Int J Adv Manuf Technol (2014) 73:1219–1236
DOI 10.1007/s00170-014-5908-2

ORIGINAL ARTICLE

Optimal geometric tolerance design framework for rigid parts with assembly function requirements using evolutionary algorithms

A. Saravanan · C. Balamurugan · K. Sivakumar · S. Ramabalan

Received: 16 September 2011 / Accepted: 28 April 2014 / Published online: 15 May 2014

Abstract Tolerance design is always a challenging task for engineers, since it need to satisfy multidisciplinary functions. Engineering design is done in two stages: assembly design and detail design. In the first stage, an assembly is designed considering certain system level functions and in secondary detail design stage; decomposition of the assembly is done and process tolerancing is employed for the parts. At the secondary detail design stage, designer adopts geometrical dimensioning and tolerancing (GD&T) concepts for process tolerancing. Hence, assembly and detail design are done in different phases with dissimilar perspectives. As a result, geometric tolerance design often lands in conflict, redesign, and in the case of concurrent engineering, costly reiterations are performed. This conflict occurs because of two vital reasons: (1) a gap exists between these two design stages and no common relation between them; (2) GD&T is adopted in the secondary stage, which is not available in primary stage. This paper offers a framework for a design engineer to bridge the gap and to establish the relation between these stages. A nonlinear combinatorial optimization problem is framed based on assembly function requirement (AFR), and tolerance values are optimized with appropriate constraints. Nontraditional Elitist Non-dominated Sorting Genetic Algorithm (NSGA-II) and differential evolution (DE) algorithms are used to solve the problem. For the allocated position tolerances, appropriate sensitive factors are indicated to facilitate design improvement. Finally, a case study is used to illustrate the complete framework.

Keywords Assembly function requirement · Composite position tolerance · Differential evolution algorithm · Elitist Non-dominated Sorting Genetic Algorithm · Geometrical dimensioning and tolerancing · Sensitive factor and tolerance optimization

Notations

C_j	Manufacturing cost tolerance function, j=1 to 4
t_i	tolerance value of the features
T_x	Translational representation along x-axis
T_y	Translational representation along y-axis
T_z	Translational representation along z-axis
u_i	Translational vector value along x-axis
v_i	Translational vector value along y-axis
w_i	Translational vector value along z-axis
R_x	Rotational representation along x-axis
R_y	Rotational representation along y-axis
R_z	Rotational representation along z-axis
α_i	Translational vector value along x-axis
β_i	Translational vector value along y-axis
γ_i	Translational vector value along z-axis
τ_i	Key matrix
ω_i	Rotational key vector
ε_i	Directional or translational key vector

A. Saravanan (✉) · C. Balamurugan
Department of Mechanical Engineering, M.A.M College of Engineering, Tiruchirappalli 621105, Tamil Nadu, India
e-mail: varunsarav@gmail.com

C. Balamurugan
e-mail: rlc_bal@yahoo.co.in

K. Sivakumar
Department of Mechanical Engineering, Bannari Amman Institute of Technology, Sathyamangalam 638401, Erode district, Tamil Nadu, India
e-mail: ksk71@rediffmail.com

S. Ramabalan
EGS Pillay Engineering College, Nagapattinam 611002, Tamil Nadu, India
e-mail: cadsrb@gmail.com

1 Introduction

Design engineers are usually anxious to tight tolerances in order to ensure proper function and quality product. But a tighter tolerance normally requires extensive manufacturing effort which in turn translates higher manufacturing cost. A manufacturing engineer is interested in lowering the cost, increasing the production, and hence, encourages loose tolerances. Also, engineering design is done in two stages. (1) Assembly design stage, in which an entire system is designed considering complicated system level issues. (2) Later detail design stage is carried out in which the assembly is decomposed and process tolerances are incorporated to meet the functions. Design engineers adopt geometrical dimensioning and tolerancing concepts, which are designed to ensure that products produced, will meet the requirements like functionality, minimum cost, and maximum interchangeability [1]. With the above complications, the following issues are faced by the engineers in industry:

1. Lack of relation between the two design stages,
2. Geometrical dimensioning and tolerancing (GD&T) methodology not available at the initial assembly design stage,
3. Conflict between design and manufacturing departments and
4. Minimum relative manufacturing cost with maximum interchangeability.

To change these undesirable situations and to accomplish the objectives of tolerance design, there have been developments in optimal tolerance designs [2].

1.1 Literature review

Appreciable researches have been carried out in the regard of geometric dimensioning and tolerancing. Formulation of tolerance assignment as a nonlinear optimization problem for GD&T assembly was introduced by Ngoi and Min [3]. At the manufacturing stage, tolerances are simply allocated to working dimensions depending upon the process capability of the manufacturing process, stock removal for each successive operation and the design tolerance level of the nominal dimension. Hu and Ziong [4] formulated a technique for optimal tolerance allocation choosing one of many possible process alternatives. They used genetic algorithm (GA), with cost as the objective function and design requirements as constraints. This technique is suitable where sequences and tolerances of operations are fixed. When geometric tolerances are designed, simultaneously evaluation of them is important.

Mohamed [5] presented an algorithm for discrete optimization problem related to tolerance design and used simulated annealing and sequential quadratic programming techniques for solving the problem. The work justifies the use of developed algorithms for obtaining near-to-global optimum points. Zhang et al. [6] introduced mathematical modeling of assembly tolerance specification and tolerance zone types by means polychromatic sets theory, and this method establishes the relation matrix. Reasoning algorithm helps to unified formal mathematical models from assembly till corresponding tolerance zone. Khodaygan et al. [7] presented a new feature-based approach to tolerance analysis for mechanical assemblies with GD&T, and it is expressed by small degrees of freedom of geometrical entities that are described by the tolerance zones.

GD&T is a multifunctional activity in which interaction is done with design, manufacturing, process planning, and quality control. Several methods have been developed by researchers to establish the multifunctional relation. Bai et al. [8] proposed a computer-aided process planning technique using GD&T and optimized the machining datum sets and machining tolerance for rotational parts. A mixed nonlinear discrete optimization problem was formed and solved by GA. Pandya et al. [9] presented a methodology to establish a datum and restrain appropriate number of degrees of freedom on each mating part. Finally, a tolerancing schemes are established to create a set of coincident frames and in turn a datum is established. Similarly Demoly et al. [10] used GD&T for finalizing the sequence of machining.

Zbigniew et al. [11] and Robin et al. [12] simulated the material condition. They developed the virtual boundaries and found the worst cases of assembling a part. When quality is important for tolerance design, the quality loss is measured as the loss to society that occurs when a product deviates from the optimum set of design parameters. Muthu et al. [13] proposed a quality loss function which estimates the cost of quality value versus target value and the variability of the product characteristic in terms of the monetary loss due to product failure in the eyes of consumer.

There have been researches carried to evaluate the geometrical tolerances assigned in design. Andrea et al. [14] proposed a roundness evaluation framework using minimum zone tolerance by specifying the tolerance zone and the same was optimized by using GA. Similarly, Yashpal et al. [15] introduced a new methodology to evaluate the form tolerances and framed a nonlinear optimization problem and solved by means of particle swarm optimization technique.

Loof and Soderberg [16] proposed a multiobjective tolerance allocation problem to minimize the manufacturing cost. He justified the artificial cost induced on the product affects it quality. A methodology was framed to balance the manufacturing cost and quality; it also facilitates an automatic decomposition of the product based on its requirement and critical dimensions. A rear lamp of a car was chosen as case study in which the requirements were identified and investigated for the permissible variations allowed owing to its location.

Iannuzzi and Sandgren [17] developed a computational design tool for optimal tolerance allocation on mechanical and electrical components. This technique allocates optimal tolerances to reduce manufacturing cost and to increase productivity, quality and customer satisfaction. Highly robust nonlinear genetic algorithm (GA) is used to solve complex tolerance problem from individual parts to assemblies. GA is used optimization and Monte Carlo simulation for tolerance analysis.

Tolerance design needs to be reiterated if the design or the assembly fails. When a design fails, identifying the root cause is a tedious task. Mohamed [18] and Zhang [19] proposed sensitive factors for the dimensions allocated. Sensitive factors were determined with respect to the datum allotted. Sensitive factor helps in identifying the opportunities for design improvement.

All these simultaneous tolerance synthesis models considered linear as well as nonlinear tolerances and never dealt alone with geometrical tolerances in both assembly and detail design stages. From the literature survey, it is clear that efforts have been taken indigenously either during assembly or during part level and partially from assembly level to part level.

2 Scope of the framework

The purpose of this paper is to establish the following works towards geometric tolerance design

a. To develop a general optimization frame work for GD&T problems,
b. To establish the assembly function requirement (AFR) as a prime constraint for the optimization framework,
c. The application of evolutionary algorithms to complex geometric tolerance design problems and
d. To initiate the relative sensitive factors among the basic dimensions.

3 GD&T design framework

Geometric definitions of tolerances were often left to assumption. If parts were made in one geographic location and mating parts in another, even though both were made as per drawing specifications, when brought together, the parts would not always mate in assembly. GD&T, the American National Standards Institute's standard (ANSI Y14.5-1994) [1] is the result of many years of study and collaboration by dedicated individuals. It is a language of symbols that allow us, perhaps for the first time, to convey those ideas in a way that is precise and logical, being based upon the principles of 'function' and 'relationship.' When used properly, GD&T will give the manufacturer increased tolerance, creating a more manufacturable product at lower cost, while not affecting the final fit of the finished product. It can provide economical and technical advantage by ensuring integrity of the design requirements, interchange ability of parts, maximum productivity, and uniformity of drawing interpretation [16, 17].

The ANSI standard [20] defines tolerances geometrically as zones within which the part features or their resolved geometries (center-plane, center-line and center-point) are constrained to lie. Now that geometric tolerances and statistical tolerancing are both becoming widely accepted, it is important to analyze geometric tolerances statistically. GD&T, thus selected and specified in the design stage, is further revised according to a detailed process plan to obtain the manufactured dimensions and tolerances of a mechanical part. The calculated manufacturing tolerances are not only functions of GD&T, but it depends on the capabilities of the manufacturing processes and of the manufacturing equipments.

3.1 Framework procedure

In order to determine design and manufacturing tolerances simultaneously for an optimum decision function such as the total relative manufacturing cost, a decision-making process is utilized with the following assumptions:

1. Each process has a normal distribution and is under statistical control.
2. The dimensions in a dimension chain and the process for each dimension are independent.
3. Geometric tolerance for a feature is considered for its actual mating envelope (AME) and corresponding tolerance zone, not for the design feature.
4. Material condition is assumed as AME and feature is obtained directly from machine.

The optimization procedure is implemented in nine steps as follows:

1. Identify the principal feature of the assembly elements with its AFR,
2. Allocate various geometrical tolerance symbols and zones as required,
3. Establish a mathematical expression for the objective function,
4. Establish the equation for AFR and use it as a constraint,
5. Establish the 3D rotational and translational stack-up constraints,
6. Institute the tolerance constraint by identifying the tolerance zone,
7. Optimize the nonlinear combinatorial problem by NSGA-II and DE,
8. Observe the results and update in the drawings.
9. Calculation of sensitive factors for position tolerances.

4 Optimal geometric tolerance design model

For optimal determination of geometric tolerances, an optimization problem with appropriate trade-off between assembly function and manufacturing cost is formulated as follows:

4.1 Objective function

The objective function minimizes the total cost in an assembly. Here, a cost–process tolerance function is adopted as the manufacturing cost component of the objective function. Each manufacturing operation is modeled with an appropriate geometric tolerance relationship. This avoids the inaccuracies of cost–design tolerance models and permits direct distribution of design tolerances to each process tolerance. The total cost is the sum of manufacturing cost of each component's tolerance.

$$\text{Minimize}: \left[\sum_{i=1}^{n}\sum_{j=1}^{4} C_j(t_i)\right] \quad (1)$$

where, $C_j(t_i)$ = manufacturing cost of tolerance t_i for model j
t_i = tolerance value of the features.

4.1.1 Manufacturing cost-tolerance function

Manufacturing cost-tolerance function describes the manufacturing cost incurred to produce the assigned geometrical tolerance. Several formulations of machining cost–tolerance models for modeling the cost–tolerance relationship such as exponential model, inverse square model, inverse power model, and inverse model have been developed and reviewed by different researchers [2]. Although nontraditional cost functions model the characteristics of the manufacturing processes more accurately, for a balance between modeling accuracy and computational simplicity, the exponential cost function model used by Hu and Xiong [4] is considered the best and the machining cost–tolerance relation is broadly classified into four models.

Model 1 Manufacturing cost-tolerance function for size tolerances (±) for shaft feature (external cylinder) is

$$C_1(t_i) = 10^{-5} + 10^{-5}t_i + 67.3e^{-2.59t_i} \quad (2)$$

Model 2 Manufacturing cost-tolerance function for size tolerances (±) for hole feature (internal cylinder) is

$$C_2(t_i) = 10^{-5} + 10^{-5}t_i + 57.6e^{-1.59t_i} \quad (3)$$

Model 3 Manufacturing cost-tolerance function for position tolerance for cylindrical feature is

$$C_3(t_i) = 8.052 + 10^{-5}t_i + 30.87e^{-12.09t_i} \quad (4)$$

Model 4 Manufacturing cost-tolerance function for perpendicular tolerance is

$$C_4(t_i) = 5.425 + 10^{-5}t_i + 12.43e^{-10.82t_i} \quad (5)$$

4.1.2 Geometrical tolerance allocation

To gain the benefits of geometric control, a number of fundamental concepts are used to build a structured design methodology. The most important of these ideas involves the use of geometric symbols. There are 15 types of geometrical tolerances and arranged in two groups refer in Fig. 1. Each is a critical element in the creation of a system of design methodologies and geometric control.

When designers do not have an operative knowledge of these elements and there is no established concurrent engineering team, an incomplete product definition results, ceding control of the product and its allied process design to individuals located downstream in the development process. These downstream "designers" now have the freedom but not necessarily the knowledge to make optimum decisions about product function; certainly, they should not be the ones to provide the primary functional definition of the geometric design. Therefore, the knowledge of the geometric tolerance symbol and corresponding tolerance zone is essential to perform tolerance allocation.

When the assembly is decomposed into features, the appropriate geometric symbol must be chosen by the designer and to cross-check the same with models discussed in Section 4.1.2. The designer must use the same symbol in the shop floor drawing.

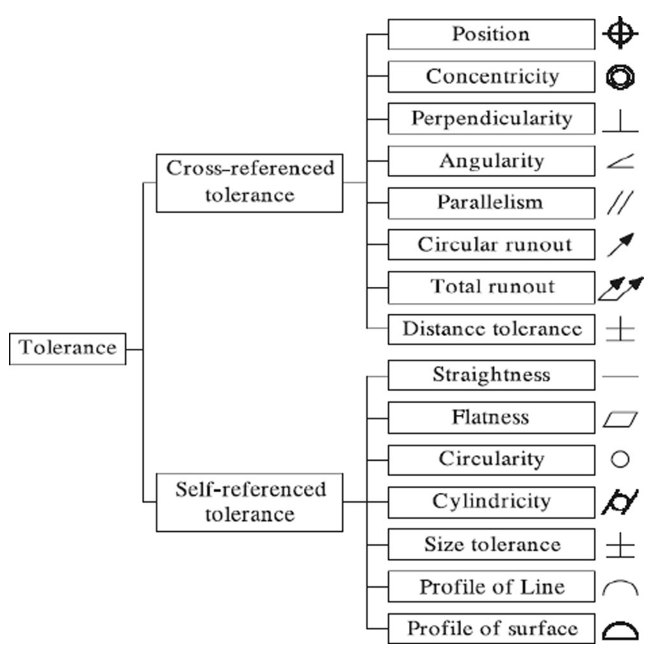

Fig. 1 Geometric tolerance symbols

4.2 Constraints

The above-mentioned optimization problem is subjected to four constraints related to both the design and manufacturing tolerances. They are as follows:

1. AFR,
2. Rotational constraints,
3. Translational constraints and
4. Tolerance zone constraints.

When geometrical tolerances are involved, the statistical tolerance design problem becomes complicated with 3D. There by study on 3D representation of geometric tolerances and tolerance zones were carried out by researchers for the last decade. Jerome and Denis [21] summarized the following models to represent the geometrical tolerances and respective zones.

a. Variational model

In this approach, the geometry of the real part is described by variations of the nominal geometry. Each surface or geometric element of the real part is associated with a perfect shape element. In the variational approach, the variations between substituted elements can be described as follows:

- by vectors and degrees of freedom (DOF),
- by small displacement torsors,
- by matrices,
- by metric tensors,
- by virtual gauges,
- by a finite set of constraints.

b. Envelope zone model

In the envelope zone approach, the real geometry has to lie in an envelope zone. This zone is obtained by offset of the nominal geometry. The real geometry of a part is described as a set of tolerance zones. In a mechanism, each tolerance zone corresponding to the real geometry of a part is connected to others by constraints.

c. Structural model

In dimensioning and tolerancing, the often-used structural model presented in [12] is based on the technologically and topologically related surface (TTRS) theory. With TTRS, a part is described by a tree of TTRS. This data structure is efficient in detail design, but it seems difficult to use in conceptual design for the product description that is based on functional requirements and a poor geometric description.

d. Set of constraint model

In the approaches presented in the defects of the real geometry are described by a finite set of geometric constraints. 3D dimension-chain computation consists in Minkowski sums and intersection operations.

They also conclude that variational model and envelope zone model are quiet reliable for analytical and computational purposes and universally accepted by several authors [4–7, 20, 21].

4.2.1 AFR constraints

AFR is defined as the prime requirement of an assembly based on which the associated parts are designed. It is acting as the link between the two stages of design as discussed in earlier chapters. From the "Variation model" of Section 4.2, it is strongly recommended that variational model is opt to represent the AFR. To analytically define an AFR is by interpreting the DOF for a part when kept in space as illustrated in Fig. 2.

There are six DOFs available for a part when kept in space. They are three independent translations in x-, y-, and z-axis (T_x, T_y, T_z), negative sign can be included on reverse translations say ($-T_x$, $-T_y$, $-T_z$). In addition to its three independent clockwise rotations (R_x, R_y, R_z) and counterclockwise may represented by negative sign inclusion like ($-R_x$, $-R_y$, $-R_z$). On forming the problem with respect to tolerance design, the values in each axis will be very small so they are designated by key parameters.

Translational movements (T_x, T_y, T_z)=key parameters (u_i, v_i, w_i) for translation along the x-, y-, and z-axis

Rotational movements (R_x, R_y, R_z)=key parameters (α_i,β_i,-γ_i) for rotation along the x-, y-, and z-axis

Key matrices are the tolerance constraints that can be formed using these parameters and it is defined as follows:

$$\tau_i = \{\varepsilon_i \omega_i\} \tag{6}$$

where, $\omega_\mathbf{i}=(\alpha_\mathbf{i},\beta_\mathbf{i},\gamma_\mathbf{i})^\mathbf{T}$ is rotational key vector and $\varepsilon_i=(u_i,v_i,w_i)^T$ is directional or translational key vector are established with

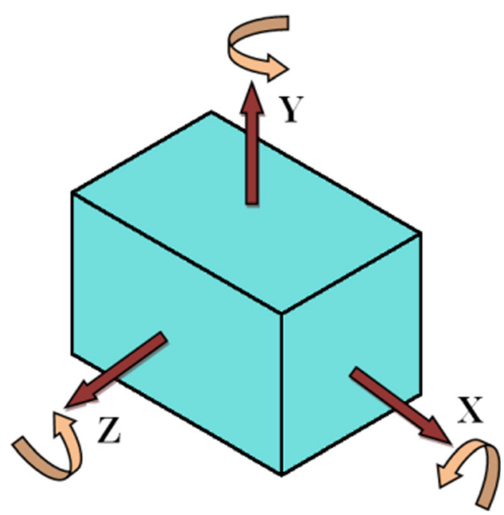

Fig. 2 Degrees of freedom (DOF) in space

3D parameters with respect to rotational and translational of a part in space. From this study, it is clear that stack-up is classified into rotational stack-up and translational stack-up, will be discussed in Section 4.2.2 and Section 4.2.3.

On considering two parts kept above each other in Fig. 2, then the resultant key matrix for the assembly is

$$\tau_{\text{assembly}} = \begin{bmatrix} \alpha_1 & u_1 \\ \beta_1 & v_1 \\ 0 & 0 \end{bmatrix} \quad (7)$$

From the key matrix, it is possible obtain the AFR constraint equation based root sum square (RSS) approach. Krulikowski [22] presented RSS approach in his book as an approach is employed to account for the low likelihood of all dimensions occurring at their extreme limits simultaneously. The sum of squares is a mathematical treatment of the data to facilitate the legitimate addition of measures of variability. The RSS method is used to determine if a functional fit is going to occur between the mating assemblies. It is assumed that the sample data we are working which comes from reasonable approximations of normal distributions.

With respect to RSS approach, the AFR constraint equation is

$$(v_1 + L_1.\alpha_1)^2 + (u_1 + L_1.\beta_1)^2 \leq (t_1)^2 \quad (8)$$

where,

L_1	feature of size (FOS) related to AFR in *x*-axis
t_1	tolerance value fixed by the designer for AFR
$v_1, \alpha_1, u_1, \beta_1$	key parameters assigned a value in the range of 0.1 mm to 0.00001 mm.

Theoretically, an assembly may have *n* number of AFR constraints, but computational time to be considered.

4.2.2 Stack-up constraints

An important step in tolerance design is to estimate the accumulated tolerance on the assembly dimension, for a given set of tolerances associated with individual part dimensions. This step is generally known as tolerances analysis. The accumulated tolerance on the assembly dimension(s) must be equal to or less than the corresponding assembly tolerances specified by the designer based on the functionality and assembly-ability requirements. Different methods used for establishing such relationships, called stack-up conditions, have been proposed over the years. They are rotational and translational constraints.

Rotational stack-up constraint equations From Fig. 2 and consequent discussion rotational constraint is the accumulated rotational tolerance in the assembly on the *x*-axis, *y*-axis, and *z*-axis.

Rotational movements (R_x, R_y, R_z)=key parameters (α_i,β_i,γ_i) for rotation along the *x*-axis, *y*-axis, *z*-axis.

Using the rotational key parameters (α_i,β_i,γ_i) along the *x*-axis, *y*-axis, and *z*-axis, the rotational constraint equations are formulated to respective axis.

$$\alpha_0 = \sum_{i=1}^{n} \alpha_i \quad (9)$$

α_0 is the cumulative rotational stack-up for α_i (i=1,2,3…n) along the *x*-axis.

$$\beta_0 = \sum_{i=1}^{n} \beta_i \quad (10)$$

β_0 is the cumulative rotational stack-up for β_i (i=1,2,3…n) along the *y*-axis.

$$\gamma_0 = \sum_{i=1}^{n} \gamma_i \quad (11)$$

γ_0 is the cumulative rotational stack-up for γ_i (i=1,2,3…n) along the *z*-axis

Translational stack-up constraint equations Translational stack-up is a state of constraint for an assembly at which minute and cumulative build up of deviation along the *x*-, *y*-, and *z*-axis. Hu and Xiong [4] proposed the translational stack-up as follows:

$$u_0 = \sum_{i=1}^{n} u_i - \sum_{i=1}^{n} \gamma_i Y_i + \sum_{i=1}^{n} \beta_i Z_i \quad (12)$$

u_0 is the translational constraint along *x*-axis.

$$v_0 = \sum_{i=1}^{n} v_i - \sum_{i=1}^{n} \gamma_i X_i + \sum_{i=1}^{n} \alpha_i Z_i \quad (13)$$

v_0 is the translational constraint along *y*-axis.

$$w_0 = \sum_{i=1}^{n} w_i - \sum_{i=1}^{n} \beta_i X_i + \sum_{i=1}^{n} \alpha_i Y_i \quad (14)$$

w_0 is the translational constraint along *z*-axis.

4.2.3 Tolerance constraints

The tolerance constraints can be regarded as limits of feature variation. The tolerance constraint in this study is based on the analysis of tolerance zones. Khodaygan et al. [7] discussed all kinds of tolerance zones. Those tolerance zones can be summarized as typical types, as shown in Fig. 5. The size of the tolerance zone is usually 10^{-3} to 10^{-5} mm of the feature size. In the figure, the tolerance zone is exaggerated for illustration. *t* represents tolerance value. There are three typical tolerance zones:

1. 1D tolerance zones.
2. 2D tolerance zones.
3. 3D tolerance zones.

Dimensional tolerance zones belong to type 1. Types 2 and 3 refer to geometrical tolerance zones. In the Cartesian coordinate system, 3D tolerance zones can be projected onto 2D tolerance zones, and 2D zones onto 1D zones, as shown in Fig. 5.

The part which shown in Fig. 1 is a design; there exist certain deviation when observed practically. Srinivasan [23] studied this tolerance zone theory and presented a geometrical product specification language for computer-aided tolerancing. The representation used in the algorithm is based on the study of variational model using key matrix.

Overview of tolerance constraints Since tolerance constraint associated with the tolerance zone is important, a brief overview is presented here.

Figure 3a represents the design part, Fig. 3b is the resultant geometry produced as per design on a machine; this is a major concern which need to be addressed during geometric tolerance design. As per ASME Y14.5M – 1994 [1] such resultant geometry can be brought under a boundary called AME, is also called as perfect feature counterpart and is defined separately for external and internal features. AME for an external feature is the smallest, similar, perfect, feature counterpart that can be circumscribed around the feature so that it just contacts the surface(s) at the highest points. Figure 3c shows the envelope for the part, for better understanding an example is revealed in Fig. 4, the actual mating envelope of a pin is the smallest precision sleeve that just fits over the pin contacting the surface at the highest points. AME for an internal feature is the largest, similar, perfect, feature counterpart that can be inscribed within the feature so that it just contacts the surface(s) at the highest points. For example, the actual mating envelope of a hole is the largest precision pin that just fits inside the hole contacting the surface at the highest points.

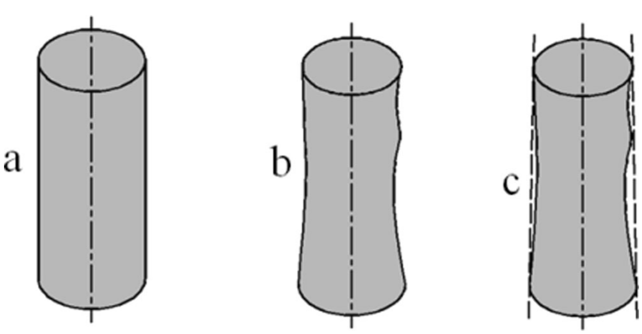

Fig. 3 Feature variables

The reason for opting AME is when GD&T is used obviously representation of material condition is important. Either maximum material condition (MMC) or least material condition (LMC) should be specified if not as per [1] Regardless feature of size (RFS) applies default. To retain the same features as obtained from machining, AME is called [1]. There by the cumbersome task of material condition is overcome.

Tolerance constraint equations Mathematical models are framed for the features considering their AME. Any complicated AME can be derived by using the following three elements [23]: point (PT), straight lines (SL), and plane (PL) and the design features are classified into seven classes: spherical, cylindrical, planar, helical, revolute, prismatic, and complex. Refer Table 1 for feature classification and their relevance to the elements and DOF.

Key matrix τ_i has been framed with respective translation and rotational vectors and shown in Table 2.

Khodaygan et al. [7] developed models for 2D and 3D tolerance zones with the above discussed rotational and translational elements. Refer Fig. 5 for the tolerance zone and relevant constraints.

From Fig. 5, appropriate tolerance zone constraint may be chosen and used as a constraint in the optimization problem.

5 Optimization method

In this section, two evolutionary optimization techniques are discussed and they are used for obtaining Pareto optimal trade-offs.

5.1 Elitist nondominated sorting genetic algorithm (NSGA-II)

Deb proposed the NSGA-II algorithm [24]. It is the advanced version which overcomes the drawbacks of genetic algorithm and NSGA. A fast nondominated sorting procedure is implemented by sorting the individuals of a given population according to the level of nondomination. NSGA-II performs a clever sorting strategy and implements elitism for multiobjective search, using an elitism-preserving approach. Elitism is introduced storing all nondominated solutions discovered so far, beginning from the initial population. Elitism enhances the convergence properties towards the true Pareto-optimal set.

Essentially, NSGA-II differs from the nondominated sorting genetic algorithm (NSGA) implementation in a number of ways. First, NSGA-II uses an elite-preserving

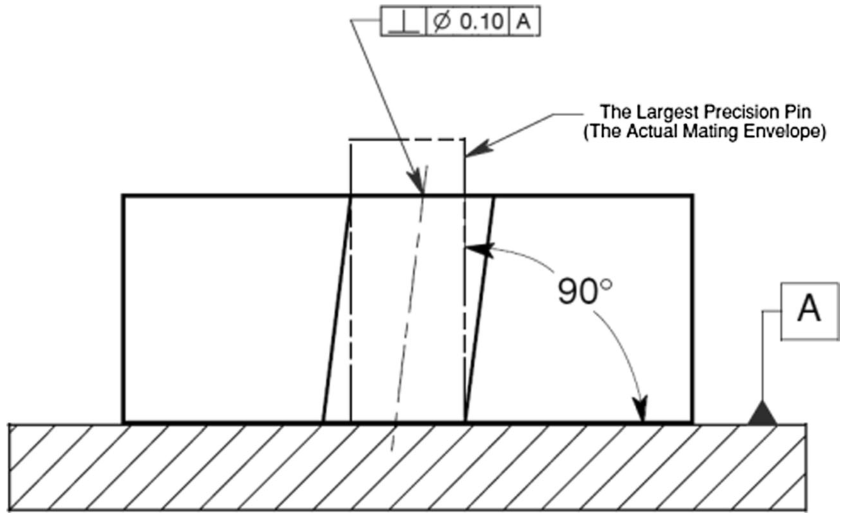

Fig. 4 Actual mate envelope with datum

mechanism, thereby assuring preservation of previously found good solutions. Second, NSGA-II uses a fast nondominated sorting procedure. Third, NSGA-II does not require any tuneable parameter, thereby making the algorithm independent of the user. Initially, a random parent population P_o was created. The population is sorted based on the nondomination. A special book-keeping procedure is used in order to reduce the computational complexity down to O (N2). Each solution is assigned a fitness equal to its nondomination level (1 is the best level). Thus, minimization of fitness is assumed. Binary tournament selection, recombination, and mutation operators are used to create a child population Q_o of size N. Thereafter, we use the following algorithm in every generation. First, a combined population $R_i=P_iUQ_i$ is formed. This allows parent solutions to be compared with the child population, thereby ensuring elitism. The population R_i is of size 2N. Then, the population R_i is sorted according to nondomination. The new parent population P_{i+1} is formed by adding solutions from the first front and continuing to other fronts successively till the size exceeds N. Thereafter, the solutions of the last accepted front are sorted according to a crowded comparison criterion and the first N points are picked. Since the diversity among the solutions is important, we use a partial order relation$\geq n$ as follows:

$$i\geq_n j \text{ if} (i_{\text{rank}} < j_{\text{rank}}) \text{ or } (i_{\text{rank}} = j_{\text{rank}}) \text{ and } (i_{\text{fitness}} > j_{\text{fitness}})$$

That is, between two solutions with differing nondomination ranks, we prefer the point with the lower rank. Otherwise, if both points belong to the same front, then we prefer the point which is located in a region with a lesser number of points (or with a larger crowded distance). This way solution from less dense regions in the search space is given importance in deciding which solutions to choose from R_i. This constructs the population P_{i+1}. This population of size N is now used for selection, crossover, and mutation to create a new population Q_{i+1} of size N. We use a binary tournament selection operator but the selection criterion is now based on the crowded comparison operator$\geq n$. The above procedure is continued for a specified number of generations. It is clear from the above description that NSGA-II uses: (1) a faster nondominated sorting approach, (2) an elitist strategy, and (3) no niching parameter. Diversity is preserved by the use of crowded comparison criterion in the tournament selection and in the phase of population reduction. NSGA-II has been shown to outperform other current elitist multiobjective evolutionary algorithms on a number of difficult test problems. The flowchart in Fig. 6 shows an iteration of the NSGA-II procedure.

The features of NSGA-II are the following:

1. Allows both continuous and discrete variables,
2. User defined discretization,
3. The constraint handling method does not make use of penalty parameters,
4. Implements elitism in multiobjective search,
5. Diversity and spread of solutions is guaranteed without use of sharing parameters,
6. Allows concurrent evaluation of the independent individuals.

Table 1 Feature classification [15]

Feature	Element	DOF	
		Translations	Rotations
Spherical	PT	3	–
Cylindrical	SL	2	2
Planar	PL	1	2
Helical	(PT, SL)	2	2
Revolute	(PT, SL)	3	2
Prismatic	(SL, PL)	2	3
Complex	(PT, SL, PL)	3	3

Table 2 Feature key matrix

Feature	Element	DOF		Key matrix	
		Translations	Rotations	ω_i	ε_i
Spherical	PT	3	–	$(u_i, v_i, w_i)^T$	$(0, 0, 0)^T$
Cylindrical	SL	2	2	$(0, v_i, w_i)^T$	$(0, \beta_i, \gamma_i)^T$
Planar	PL	1	2	$(0, 0, w_i)^T$	$(\alpha_i, \beta_i, 0)^T$
Helical	(PT, SL)	2	2	$(0, v_i, w_i)^T$	$(0, \beta_i, \gamma_i)^T$
Revolute	(PT, SL)	3	2	$(u_i, v_i, w_i)^T$	$(0, \beta_i, \gamma_i)^T$
Prismatic	(SL, PL)	2	3	$(0, v_i, w_i)^T$	$(\alpha_i, \beta_i, \gamma_i)^T$
Complex	(PT, SL, PL)	3	3	$(u_i, v_i, w_i)^T$	$(\alpha_i, \beta_i, \gamma_i)^T$

5.1.1 NSGA-II parameters

The user must specify the following NSGA-II parameters:

1. Population size, 100
2. No. of iteration, 100
3. Crossover probability, 0.7
4. Mutation probability, 0.6
5. Distribution index for crossover, 10
6. Distribution index for mutation, 100

5.2 Differential evolution

5.2.1 Brief introduction to DE

Differential evolution (DE) was introduced by Price and Storn [25] and is a branch of evolutionary algorithms for optimization problems over continuous domains. In DE, the value of each variable in the chromosome is represented by a real number. DE can be categorized into a class of floating-point-encoded evolutionary algorithms. The theoretical framework

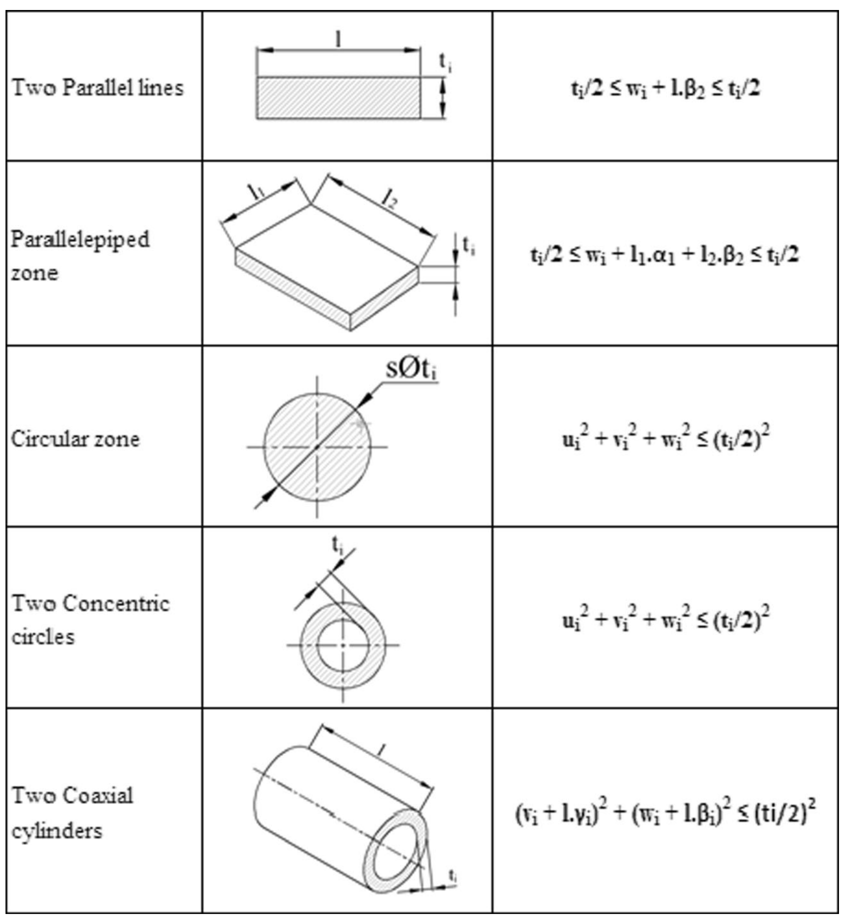

Fig. 5 Tolerance constraints with various zones

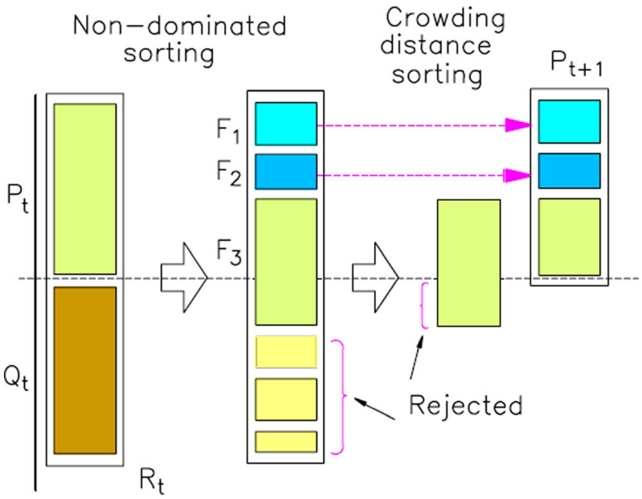

Fig. 6 An iteration of the NSGA-II procedure

of DE is very simple and DE is computationally inexpensive in terms of memory requirements and CPU times. Thus, nowadays DE has gained much attention and wide application in a variety of fields. DE starts with the random initialization of a population of individuals in the search space and works on the cooperative behaviors of the individuals in the population. It finds the global optima by utilizing the distance and direction information according to the differentiations among the population. However, the searching behavior of each individual is adjusted by dynamically altering the differentiation's direction and step length. At each generation, the mutation and crossover operators are applied to individuals to generate a new population. Then, selection takes place and the population is updated. Let the ith individual in the N-dimensional search space at generation t be

$$X_i(t) = \left[x_{i,1}, x_{i,2}, \ldots . x_{i,n}\right](i = 1, 2, \ldots .., m)$$

Here, M denotes the size of the population. The Des basic scheme, which is denoted as DE/rand/1/bin, can be described as follows. For each target individual X_i (t), according to the mutation operator, a mutant vector

$$V_i(t+1) = \left[V_{i,1}(t+1), \ldots\ldots\ldots V_{i,N}(t+1)\right] \quad (15)$$

is generated by adding the weighted difference between a defined number of individuals randomly selected from the previous population to another individual, which is described by the following equation:

$$V_i(t+1) = X_{r1}(t) + F^*[X_{r2}(t) - X_{r3}(t)] \quad (16)$$

Here, $r1$; $r2$; $r3 \in \{1, 2, \ldots M\}$ are randomly chosen and mutually different and also different from the current index i, $F \in (0, 2)$ is a constant called the scaling factor which controls amplification of the differential variation $X_{r2}(t) - X_{r3}(t)$, and M is greater than or equal to 4 so that the mutation can be applied. $X_{r1}(t)$ is the base vector to be perturbed. Following the mutation phase, the crossover operator is applied to increase the diversity of the population. Thus, for each target individual $X_i(t)$, a trial vector $U_i(t+1)=[U_{i,1}(t+1), \ldots, U_{i,N}(t+1)]$ is generated by the following equation:

$$u_{i,j}(t+1) = f(x) = \begin{cases} v_{i,j}(t+1), \textit{if}\,(\text{rand}(j) \le \text{CR or}\, j = \text{randn}\,(i), j = 1, \ldots\ldots N \\ x_{i,j}(t+1)\,, \text{otherwise} \end{cases} \quad (17)$$

Where rand (j) is the jth evaluation of a random number uniformly distributed in the range [0, 1], randn (i) is a randomly chosen index from the set $\{1,2,\ldots N\}$, CR [0,1] is a constant called the crossover parameter that controls the diversity of the population. After the crossover operation, the selection arises to determine whether the trial vector $U_i(t+1)$ would be a member of the population of the next generation $t+1$. For a minimization problem, $U_i(t+1)$ is compared to the initial target individual $X_i(t)$ by the following one-to-one greedy selection criterion:

$$X_i(t+1) = f(x) = \begin{cases} U_{i,j}(t+1), \textit{if}\, U_{i,j}(t+1) < f(X_i(t)) \\ X_i(t), \text{otherwise} \end{cases} \quad (18)$$

Here, f is the objective function and X_i ($t+1$) is the individual of the new population. The key parameters in DE are M (size of population), F (scaling factor), and CR (crossover parameter). Proper configuration of the above parameters would increase the convergence velocity and robustness of the search process. Currently, several variants of DE have been proposed depending on the selection of the base vector to be perturbed, the number and selection of the differentiation vectors, and the type of crossover operators.

5.2.2 DE parameters

The following DE algorithm parameters can be used to solve any problem

1. Strategy, DE/rand/1/bin
2. Choice of strategy, 7
3. Maximum generations, 100
4. Output refresh cycle, 2
5. Population size, 100
6. Weight factor, 0.45
7. Crossing over factor, 0.9

5.2.3 DE implementation

The basic procedure of DE is summarized as follows.

Step 1 Randomly initialize the population of individuals for DE.
Step 2 Evaluate the objective values of all individuals, and determine the best individual best fit, which has the best objective value.
Step 3 Perform mutation operation for each individual according to Eq. 16 in order to obtain each individual's corresponding mutant vector.
Step 4 Perform crossover operation between each individual and its corresponding mutant vector according to Eq. 17 in order to obtain each individual's trial vector.
Step 5 Evaluate the objective values of the trial vectors.
Step 6 Perform selection operation between each individual and its corresponding trial vector according to Eq. 18 so as to generate the new individual for the next generation.
Step 7 Determine the best individual of the current new population with the best objective value. If the objective value of the current best individual is better than that of best fit, then update best fit and its objective value.
Step 8 If a stopping criterion is met, then output is best fit and its objective value; otherwise go back to step 3.

6 Sensitive factor

Tolerance design need to be economical: minimize product cost, improve quality, and reducing overall cost. Tolerance analysis assesses the assembly tolerance from the known component tolerance. Assembly tolerance is critical to satisfy product performance and manufacturing cost reduction. Tolerance analysis finds whether to what extent the assembly tolerance exceeds the specification and its component tolerance to be reduced. Usually it is done by sensitivity analysis. Tolerance sensitivity study is associated to tolerance analysis and is very important for tolerance design improvement by assessing the influence of individual component tolerance on assembly tolerance. This influence is depicted as sensitive factor. Zhang wu [19] proposed the study and proved that components with greater sensitivity factor deserve more attention in tolerance design improvement.

Sensitive factor S_i of the ith component dimension or tolerance is

$$S_i = \left| \frac{D_i - D_0}{\Delta d_i} \right| i = 1, 2, \ldots..n \tag{19}$$

where,

$$D_i = D_i(d_{01}, d_{02}, \ldots.d_{0i} + \Delta di, \ldots..d_{0n}) \tag{20}$$

D_i is the instance of the assembly dimension value,
D_{0n} is the nominal dimension
ΔD_i is the instance of the assembly dimension value,

7 Case study

To illustrate the application of the proposed framework for optimal geometric design a simple assembly [4] is shown in Fig. 7 and detail drawings are shown in Fig. 8.

The assembly in Fig. 7 is to be assembled with an AFR of 0.017-mm gap among the corresponding mating elements. The proposed approach is discussed in this chapter and the drawings follow first angle projection with the millimeter unit.

Step 1 Identify the principal feature of the assembly elements with its AFR

From the problem, it is clear that a gap of 0.017 mm is expected and is fixed has the AFR for this problem.

$$\text{Hence AFR} = 0.017 \text{ mm}$$

Step 2 Allocate various geometrical tolerance symbols and zones as required

Now the critical task is to allocate geometric tolerance symbols. On examining the parts in Fig. 8, the features are decomposed and individual feature numbers are allocated. They are represented by the letter *t* followed by a numeral refer Fig. 9.

Appropriate geometric symbols were adopted for these individual features by analyzing their functional requirements.

*t*1: Diameter concerning to achieve the tolerance (±)
*t*2: concern with the geometry. With respect to the function concentricity cannot be selected because of manufacturing difficulty since no opposing diameter to measure. Similarly, a designer may reiterate for the related geometric tolerance. But they need to consider the manufacturing aspects. Now for this feature, perpendicularity may hold good.
*t*3: Position of the feature is important for the assembly between the holes. Hence, position tolerance. Here, bonus tolerance is not applicable because AME is assumed.

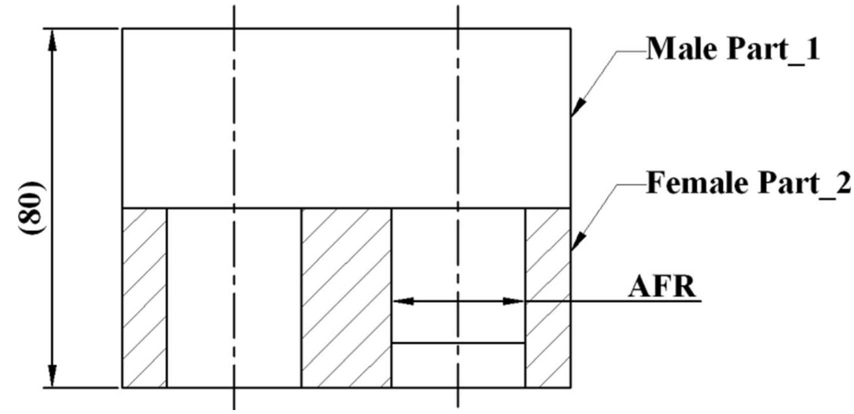

Fig. 7 Simple assembly with AFR

Similar work out is carried for the left-out features and the same represented as GD&T data sheet in Table 3. The allocated tolerances only need not be called, it is an option and opinion differs and geometric symbol may be called as per requirement.

GD&T data sheet will be useful in establishing the manufacturing cost function models, tolerance constraints, and preparation of shop floor drawings.

Step 3 Establish a mathematical expression for the objective function

Mathematical expression is essential for minimizing the relative manufacturing cost to produce that tolerance. Hence, i ranges from 1 to 12 (since 12 tolerances) and expressed as

$$\text{Minimize}: \left[\sum_{i=1}^{12}\sum_{j=1}^{4} C_j(t_i)\right]$$

Now, using the GD&T data sheet, the model numbers of c is represented in the expression as

$$\begin{aligned}\text{Minimize} = {} & c_1(t_1) + c_4(t_2) + c_3(t_3) + c_4(t_4) + c_3(t_5) \\ & + c_1(t_6) + c_2(t_7) + c_4(t_8) + c_3(t_9) + c_4(t_{10}) \\ & + c_3(t_{11}) + c_2(t_{12})\end{aligned}$$

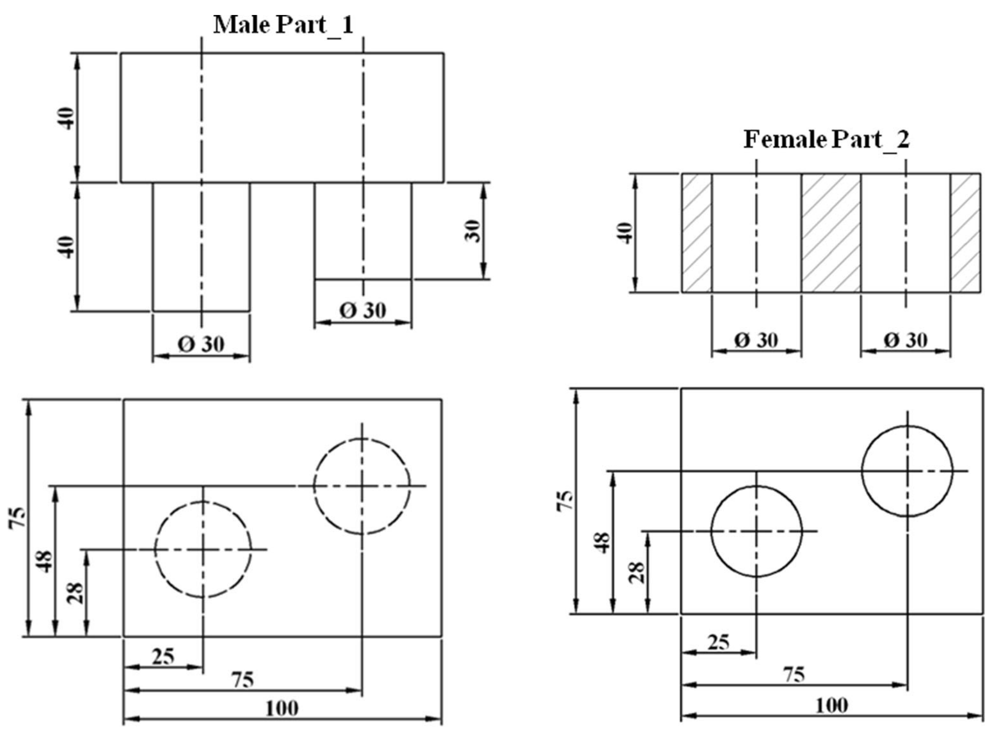

Fig. 8 Detail drawing for male and female part with dimension

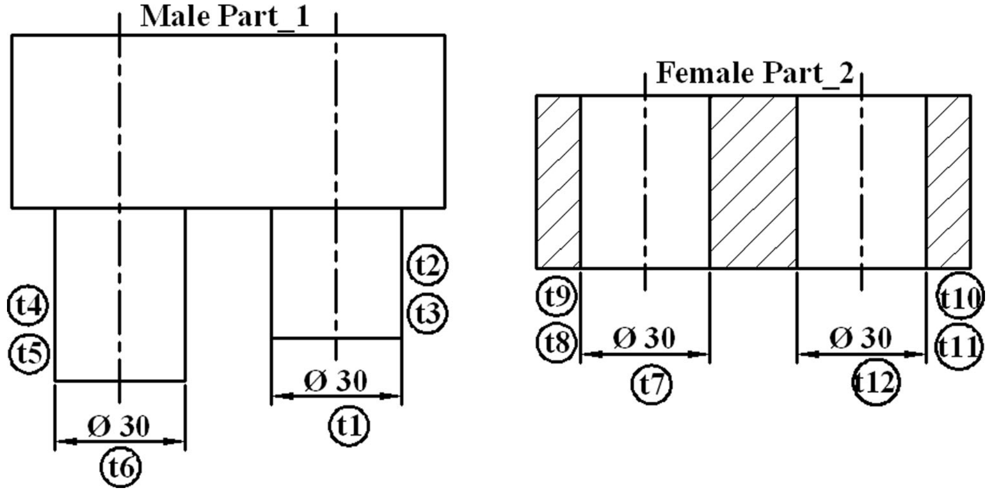

Fig. 9 Detail drawing numbered with features

The objective function is made as a combinatorial function by including the relevant $\underline{c}$ models as obtained from Eqs. 2 to 5. Hence, combinatorial equations of this objective function is

$$
\begin{aligned}
C_1(t_1) &= 10^{-5} + 10^{-5}\,t_1 + 67.3\,e^{-2.59t_1}\\
C_4(t_2) &= 5.425 + 10^{-5}\,t_2 + 12.43\,e^{-10.82t_2}\\
C_3(t_3) &= 8.052 + 10^{-5}\,t_3 + 30.87\,e^{-12.09t_3}\\
C_4(t_4) &= 5.425 + 10^{-5}\,t_4 + 12.43\,e^{-10.82t_4}\\
C_3(t_5) &= 8.052 + 10^{-5}\,t_5 + 30.87\,e^{-12.09t_5}\\
C_1(t_6) &= 10^{-5} + 10^{-5}\,t_6 + 67.3\,e^{-2.59t_6}\\
C_2(t_7) &= 10^{-5} + 10^{-5}\,t_7 + 57.6\,e^{-1.59t_7}\\
C_4(t_8) &= 5.425 + 10^{-5}\,t_8 + 12.43\,e^{-10.82t_8}\\
C_3(t_9) &= 8.052 + 10^{-5}\,t_9 + 30.87\,e^{-12.09t_9}\\
C_4(t_{10}) &= 5.425 + 10^{-5}t_{10} + 12.43e^{-10.82t_{10}}\\
C_3(t_{11}) &= 8.052 + 10^{-5}t_{11} + 30.87e^{-12.09t_{11}}\\
C_2(t_{12}) &= 10^{-5} + 10^{-5}\,t_{12} + 57.6\,e^{-1.59t_{12}}
\end{aligned}
$$

Step 4 Establish the equation for AFR and use it as a constraint

By using Eq. 8, the AFR can be expressed. From step 1, AFR is 0.017 mm. Therefore

$$
\begin{aligned}
&(v_0 + L.\alpha_0)^2 + (u_0 + L.\beta_0)^2 \le (\mathrm{AFR})^2\\
&(v_0 + 30.\alpha_0)^2 + (u_0 + 30.\beta_0)^2 \le (0.017)^2\\
&g_0 = (0.017)^2 - \left[(v_0 + 30.\alpha_0)^2 + (u_0 + 30.\beta_0)^2\right] \ge 0
\end{aligned}
$$

Step 5 Establish the 3D rotational and translational stack-up constraints

Rotational stack-up constraints: the problem deals with *x*- and *y*-axis since *z*-axis will be arrested by assembly. Refer Eqs. 9–11

$$
\begin{aligned}
\alpha_0 &= \alpha_2 + \alpha_4 + \alpha_8 + \alpha_{10}\\
\beta_0 &= \beta_2 + \beta_4 + \beta_8 + \beta_{10}
\end{aligned}
$$

Table 3 GD&T data sheet

Tolerance variables	GDT feature	Model
t_1	±	1
t_2	Perpendicularity	4
t_3	Position	3
t_4	Perpendicularity	4
t_5	Position	3
t_6	±	1
t_7	±	2
t_8	Perpendicularity	4
t_9	Position	3
t_{10}	Perpendicularity	4
t_{11}	Position	3
t_{12}	±	2

Table 4 Optimized results from NSGA-II and DE

Tolerance values	NSGA-II	DE
t_1	0.017214	0.06584
t_2	0.013081	0.09758
t_3	0.015707	0.05471
t_4	0.017851	0.04571
t_5	0.019972	0.06206
t_6	0.014105	0.02423
t_7	0.015027	0.08152
t_8	0.01168	0.06671
t_9	0.019467	0.07474
t_{10}	0.018769	0.07184
t_{11}	0.016844	0.06816
t_{12}	0.016751	0.02065
Objective function	436.97879 Cr	360.415 Cr

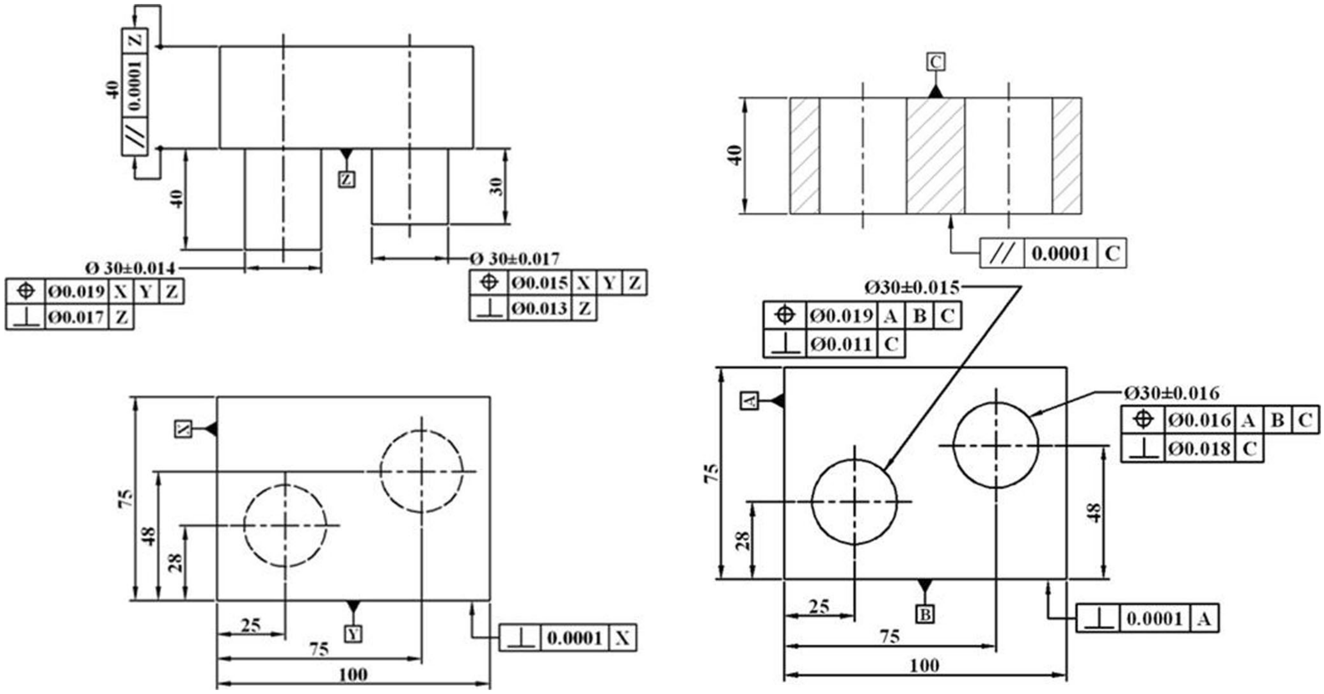

Fig. 10 GDT drawing with NSGA-II results

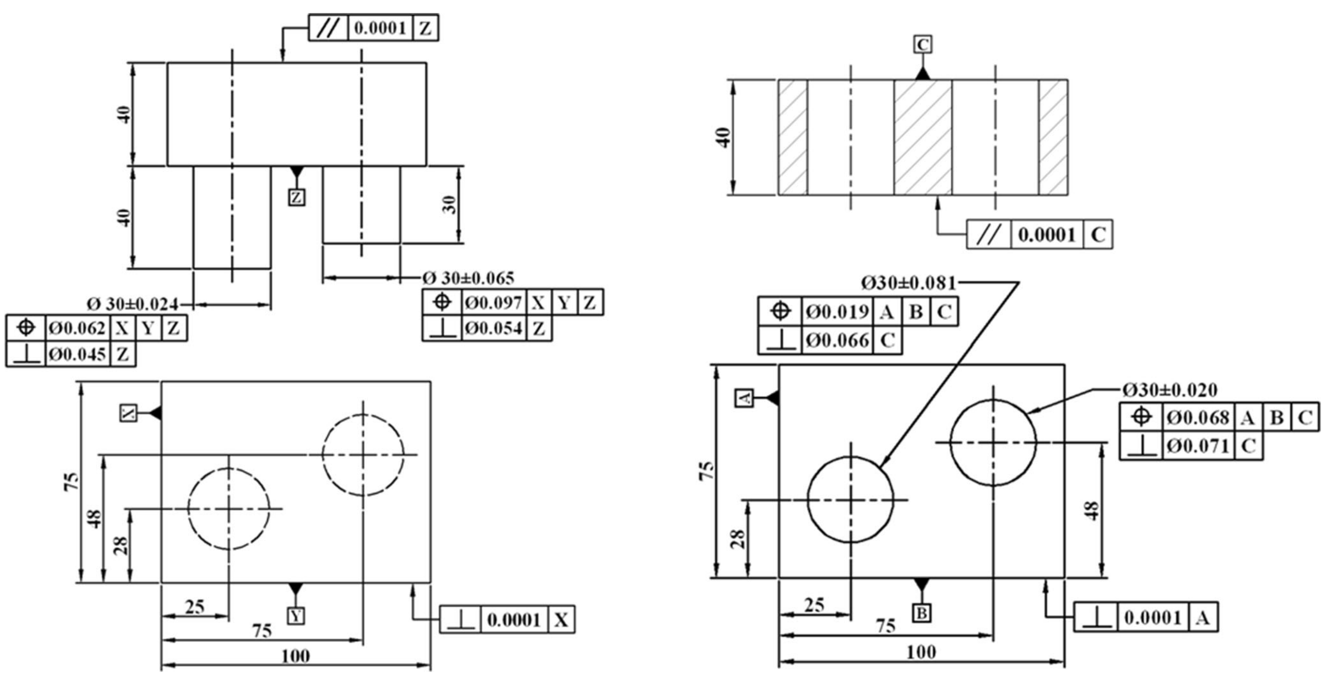

Fig. 11 GDT drawing with DE results

Table 5 Comparison of sensitive factors

Description	Sensitive factor	Sensitivity
Hole 1 in *x*-axis	0.928505	Severe
Hole 1 in *y*-axis	0.3714	Medium
Hole 2 position tolerance	0.1134	Low

Translational stack-up constraints: Similarly refer Eqs. 12–14

$$u_0 = u_1 + u_3 + u_5 + u_6 + u_7 + u_9 + u_{11} + u_{12} + 30\beta_2 + 30\beta_4 + 30\beta_8 + 30\beta_{10}$$

$$v_0 = v_1 + v_3 + v_5 + v_6 + v_7 + v_9 + v_{11} + v_{12} - 30\alpha_2 - 30\alpha_4 - 30\alpha_8 - 30\alpha_{10}$$

Step 6 Institute the tolerance constraint by identifying the tolerance zone

Referring Fig. 5, tolerance constraints are established with respect to their tolerance zones as called by the geometrical symbol.

$$\begin{aligned}
g_1 &= (t_1/2)^2 - \left(u_1{}^2 + v_1{}^2\right) \geq 0 \\
g_2 &= (t_2/2)^2 - \left[(30\alpha_2)^2 + (30\beta_2)^2\right] \geq 0 \\
g_3 &= (t_3/2)^2 - \left(u_3{}^2 + v_3{}^2\right) \geq 0 \\
g_4 &= (t_4/2)^2 - \left[(40\alpha_4)^2 + (40\beta_4)^2\right] \geq 0 \\
g_5 &= (t_5/2)^2 - \left(u_5{}^2 + v_5{}^2\right) \geq 0 \\
g_6 &= (t_6/2)^2 - \left(u_6{}^2 + v_6{}^2\right) \geq 0 \\
g_7 &= (t_7/2)^2 - \left(u_7{}^2 + v_7{}^2\right) \geq 0 \\
g_8 &= (t_8/2)^2 - \left[(40\alpha_8)^2 + (40\beta_8)^2\right] \geq 0 \\
g_9 &= (t_9/2)^2 - \left(u_9{}^2 + v_9{}^2\right) \geq 0 \\
g_{10} &= (t_{10}/2)^2 - \left[(30\alpha_{10})^2 + (30\beta_{10})^2\right] \geq 0 \\
g_{11} &= (t_{11}/2)^2 - \left(u_{11}{}^2 + v_{11}{}^2\right) \geq 0 \\
g_{12} &= (t_{12}/2)^2 - \left(u_{12}{}^2 + v_{12}{}^2\right) \geq 0
\end{aligned}$$

Step 7 Optimize the nonlinear combinatorial problem by NSGA-II and DE

Now, the combinatorial optimization problem to be solved by subjecting the AFR, stack-up and tolerance constraints and the optimized tolerance values of the 12 features need to be obtained.

Table 6 Comparison of the optimized results from NSGA-II and DE

Tolerance values	Hu and Xiong [4]	NSGA-II	DE
t_1	0.013	0.017214	0.06584
t_2	0.012	0.013081	0.09758
t_3	0.01	0.015707	0.05471
t_4	0.009	0.017851	0.04571
t_5	0.014	0.019972	0.06206
t_6	0.013	0.014105	0.02423
t_7	0.01	0.015027	0.08152
t_8	0.018	0.01168	0.06671
t_9	0.014	0.019467	0.07474
t_{10}	0.01	0.018769	0.07184
t_{11}	0.009	0.016844	0.06816
t_{12}	0.006	0.016751	0.02065
Objective function	448.5 Cr	436.97879 Cr	360.415 Cr
Percent of improvement	NA	3 %	20 %

Initially, the tolerance limits should be specified and is the discretion of the designer. For this problem, the following limits were used:

$$\begin{aligned}
&0.001 < t_1 < 0.01 \\
&0.001 < t_2 < 0.01 \\
&0.001 < t_3 < 0.01 \\
&0.0001 < t_4 < 0.01 \\
&0.001 < t_5 < 0.01 \\
&0.001 < t_6 < 0.01 \\
&0.005 < t_7 < 0.015 \\
&0.016 < t_8 < 0.02 \\
&0.01 < t_9 < 0.02 \\
&0.008 < t_{10} < 0.016 \\
&0.005 < t_{11} < 0.012 \\
&0.004 < t_{12} < 0.014
\end{aligned}$$

In addition to these limits, the problem is solved by using data's from "NSGA-II parameters" and "DE parameters." The problem was solved using Microsoft Visual C++.

Step 7 Observe the results and update in the drawings

The following results were obtained for the problem and shown in Table 4.

Here, Cr is the reference cost and these are used to prepare the shop floor drawings. The drawings are shown in Figs. 10 and 11.

Step 9 Calculation of sensitive factors for position tolerances

The result from DE algorithm is taken to demonstrate the sensitive factor for this numerical example.

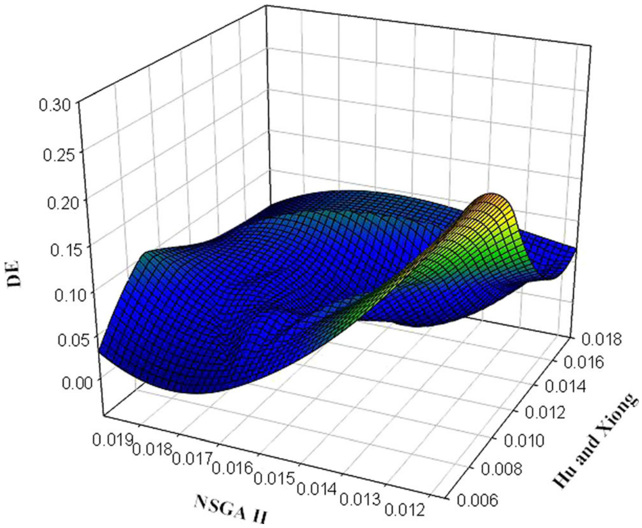

Fig. 12 Comparison graph for optimized results

The main reason to calculate the sensitive factor is to enable the designer to go for tolerance design improvement if the design fails to perform its function.

For the drawing shown in Fig. 10, Female part_2 is considered for sensitive factor calculation. There are two holes in it. The coordinates of the first hole is (x_1=25, y_1=28) consider it as datum and for the second hole is (x_2=75, y_2=48). The aligned distance between the holes $D0$ is calculated by the following"

$$D_0 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$
$$D_0 = 53.85 \text{ mm}$$

$$S_1 = \frac{x_2 - x_1}{D_0} = 0.928505$$

$$S_2 = \frac{y_2 - y_1}{D_0} = 0.37140$$

$$S_3(\alpha_0) = \left| \frac{D_3 - D_0}{\Delta d_i} \right| = 0.1134$$

The results of the sensitive factors are summarized in Table 5.

The inference from the sensitivity is if the assembly fails, then the datum hole 1 is to be modified. The rule of thumb from the sensitivity is datum should be made equidistant from all the features for optimum sensitivity.

8 Discussion

For the same optimization problem, the results vary significantly with algorithms and justification of the use of right algorithm is established as per Mohamed [5]. The proposed approach is compared with Hu and Xiong [4] and found with considerable improvement and shown in Table 6.

The reference cost has decreased by 3 % for NSGA-II and 20 % for DE. Hence, DE is superior to NSGA-II. The result of comparison is shown in Fig. 12.

Further, the results are analyzed to understand their operating effectiveness. Since the selection of off-spring population is very important for obtaining the global optimum point. The effectiveness of an algorithm must not be evaluated based on the results and off-spring or Pareto optimal trade-offs must be considered [24]. Hence, the objective value obtained in 100 iterations are plotted on a graph and illustrated in Fig. 13. With these analysis, DE is most opt for a geometric tolerance design problem.

Fig. 13 Comparison of values in all iterations

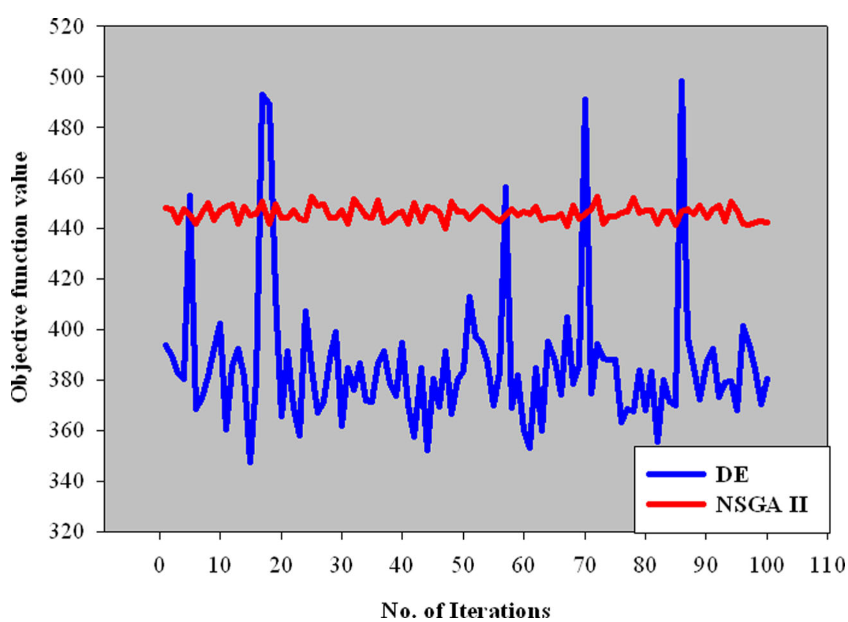

Table 7 Comparison of various frameworks

Articles	Scheme	Representation	AFR Specification	Analysis	Synthesis	Transfer	Sensitivity	Optimization
Prabhaharan et al. [2]	a	a		b		b		a
Hu and Xiong [4]	a	a	b					a
Mohamed [5]	a	a		b				a
Zhang [6]	a	a		a				
Khodaygan [7]	a	a		b	b			
Robin and Bernard [12]	a	a		a	a			
Muthu [13]	a	a		b				a
Proposed framework	a	a	a	a	b	b	a	a

a detailed, *b* semi-detailed

9 Conclusion

A method for optimal determination of geometrical tolerances with AFR and sensitive factor has been presented with a case study. Conventionally, quality and manufacturing costs are indirectly proportional to each other. But, as shown in this paper, incorporating geometric tolerance control reduces the total cost and at the same time increases interchangeability and reliability. The method, generally termed as geometric tolerance framework with AFR is well suited for engineering environment, where high quality products are designed and manufactured. Once a Visual C++ program is developed, large extent of time can be saved and quiet suitable for complex assemblies with any number of AFR. The comparison of the model with that of existing is shown in Table 7.

The proposed framework posses several advantages over others and can be readily adapted to all design problems with GD&T problems. The advantages are as follows:

1. Establishment of AFR and AFR-based geometrical tolerance values,
2. Eliminates the need for various intermediate elements like cost design tolerance functions,
3. Facilitate tolerance design improvement if design fails,
4. Improved computability and making the model easier to understand by design and manufacturing engineers.
5. Availability of 3D tolerance analysis and synthesis,
6. Enhanced application of evolutionary algorithms,
7. Ability to address tolerance transfer issues and
8. Better framework which utilizes every possible constraints from design to manufacture.

Though the framework has addressed several issues with advantages, still the following limitations hold on:

1. Applicable only for rigid parts and not for sheet metals,
2. Material condition is assumed as AME,
3. Designer should have the machine's achievable least tolerance value and
4. Composite position tolerance cannot be applied.
5. Quality loss function is not considered.

These limitations open doors for further investigation and future paper will need to consider these limitations and a multi objective optimization method to be framed and solved by an effective algorithm.

References

1. ASME Y14.5M-1994 (1994) Dimensioning and tolerancing. The American Society of Mechanical Engineers, New York
2. Prabhaharan G, Ramesh R, Asokan P (2007) Concurrent optimization of assembly tolerances for quality with position control using scatter search approach. Int J Prod Res 45(21):4959–4988
3. Ngoi BKA, Min OJ (1999) Optimum tolerance allocation in assembly. Int J Adv Manuf Technol 15:660–665
4. Hu J, Xiong G (2005) Dimensional and geometric tolerance design based on constraints. Int J Adv Manuf Technol 26:1099–1108
5. Mohamed HG (2011) Tolerance optimization problem using a near-to-global optimum. Int J Exp Des Proc Optim 2(4):318–335
6. Zhang Y, Zongbin L, Jianmin G, Jun H (2011) New reasoning algorithm for assembly tolerance specifications and corresponding tolerance zone types. Comp Aid Des. doi:10.1016/j.cad.2011.06.008
7. Khodaygan S, Movahhedy MR, Fomani MS (2010) Tolerance analysis of mechanical assemblies based on modal interval and small degrees of freedom (MISDOF) concepts. Int J Adv Manuf Technol 50:1041–1061
8. Bai G, Zhang C, Wang B (2011) Optimization of machining datum selection and machining tolerance allocation with genetic algorithms. Int J Prod Res doi:10.1080/002075400188924 pages 1407–1424
9. Pandya G, Amine LE, Cavalier TM (2010) Tolerance design of datum systems. Int J Prod Res. doi:10.1080/002075401l002101901 pages 783–807
10. Demoly F, Luis T, Eynard B, Dimitris K, Samuel G (2011) Geometric skeleton computation enabling concurrent product engineering and assembly sequence planning. Comp Aid design (article in press) (doi:22.1010/j.cad.2011.09.016)
11. Zbigniew H, Piotr T (2011) Animated visualization of the maximum material requirement. Comp Aid design (article in press) (doi:28.1011/j.cad.2011.09.028)
12. Robin C, Bernard A (2011) Functional tolerancing: virtual material condition on complex junctions. Comp Aid design (article in press) (doi:31.1005/j.cad.2011.11.09)

13. Muthu P, Dhanalakshmi V, Sankaranarayanasamy K (2010) Design and manufacturing tolerances optimization with quality loss functions. Int J Adv Oper Manag 2(1/2):90–107
14. Andrea R, Michele A, Matteo B, Michele L (2011) Fast genetic algorithm for roundness evaluation by the minimum zone tolerance method. Comp Aid design (article in press) (doi:25.1016/j.cad.2011.03.31)
15. Yashpal K, Hemant R, Raj BA, Sam A (2008) Minimum-zone form tolerance evaluation using particle swarm optimization. Int J Intell Syst Technol Appl 4(1/2):79–96
16. Loof J, Soderberg R (2007) Top down decomposition of multi-product requirements onto locator tolerances. Proc. Of ASME IMECE
17. Iannuzzi M, Sandgren E (1994) Optimal tolerancing: the link between design and manufacturing productivity. ASME Design, Theory and Methodology. DTM 94-Vol. 68, pp.29-42
18. Mohamed HG (2011) Least sensitive tolerance allocation. Int J Qual Eng Technol 2(4):344–356
19. Wu Z (1997) Sensitive factor for position tolerance. Res Eng Des 9: 228–234
20. ASME Y14.5.1M-1994. Mathematical definition of dimensioning and tolerancing principles. The American Society of Mechanical Engineers, New York, 1994
21. Jerome D, Denis T (2008) A tolerancing frame work to support geometric specifications traceability. Int J Adv Manuf Technol 36: 894–907
22. Krulikowski A. Tolerance stack—a self study course, Vol. II, 1992. Effective Training Inc.: Westland, Michigan
23. Srinivasan V (1999) A geometrical product specification language based on a classification of symmetry groups. Comput Aided Des 31:659–668
24. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197
25. Price K, Storn R (1997) Differential evolution: a simple evolution strategy for fast optimization. Dr Dobb's J 22(4):18–24 and 78