INT. J. PROD. RES., 2000, VOL. 38, NO. 15, 3471–3483

Tolerance synthesis using second-order fuzzy comprehensive evaluation and genetic algorithm

SHUPING JI†, XIAOLI LI‡* and R. DU§

In modern manufacturing engineering, tolerance synthesis is important because it directly effects product quality and manufacturing cost. This paper introduces a new method for tolerance synthesis of machining parts. The new method consists of three steps. First, machining parts are evaluated using the second-order fuzzy comprehensive evaluation (FCE). Then, a mathematical model for tolerance allocation is formed based on the machinability of the parts. Finally, the model is solved using the genetic algorithm (GA). The feasibility of the method is validated using a practical gearbox design example.

1. Introduction

Following the definition by Salomons *et al.* (1995), tolerance synthesis is concerned with tolerance optimization, or with completing partial tolerance schemes. Tolerance synthesis plays an important role in modern manufacturing. It directly effects the product quality and its manufacturing cost. In the tolerance synthesis, the biggest dilemma is product quality versus manufacturing cost. In general, the smaller the tolerance, the better the product quality, but the higher the manufacturing cost. To choose from a proper balance between the product quality and the manufacturing cost is not always an easy task. Traditionally, the tolerance synthesis is carried out based on the designers' experience, handbooks and standards (Ballu and Matheu 1993). As a result, the quality of the assembly is not guaranteed and/or the manufacturing cost may be higher than necessary. Currently, for mass production, the tolerance synthesis is mainly based on statistical techniques, in which the tolerance allocation is based on the statistical significance. On the other hand, for small batch manufacturing it is usually based on worst case scenarios (Kals 1996). They both do not guarantee the optimal solution.

In order to solve this problem, a number of methods have been developed using techniques, e.g. fuzzy logic, artificial neural network (ANN) and genetic algorithm (GA). For example, Hsu and Lee (1994) used the fuzzy logic to express the relationships among the different parts, which provided a more intuitive and realistic view. Chen and Chan (1993) presented a procedure that included an ANN and a fine-

Revision received March 2000.

† P.O. Box 427, Advanced Machining Technology Center, Harbin Institute of Technology, Harbin, 150001, PR China.

‡ P.O. Box 426, Institute of Precision Engineering, Harbin Institute of Technology, Harbin, 150001, PR China.

*Department of Industrial Engineering, University of Miami, Coral Gables, FL 33124, USA.

* To whom correspondence should be addressed at the Department of Manufacturing Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon Tong, Kowloon, Hong Kong, PR China. e-mail: me150001@cityu.edu.hk

International Journal of Production Research ISSN 0020–7543 print/ISSN 1366–588X online
http://www.tandf.co.uk /journals

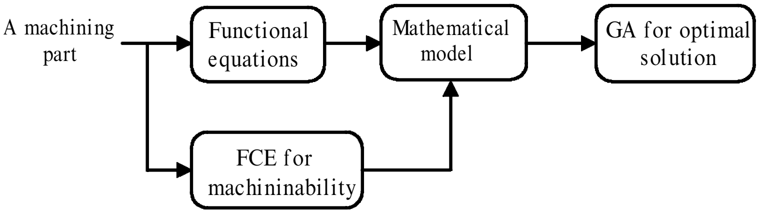

Figure 1. The proposed tolerance synthesis procedure.

tuning algorithm to optimize the tolerance allocations for achieving minimum cost. Kopardekar and Anand (1995) presented an ANN-based method for the tolerance allocation, which took the machinability and the machine tool inaccuracy into consideration. The method could predict the resulting tolerance of the assembly well. However, it required a large amount of background data and hence, it could be difficult to use. Iannuzzi and Sandgren (1994) presented a mathematical model that could be solved using GA. The model could derive the optimal tolerance allocation for small mechanical and electrical assemblies. The resulting tolerance allocation would minimize the production cost while simultaneously meeting all critical dimensional and functional constraints. Eupinet *et al.* (1996) used fuzzy logic to evaluate the machinability of a part. In their method, all fuzzy rules were derived from domain experts and could be updated when necessary. This method could be applied in different design problems, though the fuzzy design rule would be the bottleneck. Ji *et al.* (1999) presented a fuzzy comprehensive evaluation method to determine the machinability of parts, and accordingly established a tolerance allocation model. However, it did not consider the geometric complexity of the parts as well as the material property of the parts.

This paper introduces a new tolerance synthesis method for machining parts based on the second-order fuzzy comprehensive evaluation (FCE) and GA. Figure 1 shows the flow chart of the new method. From the figure, it is seen that the machinability of a part is first accessed using the second-order FCE. Then, a mathematical model is developed. Finally, the model is solved using GA. These steps are described in the subsequent sections.

2. Machinability evaluation using FCE

Before discussing the tolerance synthesis for machining parts, it is necessary to study the machinability of the parts. Note that here the machinability should not be confused with the concept of machinability in metal cutting. The later refers to the easiness of cutting of the workpiece material. In this study, however, the machinability of a part is referred to as the tardiness of machining of the part and is related to various factors, including its dimension, geometrical structure, material property and the accuracy of the machine tool. Because these factors are somewhat 'fuzzy' (e.g. it is difficult to measure the complexity of the geometrical structure of a part), the machinability of a part can only be evaluated using a fuzzy set. In this study, the second-order FCE method is used to evaluate the machinability of the part. It consists of three steps as shown below.

Step 1. Defining the fuzzy set.

In general, a machining part can be characterized by four factors: dimension; geometrical structure; material property; and machining accuracy; i.e.:

$$U = \{u_1, u_2, u_3, u_4\} = \{DS, GS, MM, MA\}, \tag{1}$$

where DS is the dimension size, GS is the geometrical structure, MM is the material property, and MA is the machining accuracy. Each factor is described by a fuzzy set,

$$u_i = \{u_{i1}, u_{i2}, \ldots, u_{ij}, \ldots, u_{in_i}\}, \tag{2}$$

where $u_{ij}, i = 1,2,3,4;\ j = 1,2,\ldots,n_i$ denotes the jth fuzzy grade of the ith factor, and n_i represents the number of grade of the ith factor. The fuzzy definitions of the four factors are shown in table 1 and their fuzzy membership functions are shown in figures 2–5, respectively.

Step 2. First-order fuzzy evaluation.

In order to impose a numeric measure, the machinability is divided into 10 equally spaced levels, i.e.

$$\varsigma = \{\varsigma_1, \varsigma_2, \ldots, \varsigma_{10}\} = \{0.1, 0.2, \ldots, 0.9, 1.0\}. \tag{3}$$

It measures the fuzzy degree that a part belongs to a specific category of machinability. Based on Wang (1996), the first-order FCE matrix is as follows:

$$R_1 = \begin{bmatrix} 0.0 & 0.0 & 0.2 & 0.4 & 1.0 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.4 & 1.0 & 0.4 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.4 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.8 \end{bmatrix}$$

		Grades of each factor			
Factors		Grade 1	Grade 2	Grade 3	Grade 4
u_1	*DS*	~5 mm	~ 25 mm	~70 mm	~ 120 mm
u_2	*GS*	Difficult	Normal	Good	
u_3	*MM*	Difficult	Normal	Easy	Very easy
u_4	*MA*	Very high	High	Normal	Low

Table 1. Main effective factors and its grade-division.

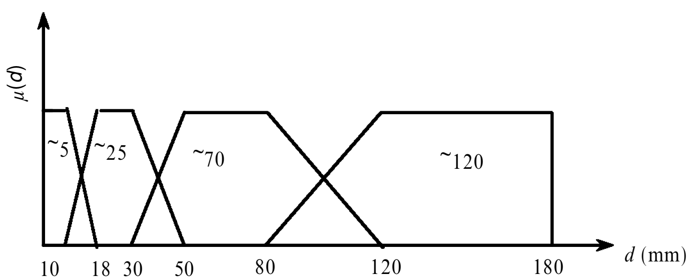

Figure 2. The fuzzy membership function of the dimension.

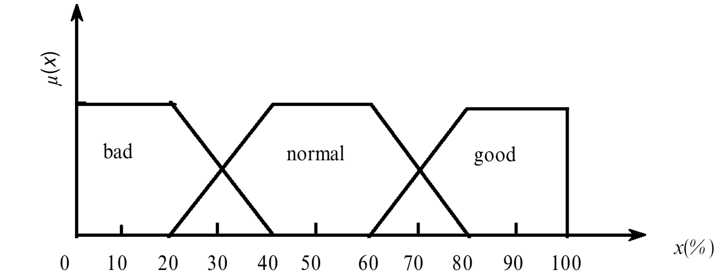

Figure 3. The fuzzy membership function of the geometry structure.

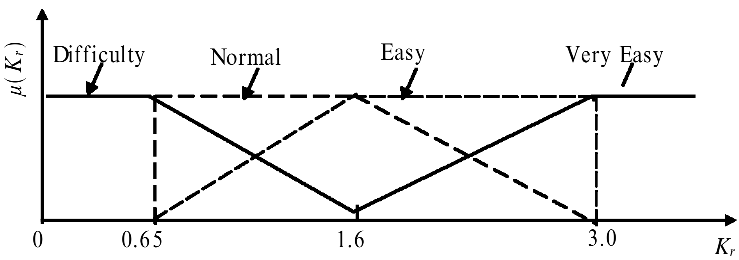

Figure 4. The fuzzy membership function of material property (K_r is the relative machinability of the material).

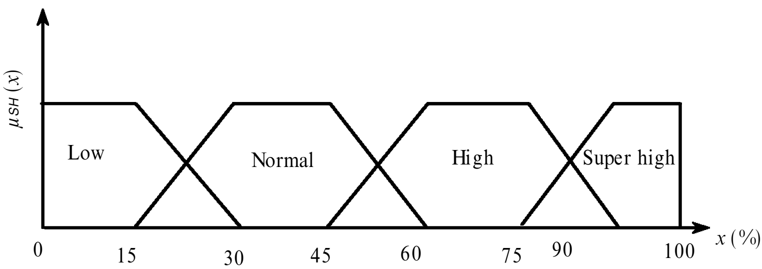

Figure 5. The membership function of process accuracy.

$$R_2 = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.8 \\ 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.6 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.2 & 0.4 & 1.0 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$R_3 = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.8 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.6 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.4 & 1.0 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$R_4 = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.8 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.6 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.4 & 1.0 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.4 & 1.0 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

Note that these matrices represent the fuzzy degree assignment to the four grades of the four factors defined in table 1. Their values are somewhat empirical and can be modified to suit specific applications.

On the other hand, given a part, a weighting vector is used to describe its fuzzy degrees:

$$A_i = (a_{i1}, a_{i2}, \ldots, a_{ini}), \tag{4}$$

where, $a_{ij} = \mu_{ij} / \sum_{i=1}^{n_i} \mu_{ij}$, $i = 1,2,3,4$; $j = 1,2,\ldots,n_i$, and μ_{ij} is the fuzzy membership of the jth grade of the ith factor of the part.

Combine the pre-defined FCE matrix and the fuzzy degrees of a part, the first-order FCE set is obtained:

$$B_i = A_i \circ R_i = (a_{i1}, a_{i2}, \ldots, a_{in_i}) \circ \begin{bmatrix} r_{i11}, r_{i12}, \ldots, r_{i1p} \\ r_{i21}, r_{i22}, \ldots, r_{i2p} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ r_{in_i1}, r_{in_i2}, \ldots \; r_{in_ip} \end{bmatrix}$$

$$= (b_{i1}, b_{i2}, \ldots, b_{ip}), \tag{5}$$

where $p = 10$ and the symbol " $\circ$ " represents the fuzzy operator. From a mathematical point of view, B_i represents the machinability fuzzy degree of the ith factor of the part, and it can be expressed as follows:

$$b_{ik} = \sum_{j=1}^{n_i} a_{ij} r_{ijk}, i = 1,2,3,4; \; k = 1,2,\ldots,10. \tag{6}$$

In summary, the first-order FCE matrix of the part is as follows:

$$R = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{bmatrix} = \begin{bmatrix} b_{11}, b_{12}, \ldots, b_{1p} \\ b_{21}, b_{22}, \ldots, b_{2p} \\ \vdots \\ b_{m1}, b_{m2}, \ldots, b_{mp} \end{bmatrix}, \tag{7}$$

where $m = 4$ and $p = 10$.

Step 3. Second-order fuzzy evaluation.

In practice, not all the factors (i.e. DS, GS, MM and MA) are equally important. Hence, the second weighting vector, A, is introduced:

$$A = (a_1, a_2, \ldots, a_m). \tag{8}$$

From a mathematical point of view, A represents the degree of importance of the factors and can be determined by the domain expert. In this study, it is calculated using the following equation:

$$A = C \circ M. \tag{9}$$

Assuming that the importance is divided into seven levels:

I_1 irrelevant,
I_2 very unimportant,
I_3 unimportant,
I_4 somewhat important,
I_5 important,
I_6 very important,
I_7 extremely unimportant.

Then, C represents the fuzzy degree of the importance and can be represented as:

$$C = [0, 0.2, 0.35, 0.5, 0.65, 0.8, 1]. \tag{10}$$

Furthermore, M is the weighting matrix defined in table 2.
For example, given a part whose factors are defined below,

dimension size — very important,
geometrical structure — important,
material machining — important,
process accuracy — important.

Then, the weighting matrix, M, is as follows:

$$M = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.50 & 0.5 & 0.5 \\ 0.50 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.0 & 0.0 & 0.25 \end{bmatrix}.$$

Using equation (9), it follows that:

$$A = (a_1, a_2, a_3, a_4) = (0.294, 0.235, 0.235, 0.235).$$

In this way, the importance of the factors can be readily assessed. Then, combine A and R, the second-order FCE is obtained:

Level of importance	I_1	I_2	I_3	I_4	I_5	I_6	I_7
0	0.67	0.25	0	0	0	0	0
0.20	0.33	0.50	0.25	0	0	0	0
0.35	0	0.25	0.50	0.25	0	0	0
0.5	0	0	0.25	0.50	0.25	0	0
0.65	0	0	0	0.25	0.5	0.25	0
0.80	0	0	0	0	0.25	0.50	0.33
1.00	0	0	0	0	0	0.25	0.67

Table 2. Fuzzy subsets that denote the important degree.

$$B = A \circ R = (a_1, a_2, \ldots, a_m) \cdot \begin{bmatrix} b_{11}, b_{12}, \ldots, b_{1p} \\ b_{21}, b_{22}, \ldots, b_{2p} \\ \vdots \quad \vdots \\ b_{m1}, b_{m2}, \ldots, b_{mp} \end{bmatrix}$$
$$= (b_1, b_2, \ldots, b_p), \tag{11}$$

where $m = 4$ and $p = 10$.

Finally, combine equations (3) and (11), the machinability of the parts is determined as follows:

$$\varsigma = \sum_{p=1}^{n} b_p \varsigma_p \bigg/ \sum_{p=1}^{n} b_p. \tag{12}$$

Following the example above, the machinability is

$$\vec{\zeta} = \{0.73, 0.40, 0.40, 0.48\}.$$

3. Modelling of tolerance allocation

Without loosing generality, it can be assumed that the chain of assembly dimension can be described as follows:

$$A_0 = f(A_1, A_2, \ldots, A_n), \tag{13}$$

where A_0 is the required assembly accuracy, and $A_i, i = 1, 2, \ldots, \mathrm{n}$, denotes the ith corresponding dimension variable. When each variable has a little increment, which is called the dimension tolerance in the dimension chain, the above equation becomes:

$$A + \Delta A_0 = f((A_1 + \Delta A_i), (A_2 + \Delta A_2), \ldots, (A_n + \Delta A_n)). \tag{14}$$

Applying Taylor expansion and omitting the higher terms, it follows that:

$$A_0 + \Delta A_0 = f(A_1, A_2, \ldots, A_n) + \frac{\partial A_0}{\partial A_1} \Delta A_1 + \frac{\partial A_0}{A_2} \Delta A_2 + \cdots + \frac{\partial A_0}{\partial A_n} \Delta A_n. \tag{15}$$

Hence,

$$\Delta A_0 = \frac{\partial A_0}{A_1} \Delta A_1 + \frac{\partial A_0}{A_2} \Delta A_2 + \cdots + \frac{\partial A_0}{\partial A_n} \tag{16}$$

or

$$\Delta A_0 = \sum_{i=1}^{n} \frac{\partial A_0}{\partial A_i} \Delta A_i = \sum_{i=1}^{n} \xi_i \Delta A_i, \tag{17}$$

where $\xi_i = \partial A_0 / \partial A_i$ represents the degree of importance of each design tolerance on the assembly, and is called the assembly sensitivity coefficient of the ith component tolerance. Also, ΔA_0 is the assembly tolerance and ΔA_i denotes the dimension tolerance of the corresponding component. Let $T_0 = \Delta A_0$ and $T_i = \Delta A_i$, equation (17) becomes:

$$T_0 = \sum_{i=1}^{n} \frac{\partial f}{\partial A_i} T_i = \sum_{j=1}^{n} \xi_i T_i. \tag{18}$$

From a physical point of view, the coefficient, ξ_i, controls the tolerance allocation. The larger the value of ξ_i, the smaller the tolerance that can be allocated. Hence, an evaluation function for the tolerance allocation, ψ_i, is introduced:

$$\psi_i = \frac{\zeta_i}{\xi_i^2}, \tag{19}$$

where, ζ_i is the machinability of the ith part obtained using the second-order FCE. Using the reciprocal model, the model of optimal tolerance allocation is:

$$\text{maximize } C = g(T_1, T_2, \ldots, T_n) = C_0 + \sum_{j=1}^{n} \frac{\psi_i}{T_i} \tag{20}$$

$$\text{subject to: } \quad l_i \le T_i \le u_i, 1 \le i \le n, l \le T_i \le u,$$

where C is the machining costs, $g(T_1, T_2, \ldots, T_n)$ describes the reciprocal relationship of the cost and the tolerance. Specifically, C_0 is the setup cost, $\vec{L} = (l_1, l_2, \ldots, l_n)$ and $\vec{U} = (u_1, u_2, \ldots, u_n)$ represent the constraints for tolerance synthesis, and l and u represent the upper and lower limits of the tolerance, respectively.

4. Solving the optimization problem using genetic algorithm

In recent years, genetic algorithm (GA) has emerged as an effective method for global optimization and found applications in many different areas. A number of excellent papers and monographs are available, e.g. Goldberg (1989) and Pirlot (1996), and the reader is referred to them.

In this study, GA is used to solve the optimization problem defined in equation (20). This is based on the fact that tolerance synthesis may involve a large number of variables. Although many optimization methods are available, however, most methods may be trapped in local optimal. Hence, in order to obtain the global optimal, GA is used. It takes three steps as shown below.

Step 1. Setting up the optimization problem using penalty function.

As shown in equation (20), tolerance synthesis is a constrained optimization problem defined below:

$$\begin{aligned} &\text{Minimize} \quad && g(T_1, T_2, \ldots, T_i, \ldots, T_n) \\ &\text{subject to} \quad && \varepsilon_{sj}(T_1, T_2, \ldots, T_I, \ldots, T_n) \le \varepsilon_{aj}, \quad j = 1, 2, \ldots, m, \end{aligned} \tag{21}$$

where $\varepsilon_{sj}(T_1, T_2, \ldots, T_i, \ldots, T_n)$ are the constrains, and ε_{aj} denotes the limits of the stack-up tolerance deviations. Also, it is well known that the constrained optimization problem is difficult to solve. Hence, it is desirable to convert it to a non-constrained optimization problem through a penalty function. This results in the non-constrained optimization problem below:

$$\text{Minimize} \quad \Psi(T_1, T_2, \ldots, T_i, T_i, \ldots, T_n, w), \tag{22}$$

where

$$\Psi = g(T_1, T_2, \ldots, T_i, \ldots, T_n) + w \cdot \sum_{j=1}^{m} \left(q(\varepsilon_{sj}(T_1, T_2, \ldots, T_i, \ldots, T_i, \ldots, T_n))\right)^2 \tag{23}$$

$$q(\varepsilon_{sj}) = \begin{cases} 0 & (\varepsilon_{sj} \leq \varepsilon_{aj}) \\ \dfrac{\varepsilon_{sj}}{\varepsilon_{aj}} - 1 & (\varepsilon_{sj} >_{sj}). \end{cases} \tag{24}$$

Note that the scalar quadratic function, $q(\cdot)$, represents the penalty when the stack-up tolerance conditions are not satisfied, and w is a large positive number representing the cost of the penalty.

It should be noted that with a finite value of w, the optimal solution of equation (22) is slightly different from that of equation (21). However, it has been shown that the difference is small when the GA is used [Pirlot 1996].

Step 2. Coding and fitness function
In order to apply the GA, it is necessary to convert the variables of the model to codes. Specifically, each tolerance variable, T_i, is considered as an individual. The genotypes of the individuals are coded as $I = [G_1, G_2, \ldots, G_i, \ldots, G_n]$, where G_i is a gene correspondent to the tolerance variable T_i. Furthermore, the cost function is converted to the fitness function f as defined below:

$$f = \frac{1}{\psi} \tag{25}$$

Step 3. Solving the model by genetic manipulations.

Step 3.1. Creating the initial population. The operation starts with seeding, which is a process of creating an initial set of candidate solutions called the initial population. Seeds can be chosen heuristically or generated at random. For example, Q individuals $\{I_1, I_2, \ldots, I_k, \ldots, I_Q\}$, where, $I_k = [G_{k1}, G_{k2}, \ldots, G_{ki}, \ldots, G_{kn}]$ can be chosen from the gene pool, and the initial genotypes, G_{ki}, are randomly chosen. In this study, the tolerance population is $Q = 100$.

Step 3.2. Choosing selection operator. The selection operator determines the sets of individuals that survive at the next generation. Let the fitness value for the individual I_k be f_k, then the probability P_{sk} that the individual I_k will be selected from the current generation is calculated by:

$$P_{sk} = (f_k)^r \Big/ \sum_{k=1}^{Q} (f_k)^r, \tag{26}$$

where r is a positive number. Based on the fitness survive rule, the individuals with large P_{sk} will be chosen.

Step 3.3. Choosing crossover operator. The crossover operator allows the selected members of the population to exchange characteristics among themselves. In the crossover operation, two parents I_a

and I_b are selected from the population with the probabilities P_{sa} and P_{sb}. Furthermore, the new individuals I'_a and I'_b are chosen to crossover from their parents I_a and I_b with the probability P_c (in this study, $P_c = 0.90$). The crossover divides the two parents into sub-parts at the points of crossover, and the new individuals are formed by swapping the sub-parts. For example, the crossover between mating parents represented by seven-digit binary strings can be embodied artificially as follows:

$$I_a = 110|1101 \quad I'_a = 110|0010$$

$$I_b = 101|0010 \quad I'_b = 101|1101.$$

where the line between the third and fourth digits represents the crossover point. Based on the fitness survive rule, the individuals that do not survive to the next generation will undergo the crossover operation.

Step 3.4. Choosing mutation operator. Mutation operator plays an important role in safeguarding the process from premature loss of valuable breeds during the selection and crossover operations. In the mutation, the contents of genes were randomly selected and changed with the probability P_m. The mutation probability P_m should be carefully chosen. If it is too low, then the search may be trapped at a local optimum. On the other hand, if it is too high, then the propagation of good genotypes will be unduly hindered and the search will degenerate to a random search. In this study, $P_m = 0.01$.

The GA algorithm starts from the initial condition, then searching using selection operator, crossover operator and mutation operator until a convergence criterion is met, the optimal solution is found.

5. A case study

In order to validate the presented method, the tolerance synthesis for a gearbox assembly is studied. As shown in figure 6, the critical dimension is the clearance between the left of the gear and the side of gearbox, A_0, which is related to the dimensions A_1, A_2, A_3 and A_4. Hence, the assembly dimension chain can be expressed as follows: $A_0 = A_1 - A_2 - A_3 - A_4$.

The required tolerance is $0.0 < A_0 < 2\,\mathrm{mm}$. The importance coefficients are assumed as: $\xi_1 = 1.0, \xi_2 = \xi_3 = \xi_4 = -1.0$.

Table 3 shows the required design conditions. Following the example in section 2, the machinability vector of four parts is $\vec{\zeta} = \{0.73, 0.40, 0.40, 0.48\}$. This indicates that A_1 is the most difficult to machine, while A_2 and A_3 are the easiest.

Furthermore, using equation (19), the comprehensive factor vector of the tolerance allocation is obtained: $\vec{\psi} = \{0.73, 0.40, 0.40, 0.48\}$. Then, using equation (20), the model for the tolerance allocation is:

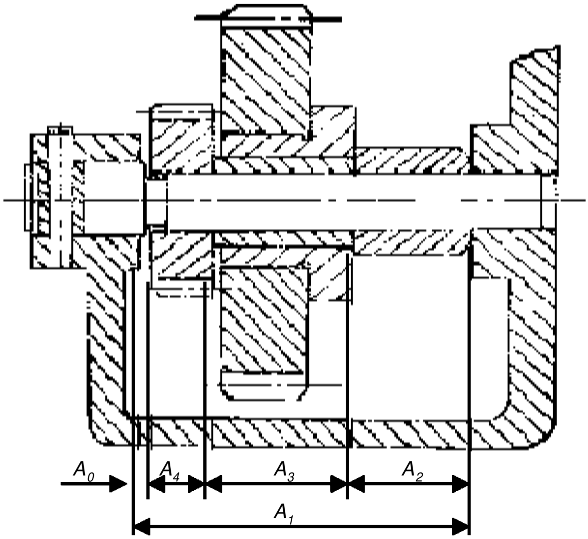

Figure 6. A gearbox assembly.

Factor		The dimension of parts			
		A_1	A_2	A_3	A_4
u_1	Dimension size	190 (mm)	74 (mm)	78 (mm)	36 (mm)
u_2	Geometrical structure	Bad (20, 20, 5, 5, 1)	Good (85, 85, 5, 5, 1)	Good (85, 85, 5, 5, 1)	Good (85, 85, 5, 5, 1)
u_3	Material machinability	Casting iron ($K_r = 1.5$)	Cu-Al alloy ($K_r = 4.0$)	Cu-Al alloy ($K_r = 4.0$)	45# Steel ($K_r = 1.2$)
u_4	Process accuracy	End milling (15, 15, 5, 5, 1)	Turning (35, 35, 5, 5, 1)	Turning (35, 35, 5, 5, 1)	Turning (35, 35, 5, 5, 1)

Table 3. The known design conditions.

$$\text{Minimize} \qquad C = C_0 + \frac{0.73}{T_1} + \frac{0.40}{T_2} + \frac{0.40}{T_3} + \frac{0.48}{T_4}$$

$$\text{Subject to:} \qquad 0.0 \le T_t \le 2.0, \quad i = 1,2,3,4$$

$$0.0 \le T_0 \le 2.0.$$

Assuming that the setup cost is $C_0 = 0$, and solving the above model using the GA algorithm, the dimension tolerances are obtained as shown in table 4. Here, the worst case analysis and statistical analysis are convergence criteria. Under the worst case analysis criteria, the GA search is stopped as long as all the tolerances meet the boundary condition (the constraints). On the other hand, under the statistical analysis criterion, the GA search does not stop until no significant change can be made in a statistical sense. From the table, it is seen that the statistical analysis criterion works better.

Dimension	A_1 (190 mm)	A_2 (74 mm)	A_3 (78 mm)	A_4 (36 mm)	Total cost (C)
Worst-case analysis criterion	0.64 mm	0.44 mm	0.45 mm	0.47 mm	3.960
Statistical analysis criterion	1.20 mm	0.84 mm	0.85 mm	1.06 mm	2.000

Table 4. The tolerances obtained using the presented method.

Methods	A_1 (190 mm)	A_2 (74 mm)	A_3 (78 mm)	A_4 (36 mm)	Total cost (C)
ST Worst-case analysis	0.50 mm	0.50 mm	0.50 mm	0.50 mm	4.020
ST Statistical analysis	1.0 mm	1.0 mm	1.0 mm	1.0 mm	2.010
PS Worst-case analysis	1.0 mm	0.396 m	0.412 mm	0.490 mm	5.237
PS Statistical analysis	1.71 mm	0.676 mm	0.704 mm	0.625 mm	3.064
CP Worst-case analysis	0.65 mm	0.48 m	0.49 mm	0.38 mm	4.036
SI Worst-case analysis	0.5 mm	0.5 mm	0.5 mm	0.5 mm	4.020
SI Statistical analysis	1.0 mm	1.0 mm	1.0 mm	1.0 mm	2.010
CF Worst-case analysis	0.72 mm	0.40 mm	0.40 mm	0.48 mm	4.014
CF Statistical analysis	1.40 mm	0.78 mm	0.78 mm	0.90 mm	2.080

Table 5. A comparison with the other tolerance synthesis methods.

For the purpose of comparison, several traditional tolerance synthesis methods are also tested including: (i) same tolerance method (ST); (ii) proportional-scaling method (PS); (iii) constant precision factor method (CP); (iv) same influence method (SI); and (v) comprehensive factor method (CF). The results are shown in table 5.

From tables 4 and 5, the following observations can be made.

(1) The new method has the best performance (the total cost is minimum) among all the methods compared. This may be attributed to the fact that the new method takes various factors, e.g. geometry complexity and material machinability, into consideration. Hence, the total cost is minimized.
(2) Statistical analysis methods usually outperform the worst-case analysis method.
(3) Worst-case analysis methods usually result in large deviation in tolerance synthesis.
(4) ST statistical analysis method, SI statistical analysis method and CF statistical analysis method are also very effective.

6. Conclusion

Based on the discussions above, the following conclusions can be drawn.

(1) The new tolerance synthesis method is effective in delivering the tolerance allocation that will minimize the manufacturing cost. In comparison to the existing methods (the same tolerance method, the proportional-scaling method, the constant precision factor method, the same influence method and the comprehensive factor method), the new method provides the tolerance allocation that has the minimum manufacturing cost.

(2) The new method is easy to use because it converts the linguistic information (e.g. the workpiece material is difficult to cut) to numerals and accordingly, makes use of it for the best of tolerance synthesis.
(3) Based on a test on a practical gearbox example, it is seen that the new method is reliable.

References

BALLU, A. and MATHEU, L., 1993, Analysis of dimensional and geometrical specifications: standards and models. In *Proceedings of 3rd CIRP Seminars on Computer Aided Tolerancing*, pp. 157–170.

CHEN, R. M. and CHAN, W. W., 1993, Efficient tolerance design procedure for yield maximization using optimization techniques and neural network. In *IEEE International Symposium on Circuits and Systems*, Vol. 3, pp. 1793–1796.

EUPINET, E., BALAZINSKI, M. and CAOGALA, E., 1996, Tolerance allocation based on fuzzy logic and simulated annealing. *Journal of Intelligent Manufacturing*, **7**, 487–497.

GOLDBERG, D. E., 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning* (New Jersey: Addison-Wesley).

HSU, W. and LEE, C. S., 1994, Fuzzy application in tolerance design. *IEEE International Conference on Fuzzy Systems*, 94CH3430-6, Vol. 2. pp. 1182–1186.

IANNUZZI, M. and SANDGREN, E., 1994, Optimal tolerancing: the link between design and manufacturing productivity. In *The 6th International Conference on Design Theory and Methodology*, ASME, Design Engineering Division, Vol. DE-68, pp. 29–42.

JI, SHUPING, LI, XIAOLI, and MA, YULIN, 2000, Optimal tolerance allocation using fuzzy comprehensive evaluation and genetic algorithm. *International J. Advanced Manufacturing Technology*, **16**, 451–468.

KALS, H., 1996, A computer aided tolerancing tool, I: tolerance specification, and II: tolerance analysis. *Computers in Industry*, **31**, 161–186.

KOPARDEKAR, P. and ANAND, S., 1995, Tolerance allocation using neural networks. *International J. Advanced Manufacturing Technology*, **10**, 269–276.

PIRLOT, M., 1996, General local search methods. *European J. Operational Research*, **92**, 493–511.

SALOMONS, O. W. and HARTEN, A., 1995, A computer-aided tolerancing tool based on kinematic analogies. *Proc. 4th CIRP Seminar on Computer-Aided Tolerancing*, Tokyo, pp. 53–72.

WANG, CHENBAO, 1996, *The Basic Knowledge of Mechanics Manufacturing Process* (Nanjin, PR China: Southeast University Press), in Chinese.