

Allahverdi, A. and Al-Anzi, F. S. (2006), A PSO and a Tabu Search Heuristics for the Assembly Scheduling Problem of the Two-stage Distributed Database Application, *Computers & Operations Research*, **33**(4), 1056-1080.

Liu, S., Tang, J., and Song, J. (2006), Order-planning Model and Algorithm for Manufacturing Steel Sheets, *International Journal of Production Economics,* **100**(1), 30-43.

Paterlini, S. and Krink, T. (2006), Differential Evolution and Particle Swarm Optimisation in Partitional Clustering, *Computational Statistics & Data Analysis,* **50**(5), 1220-1247.

Lawtrakul, L. and Prakasvudhisarn, C. (2005), Correlation Studies of HEPT Derivatives Using Swarm Intelligence and Support Vector Machines, *Monatshefte für Chemie Chemical Monthly,* **136**, 1681-1691.

Chuanwen, J. and Bompard, E. (2005), A Hybrid Method of Chaotic Particle Swarm Optimization and Linear Interior for Reactive Power Optimization, *Mathematics and Computers in Simulation,* **68**(1), 57-65.

Elbeltagi, E., Hegazy, T., and Grierson, D. (2005), Comparison Among Five Evolutionary-based Optimization Algorithms, *Advanced Engineering Informatics*, **19**, 43-53.

Hong, T., Liao, Y., and Fan, Y. (1991), A Computational Geometry Approach to Minimum Zone Evaluation of Straightness, *Journal of the Chinese Institute of Engineers*, **14**(5), 463-470.

Le, V. and Lee, D. (1991), Out-of-roundness Problem Revisited, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* **3**(3), 217-223.

Roy, U. and Zhang, X. (1992), Establishment of a Pair of Concentric Circles with the Minimum Radial Separation for Assessing Roundness Error, *Computer- Aided Design,* **24**(3), 161-168.

Roy, U. and Zhang, X. (1994), Development and Application of Voronoi Diagram Technology in the Assessment of Roundness Error in an Industrial Environment, *Computers & Industrial Engineering*, **26**, 11-26.

Roy, U. (1995), Computational Methodologies for Evaluating Form and Positional Tolerances in a Computer Integrated Manufacturing System, *International Journal of Advanced Manufacturing Technology*, **10**, 110-117.

Dhanish, P., and Shunmugam, M. (1991), An Algorithm for Form Error Evaluation Using the Theory of Discrete and Linear Chebyshev Approximation, *Computer Methods in Applied Mechanics and Engineering*, **20**, 309-324.

Wang, Y. (1992), Minimum Zone Evaluation of Form Tolerances, *Manufacturing Review*, **5**(3), 213-220.

Kanada, T., and Suzuki, S. (1993), Evaluation of Minimum Flatness by Means of Nonlinear Optimization Techniques and Its Verification, *Precision Engineering*, **15**(2), 93-99.

Carr, K. and Ferreira, P. (1995a), Verification of Form Tolerances Part II: Cylindricity and Straightness of a Median Line, *Precision Engineering*, **17**(2), 144-156.

Carr, K. and Ferreira, P. (1995b), Verification of Form Tolerances Part I: Basic Issues, Flatness, and Straightness, *Precision Engineering*, **17**(2), 131-143.

Prakasvudhisarn, C., and Raman, S. (2004), Framework for Cone Feature Measurement Using Coordinate Measuring Machines, *ASME Journal of Manufacturing Science and Engineering*, **126**, 169-177.

Aguirre-Cruz, J. A., and Raman, S. (2005), Torus Form Inspection Using Coordinate Sampling, *ASME Journal of Manufacturing Science and Engineering*, **127**, 84-95.

Lai, H., Jywe, W., Chen, C., and Liu, C. (2000), Precision Modeling of Form Errors for Cylindricity Evaluation Using Genetic Algorithms, *Precision Engineering*, **24**, 310-319.

Sharma, R., Rajagopal, K., and Anand, S. (2000), A Genetic Algorithm Based Approach for Robust Evaluation of Form Tolerances, *Journal of Manufacturing Systems*, **19**(1), 46-57.

Liu, C., Chen, C., and Jywe, W. (2001), Evaluation of Straightness and Flatness Using A Hybrid Approach Genetic Algorithms and Geometric Characterisation Method, *Journal of Engineering Manufacture*, **215**, 377- 382.

Hong, J., Shultes, C., and Anand, S. (2001), Robust

Evaluation of Flatness and Straightness Tolerance Using Simulated Annealing, *Transactions of NAM RC/SME*, **29**, 553-560.

Wen, X., and Song, A. (2004), An Immune Evolutionary Algorithm for Sphericity Error Evaluation, *International Journal of Machine Tools and Manufacture*, **44**, 1077-1084.

Eberhart, R., and Kennedy, J. (1995), A New Optimizer Using Particle Swarm Theory, *Proceedings of the Sixth International Symposium on Micromachine and Human Science*, 39-43.

Eberhart, R., and Hu, X. (1999), Human Tremor Analysis Using Particle Swarm Optimization, *Proceedings of the Congress of Evolutionary Computation,* **3**, 78-83.

Yoshida, H., Kawata, K., Fukuyama, Y., and Nakanishi, Y. (1999), A Particle Swarm Optimization for Reactive Power and Voltage Control Considering Voltage Stability, *Proceeding International Conference on Intelligent System Application to Power Systems*, 117-121.

Laskari, E. C., Parsopoulos, K. E., and Vrahatis, M. N. (2002), Particle Swarm Optimization for Minimax Problems, *Proceedings of the 2002 Congress on Evolutionary Computation*, IEEE Service Center, Piscataway, NJ, 1576-1581.

Eberhart, R.C., and Shi, Y. (2001), Particle Swarm Optimization: Developments, Applications and Resources, *Proceedings of the 2001 Congress on Evolutionary Computation*, IEEE Service Center, Piscataway, NJ, 81-86.

Eberhart, R. C., and Shi, Y. (1998), Comparison between Genetic Algorithms and Particle Swarm Optimization, *The Seventh Annual Conference on Evolutionary Programming*, 611-616.

Kennedy, J. (1997), The Particle Swarm: Social Adaptation of Knowledge, *Proceeding IEEE International Conference on Evolutionary Computation*, 303-308.