TOLERANCE SYNTHESIS FOR NONLINEAR SYSTEMS BASED ON SENSITIVITY ANALYSIS

Woo-Jong Lee
Technical Center
Daewoo Motor Co.
Incheon, Korea

Tony C. Woo
and
Shuo-Yan Chou
Department of Industrial & Operations Engineering
University of Michigan
Ann Arbor, MI 48109-2117

Technical Report 89-18

Revised April 1990

«Submitted to IIE Transactions»

Abstract

Automatic assignment of tolerances to dimensioned mechanical assemblies is studied as an optimization problem: the objective of which is to minimize the (manufacturing) cost, subject to the constraints of (design) functionality and (assembly) interchangeability. By associating a nominal dimension and a tolerance to the variance, the probabilistic approach is taken.

Trigonometric functions relating the component geometries give rise to the nonlinearity in the system. Estimating an n-dimensional nonlinear integral by a polytope converts the probabilistic optimization formulation to a deterministic one. It also allows speedy evaluation of tolerance analysis embedded in tolerance synthesis.

Local optimality is ensured by analysis of convexity and quasi-concavity of the objective function and some of the constraints. Sensitivity analysis is performed to provide search directions for global optimality. An implementation is reported with an example.

1. Introduction

Tolerances in an engineering design are intended to capture variations from the ideal, as introduced by the very process of realization such as manufacturing and assembly. Nominal dimensions specify idealized geometries by size, location and form. The range between the upper and lower limits of the variation from the nominal dimension is called *tolerance* [1].

At the design stage, functionality, performance and reliability are the major issues under consideration. Tolerance, or variation from the ideal, should be set to be as close to zero as possible. However, high precision or tight tolerances are usually associated with high costs; looser tolerances are less costly at the manufacturing stage. Yet, at the assembly stage, due to the objective of optimizing the interchangeability of components, tight tolerances are desirable. While design and assembly prefer tight tolerancing, virtual components (from design) must be brought to realization by manufacturing before physical components can be assembled. This requirement results in a three-way trade-off amongst design, manufacturing and assembly as shown in Figure 1.

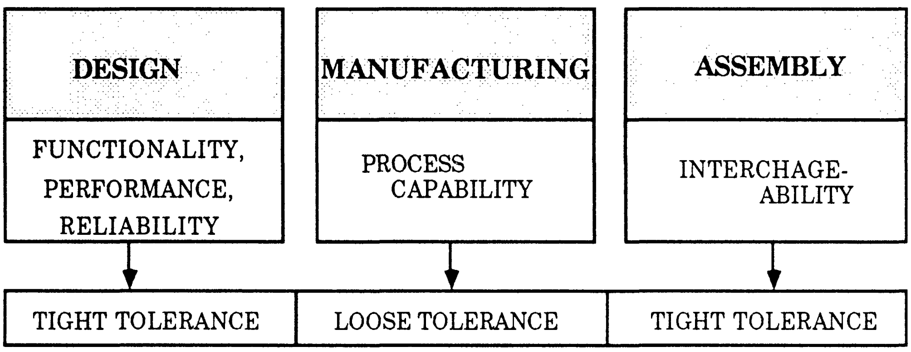


Figure 1. Concurrent Consideration of Tolerance

The concurrent consideration of opposing criteria on tolerancing, between design and manufacturing, and between manufacturing and assembly, can only be resolved by compromise. The result of such a rationalization, if agreeable to all concerns, is effectively a synthesis of tolerances. This paper deals with computational techniques for tolerance synthesis by analysis.

Basic to component manufacturing is cost. To assign cost-effective product tolerances, probabilistic tolerancing is considered in this paper. The probabilistic approach is considered to be advantageous over the deterministic approach, because it is possible to perform trade-off analysis with the probabilistic approach [13,19]. The problem, simply stated, is to convert the designer's "function-oriented" specifications into manufacturable specifications by allocating the tolerances to the ideal dimensions such that the manufacturing cost is minimized. Models of tolerance-cost functions from [10,19,22,23,25] are employed. Figure 2 shows a typical inverse relation of manufacturing cost to tolerances: the tighter the tolerance the higher the cost.

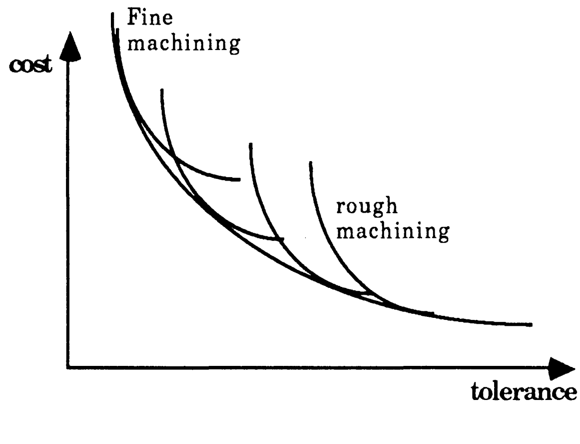


Figure 2. A Typical Tolerance-Cost Curve

To make the optimization effective for the probabilistic tolerancing approach requires efficient algorithms to estimate the yield and its sensitivity with respect to tolerances. A major obstacle of probabilistic tolerancing for general nonlinear system is the tremendous amount of CPU time required to recompute the yield at each iteration of the optimization. To reduce the time for the yield computation, Parkinson [20] and the authors [14,15] used the notion of reliability index [9] as an approximation of the yield. However, the computation of the approximation is still intensive. In [14], a local circular search was used to find a global optimal solution. Michael and Siddall [19] decomposed the random variable space into orthogonal n-dimensional cubes, where n is the number of dimensions considered; that gives $O(2^n)$ cells to be tested. Other attempts which impose restrictions on the domain of the problem such as, linearity [2,7,11,24,25,26] and single design constraint [22], have practical limitations.

The purpose of this paper is to provide a general framework for tolerance synthesis for nonlinear systems with multiple, dependent[1] design constraints. Least-cost tolerance synthesis is mathematically formulated as a nonlinear programming (NLP) problem. Algorithms are provided and illustrated by an example. Post-optimality analysis is also considered so as to allow the possibility of modifying some yield constraints in the design process.

[1]. The design constraints are termed "dependent" if they share the same dimension variables. For instance, the two design constraints x_1-x_2-$0.1\leq 0$ and x_1+x_3-$0.05\leq 0$ are dependent because they share the same variable x_1.

2. Basic Concepts

2.1 Design Function

Design and assembly are concerning more with inter-component relations. Consider a simple assembly as shown in Figure 3.

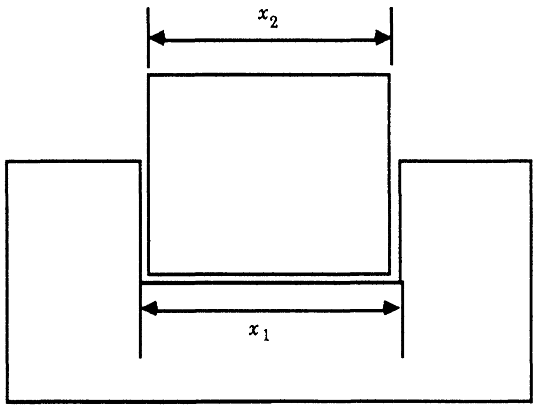


Figure 3. An Assembly

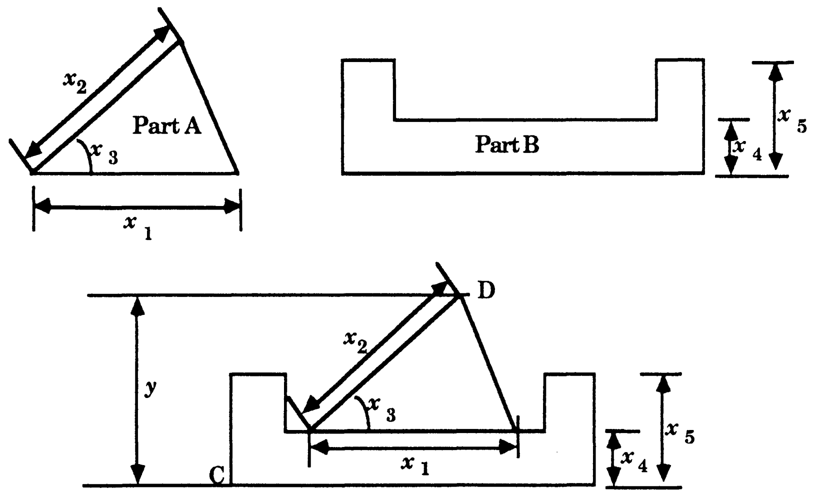
Suppose the clearance between the components has to be greater than or equal to 0.01 to achieve certain performance and assemblability. This requirement can be expressed mathematically as:

$$x_1 - x_2 \geq 0.01, \tag{1}$$

where x_1 and x_2 denote the dimensions of the hole and the shaft respectively as shown in the figure. Rewriting inequality (1) as a function $F(\boldsymbol{x},c)$ gives:

$$F(\boldsymbol{x},0.01) = - x_1 + x_2 + 0.01 \leq 0. \tag{2}$$

Functions such as (2) are called *design functions*. They describe the inter-component relations, and provide a mathematical basis for controlling functionality and interchangeability. In (2), 0.01 is referred to as a *design constant* c of the design function $F(\boldsymbol{x},c)$. A design function is not always linear. For example, if the assembly is not recti-linear, as illustrated in Figure 4, some of the design functions will take on trigonometric terms.



y : vertical distance from C (of part B) to D (of part A) is equal to $x_4 + x_2 \sin x_3$

Figure 4. A Non-linear Design Function

A linear system is one in which an assembly dimension y is defined by a linear combination of component dimensions $x_1, x_2, \ldots, x_n$:

$$y = \sum_{j=1}^{n} a_j x_j \tag{3}$$

where a_j is a signed binary integer. As variations accumulates, the tolerance analysis (or "stack-up", as it is commonly referred to) for linear systems is useful and has been studied extensively [2,7,11,24,25,26]. The key property that facilitates the analysis is that the variance of the linear sum is the sum of the variances of component dimensions under the assumption of independence, that is,

$$\sigma_y^2 = \sum_{j=1}^{n} (a_j)^2 \, \sigma_{x_j}^2. \tag{4}$$

When tolerance t_j of dimension x_j is defined as $\pm 3\sigma_j$ from the mean μ_j, the assembly tolerance t_y can be then represented by

$$t_y = k_y \sqrt{\sum_{j=1}^{n} (a_j)^2 \left(\frac{t_j}{6}\right)^2} \tag{5}$$

where k_y is a constant derived from an allowable percentage λ_y of defect in the assembly. In the case of normal distribution, $k_y = 2 * \Phi^{-1}(1 - \frac{\lambda_y}{2})$. However, the analysis procedure for linear system can not be extended to nonlinear system because of the lack of a general rule for an aggregate such as equation (4).

2.2 Yield

The probabilistic approach is performed under the assumption that the dimension vector $\boldsymbol{x}$ follows the multivariate normal distribution. (Indeed, Mansoor [18] shows that most manufacturing processes produce dimensions with normal distributions.) Let μ_j and σ_j denote the mean and the standard deviation of the normally distributed random variable x_j. The mean μ_j is typically fixed by the designer, whereas the standard deviation σ_j is chosen according to the precision of the controls exercised over the manufacturing process. This parameter σ_j is therefore a function of the tolerance t_j, and according to standard practice, σ_j is normally set to $\frac{t_j}{6}$. Clearly, if t_j is given, x_j is a well defined random variable.

Probabilistic tolerance analysis can be stated more succinctly in a mathematical formulation: Given tolerances t_j (or standard deviations σ_j), determine the probability such that the design function $F(\boldsymbol{x},c)$ is less than or equal to zero, i.e., $Pr(F(\boldsymbol{x},c) \leq 0)$. In other words, evaluate the integral,

$$\text{yield} \equiv Y(\boldsymbol{t}) = \int_{F(\boldsymbol{x},c)\leq 0} f(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{t},\boldsymbol{R})\, \mathrm{d}\boldsymbol{x} \tag{6}$$

where $F(\boldsymbol{x},c)$ is a design function, and $f(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{t},\boldsymbol{R})$ is the probability density function of multivariate normal vector $\boldsymbol{x}$ for which $\boldsymbol{\mu},\boldsymbol{t}$, and $\boldsymbol{R}$ denote the mean vector, the tolerance vector (standard deviation vector), and the correlation matrix of $\boldsymbol{x}$, respectively. (If the random variables x_j are independent, it will not be necessary to specify covariances.) The integral (6) shall be referred as the *yield*.

It helps the intuition to visualize the interaction between the tolerance and the design function. Consider an assembly of two random variables, x_1 and x_2. A design function $F(\boldsymbol{x},c)$ partitions the two-dimensional space into two regions. The *safe region* $\mathbf{R}_S$ in Figure 5(a) corresponds to the region in which $F(\boldsymbol{x},c) \leq 0$. (The complement of the safe region is called the *failure region* $\mathbf{R}_F$.) Now, the given tolerances also prescribe a region in the same space. In Figure 5(b), the upper and lower limits of a random variable x_j define a strip. Intersecting the strips for x_1 and x_2 gives the *tolerance region* $\mathbf{R}_T$. It is noted that the size of $\mathbf{R}_T$ varies with tolerances, but $\mathbf{R}_S$ (or $\mathbf{R}_F$) is independent of tolerances. Combining $\mathbf{R}_T$ with $\mathbf{R}_S$ gives the *reliable region* $\mathbf{R}_R$ as shown in Figure 5(c). For any system to perform reliably, $\mathbf{R}_R$

must be non-empty. (Such would be the case if the tolerances were not assigned properly or the design function was incorrectly specified.)

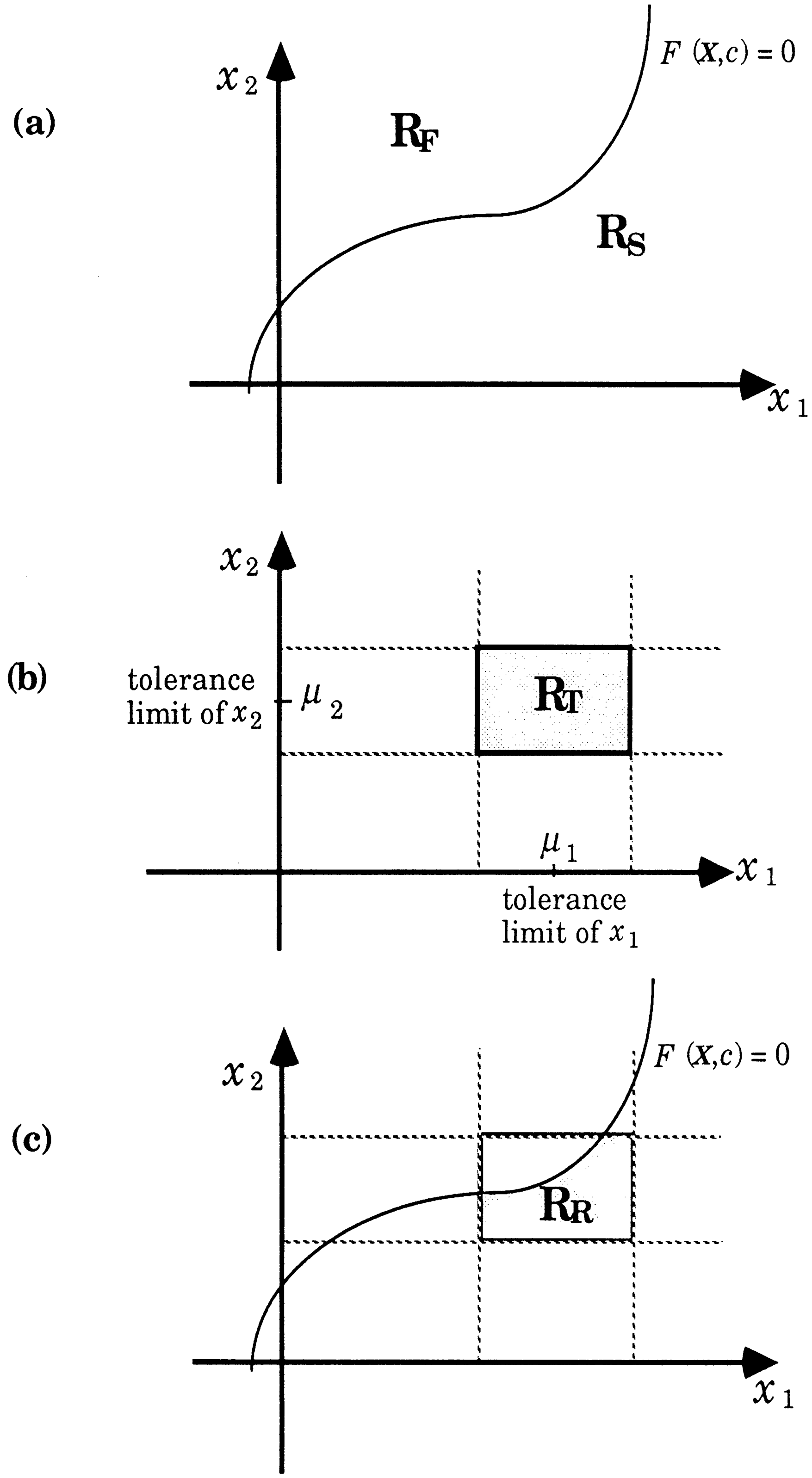


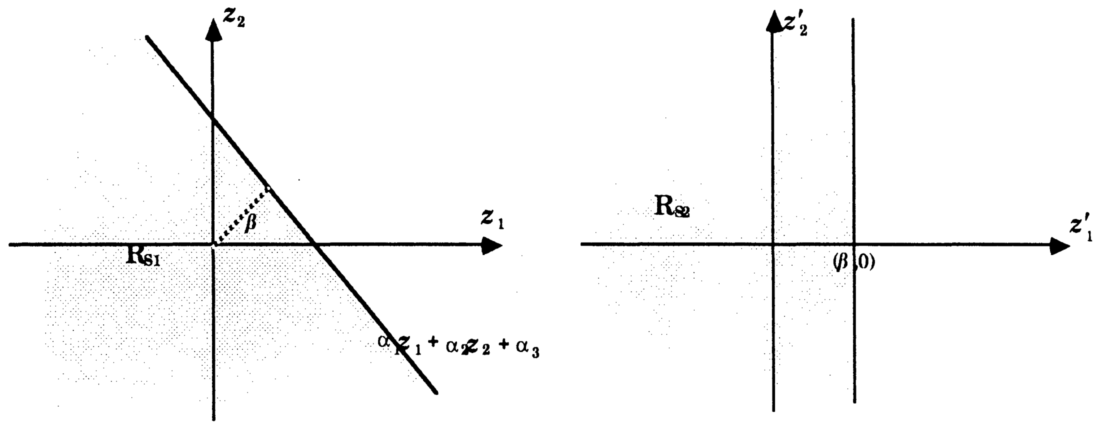
Figure 5. Safe, Tolerance and Reliable Regions

3. Tolerance Analysis

3.1 Yield of Linear Design Function

While this paper addresses systems with nonlinear design functions, discussion of the linear case facilitates subsequent development as non-linear functions will be approximated by hyperplanes. The yield of a linear design function $F(\boldsymbol{x},c)$ as computed from (6), involves multiple integration. An approximation method which can be extended to the nonlinear case[2] adopts the notion of the *reliability index*, which was introduced by Hasofer and Lind [9]. Consider Figure 6, in which there are two independent random variables z_1 and z_2 both following the standard normal distribution $N(0,1)$. The (standardized) linear design function is given by the form $\alpha_1 z_1+\alpha_2 z_2+\alpha_3$. The desired yield $Pr(\alpha_1 z_1+\alpha_2 z_2+\alpha_3 \leq 0)$ is equal to $\Phi(\beta)$, where $\beta=\frac{|\alpha_3|}{\sqrt{\alpha_1^2+\alpha_2^2}}$ is the minimum distance from the design function to the origin. The distance β is referred to as the *reliability index* of the design function. Because of the rotational symmetry of the standard normal coordinate, the yield can be obtained by looking up the value of $\Phi(\beta)$ in the univariate standard normal distribution table. The above technique is next generalized for computing the yield of a design function with any number of random variables.

2. The observation that a linear combination of multivariate normal random variables follow a univariate normal distribution [6, p.56] does not extend readily to the case of nonlinear design functions.



$$Pr(\mathbf{R}_{S1}) = Pr(\alpha_1 z_1 + \alpha_2 z_2 + \alpha_3 \le 0) = \Phi(\beta) = Pr(\mathbf{R}_{S2})$$

Figure 6. Rotational Symmetry of Standard Coordinate

In general, tolerance analysis for linear design functions $F(\boldsymbol{x},c)=\boldsymbol{a}^T\boldsymbol{x}+c$ requires the following steps:

(i) **Standardization** : Transform the dependent normal variables to the independent standard normal variables by using

$$\boldsymbol{z} = (\boldsymbol{PD})^{-1}(\boldsymbol{x} - \mu) \tag{7}$$

where $\boldsymbol{z}$ is the transformed standard normal vector and $\boldsymbol{P}$ is the orthogonal matrix for diagonalizing a given covariance matrix $\boldsymbol{V}$ such that $\boldsymbol{P}^T\boldsymbol{V}\boldsymbol{P}=\boldsymbol{D}\boldsymbol{D}^T$. (The detailed transformation procedure is given in Appendix A.) The transformed $\boldsymbol{z}$ space is referred to as the *standard coordinates*.

(ii) **Reliability Index Computation** : Compute the minimum distance β from the origin to the transformed design function in the standard coordinates. First, represent the design function in terms of $\boldsymbol{z}$:

$$\boldsymbol{a}^T\boldsymbol{x} + c = \boldsymbol{a}^T\boldsymbol{PDz} + \boldsymbol{a}\mu + c. \tag{8}$$

The distance from the origin to the right hand side of equation (8) corresponds to β which can be then expressed as

$$\beta = \frac{-\boldsymbol{a}^T\mu - c}{\sqrt{(\boldsymbol{a}^T\boldsymbol{PD})(\boldsymbol{a}^T\boldsymbol{PD})^T}} = \frac{-\boldsymbol{a}^T\mu - c}{\sqrt{\boldsymbol{a}^T\boldsymbol{V}\boldsymbol{a}}} \tag{9}$$

since $\boldsymbol{PD}(\boldsymbol{PD})^T=\boldsymbol{PDD}^T\boldsymbol{P}^T=\boldsymbol{P}(\boldsymbol{P}^T\boldsymbol{VP})\boldsymbol{P}^T=\boldsymbol{V}$. Note that the numerator of equation (9), $-\boldsymbol{a}^T\mu - c$, is always positive since the given nominal dimensions, μ, are assumed to satisfy the design condition.

(iii) **Table Look-Up** : Look up the univariate standard normal distribution table for $\Phi(\beta)$.

3.2 Yield of Nonlinear Design Functions

To solve the integral (6) under nonlinear function $F(\boldsymbol{x},c)$ and multivariate normal PDF, two techniques are considered: (i) Monte-Carlo simulation [5,8] and (ii) approximation through the linearization of $F(\boldsymbol{x},c)$ [14,15,20].

Monte-Carlo simulation starts with generating N sets of random samples $(x_{11},x_{21},\ldots,x_{n1}),\ldots,(x_{1N},x_{2N},\ldots,x_{nN})$ from the given multivariate normal PDF, where x_{jl} denotes the l-th sample (l=1,...,N) of the j-th dimension (j=1,...,n). Then, each set is substituted into the design function and the sign of the functional value is checked. Suppose T sets out of N sets have a nonpositive sign. Then, the estimate of the yield is $\frac{T}{N}$. While Monte-Carlo simulation can be applied to the linear or nonlinear $F(\boldsymbol{x},c)$, it is time intensive because a large number of random samples needs to be taken to have an accurate result. This intensive consumption of time is compounded in *tolerance synthesis*, which requires iterative tolerance analysis to estimate the yield and the gradient.

In order to reduce the computational time, approximation of yield by linearization of nonlinear function is used. An expansion point of a design function $F(\boldsymbol{x},c)$ is selected, then linearization is done through Taylor series expansion[3]. Note that the probability density in the standard coordinate decreases exponentially as the distance from the origin increases. This suggests that for an expansion point to be well-selected the (probabilistically) densest area should be preserved after linearization.

The "dense" area is estimated by the distance β. The "densest" area is estimated by minimizing β. Each design function is standardized by transformation (7), and the point on each standardized design function which has the minimum distance from the origin is selected as the expansion point. Consider two points p_1 and p_2 on $G(\boldsymbol{z},\gamma)$, the standardized $F(\boldsymbol{x},c)$, as candidates for expansion points. In figure 7, lines L_1 and L_2 correspond to linearization of $G(\boldsymbol{z},\gamma)$ through p_1 and p_2. Since $d_1 < d_2$, p_1 is closer to the origin than p_2.

[3]. The first order Taylor series expansion gives the tangent hyperplane passing through the expansion point.

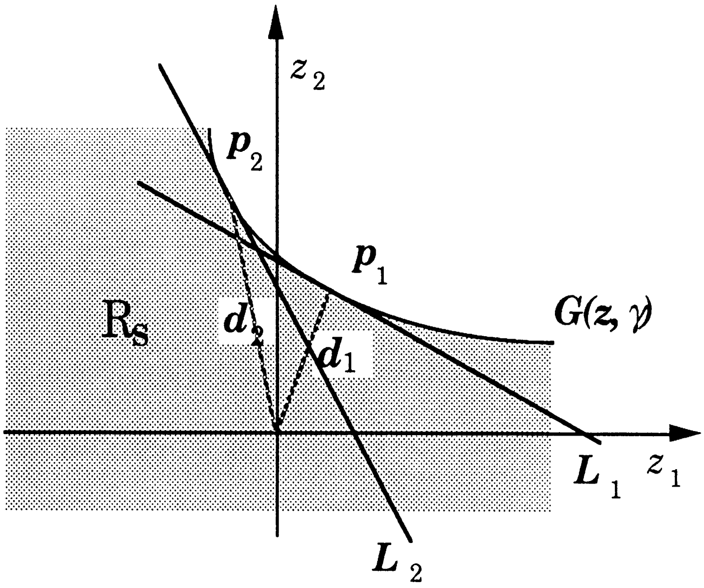


Figure 7. Expansion Point for Linearization

The expansion point having the minimum distance from the origin is obtained by solving the following single constraint NLP:

$$\text{Min } \beta = \sqrt{\boldsymbol{z}^T\boldsymbol{z}}\ , \quad \text{subject to } G(\boldsymbol{z},\gamma) = 0. \tag{10}$$

Based on equation (7), formulation (10) can be rewritten with the original variables:

$$\text{Min } \beta^2 = (\boldsymbol{x}-\mu)^T\boldsymbol{V}^{-1}(\boldsymbol{x}-\mu)\ , \quad \text{subject to } F(\boldsymbol{x},c) = 0. \tag{11}$$

Here, the objective function is expressed as β^2 instead of β since the positive definiteness of the covariance matrix $\boldsymbol{V}$ always guarantees the same solution. As a solution scheme for (11), the iterative method based on Newton-Raphson method [17] is used:

$$\boldsymbol{x}^{(k+1)} = \mu + \boldsymbol{V}\ \nabla F(\boldsymbol{x}^{(k)},c)\ \frac{(\boldsymbol{x}^{(k)} - \mu)^T\ \nabla F(\boldsymbol{x}^{(k)},c) - F(\boldsymbol{x}^{(k)},c)}{\nabla F(\boldsymbol{x}^{(k)},c)^T\ \boldsymbol{V}\ \nabla F(\boldsymbol{x}^{(k)},c)} \tag{12}$$

where $\boldsymbol{x}^{(k)}$ denotes the solution after the k-th iteration and $\nabla F(\boldsymbol{x},c)$ is the n×1 gradient vector of $F(\boldsymbol{x},c)$ at $\boldsymbol{x}$. As an initial solution for (12), the mean nominal vector μ is used.

Tolerance analysis for nonlinear design function therefore takes the following steps:

(i) **Expansion Point Finding** : Find the expansion point $\boldsymbol{x}^*$ by iteratively using equation (12).

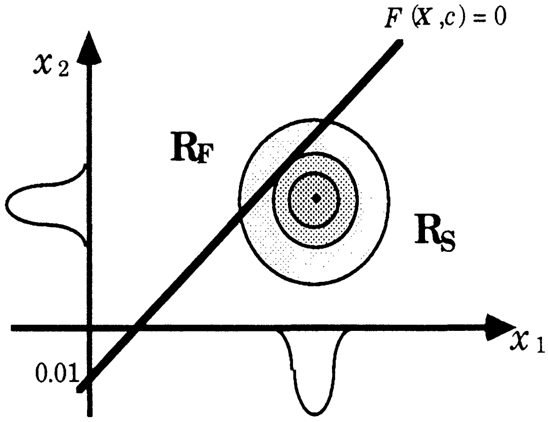
(ii) **Reliability Index Computation** : The reliability index β is computed by $\sqrt{(\boldsymbol{x}^*-\mu)^T\boldsymbol{V}^{-1}(\boldsymbol{x}^*-\mu)}$

(iii) **Table Look-Up** : The yield $Y(\boldsymbol{t})$ (i.e.,$Pr(F(\boldsymbol{x},c) \leq 0)$) is approximated by $\Phi(\beta)$, which can be looked up in the univariate standard normal distribution table.

4. Least Cost Tolerance Synthesis

The problem in the context of tolerance analysis is: given design functions and tolerances, compute the yield. Suppose, instead, the yield is given, how are the tolerances to be assigned to each dimension? Figure 8 illustrates the impact of two different sets of tolerances on the yield. In the figures, the concentric circles represent the contour of equal probability density. It is noted that the tighter set of tolerances (as in Figure 8(a)) results in a higher yield as compared to the yield with the looser set of tolerances (as in Figure 8(b)).

(a)



(b)

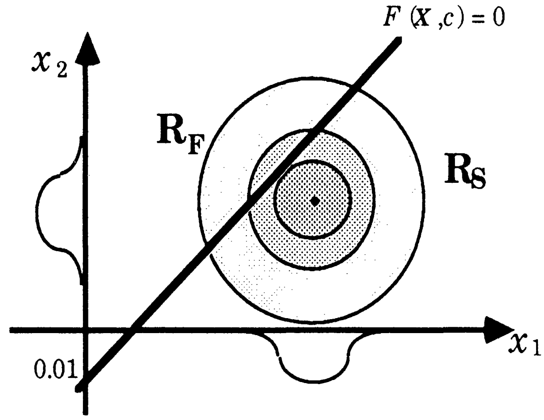


Figure 8. Impact of Different Tolerances on Yield

A higher yield, while desirable, is achieved at the expense of other considerations such as cost and manufacturability. To resolve this conflict, heuristic rules have been proposed [2]: (i) equal tolerances, (ii) tolerances proportional to dimensions, and (iii) tolerances proportional to process deviations. While heuristics are practical, it would be intellectually satisfying to see if assignments of tolerances can be optimized.

4.1 Tolerance-Cost Model

As indicated in the introduction, it is generally accepted that there is an inverse relationship between tolerance and manufacturing cost. A number of cost models have been employed to fit manufacturing tolerance-cost sampled data [10,19,22,23,25], as shown in Table 1.

Model Name	**References**	**Cost Function** (*)
Sutherland-Roth Model	[23]	$C(t) = a\,t^{-b} + f$
Reciprocal Squared Model	[10, 22]	$C(t) = \frac{a}{t^2} + f$
Exponential Model	[25]	$C(t) = a\ \exp\{-\frac{t}{b}\} + f$
Michael-Siddall Model	[19]	$C(t) = a\,t^{-b}\ \exp\{-e\,t\} + f$

(*) a , b , e constants for variable manufacturing cost
f : constant for fixed manufacturing cost

Table 1. Tolerance-Cost Models

With such tolerance-cost models for the toleranced dimensions, the total manufacturing cost can be obtained by summing the individual manufacturing cost:

$$C(\boldsymbol{t}) = \sum_{j=1}^{n} C(t_j). \tag{13}$$

Note that model (13) is based on the "throw-aways" strategy, that is, the repair cost for the defect is not considered in the model. However, if the reworking cost is considered, the model becomes:

$$C(\boldsymbol{t}) = \sum_{j=1}^{n} C(t_j) + C_r(p_1, \dots, p_n) \{1 - Y(\boldsymbol{t})\} \tag{14}$$

where $C_r(\bullet)$ is the cost function due to reworking and p_j is the probability of reworking the j-th dimension in case that the design function is not satisfied. The difficulty of employing model (14) is in procuring the empirical values for p_j. In this paper, the throw-away cost model (13) is adopted.

4.2 Mathematical Formulation

Least-cost tolerance allocation is a procedure for determining an optimal set of tolerances which minimizes the manufacturing cost and satisfies the performance requirement; the decision variables are the tolerances. Therefore, we can formulate the problem with minimizing the manufacturing cost as the objective function and satisfying the performance requirements as the constraints:

$$\text{Min} \quad C(\boldsymbol{t}), \quad \text{subject to } Y(\boldsymbol{t}) \geq 1 - \lambda. \tag{15}$$

where $C(\boldsymbol{t})$ is the manufacturing cost function in terms of tolerance $\boldsymbol{t}$, and (1-λ) is the minimal satisfactory yield, i.e., the yield given by tolerances $\boldsymbol{t}$ should be greater than or equal to the given level 1-λ. Formulation (15) implies that tolerance synthesis includes tolerance analysis.

With cost model (13) and multiple design functions, formulation (15) is rewritten as

$$\text{Min} \quad \sum_{j=1}^{n} C(t_j) \tag{16}$$

subject to

$$Y_i(\boldsymbol{t}) \geq 1 - \lambda_i \qquad \text{for } i = 1, 2, ..., m.$$

This probabilistic optimization problem is simplified into a deterministic optimization problem through approximation of the yield by the reliability index.

This conversion into deterministic optimization differs from chance constrained programming due to Charnes and Cooper [3] in that the yield is approximated not at the origin of the standard coordinates but at the point of minimum distance from the origin. Thus, $Y_i(\boldsymbol{t})$ is approximated by $\Phi(\beta_i)$, as suggested in Section 3.2. The constraints of (16) are rewritten as $\Phi(\beta_i) \geq 1 - \lambda_i$. Furthermore, by the monotonic property of the function $\Phi(\cdot)$ the constraint $\Phi(\beta_i) \geq 1\text{-}\lambda_i$ can be inverted to the constraint $\beta_i \geq \Phi^{-1}(1\text{-}\lambda_i)$, and the formulation becomes

$$\text{Min} \quad \sum_{j=1}^{n} C(t_j) \tag{17}$$

subject to

$$\beta_i \geq \Phi^{-1}(1 - \lambda_i) \qquad \text{for } i = 1, 2, ..., m.$$

Note that β_i is a function (with respect to σ_j); and $\Phi^{-1}(1-\lambda_i)$ is equal to a value q_i which can be obtained from the standard normal distribution table.

Recall that an expansion point x_j^* is obtained from solving formulation (11) with a fixed σ_j. However, the "optimal" σ_j obtained from solving formulation (17) is in general different from the initial σ_j. Therefore, in the solution process, as an x_j^* is changed, the constraints of (17) are changed as well. Consequently, σ_j may no longer be optimal or even feasible. In order to reflect the changes in x_j^*, formulation (11) (for all design functions) is added to formulation (17) as constraints, hence, guaranteeing the satisfaction of locally optimal conditions.

The local optima of (11) are the solutions that satisfy the Kuhn-Tucker necessary condition[4] [27]. Since (11) is a minimization problem, its constraints, $F_i(x,c)=0$, can be relaxed to $F_i(x,c)\geq 0$. With the Kuhn-Tucker necessary conditions, formulation (17) becomes:

$$\text{Min} \quad \sum_{j=1}^{n} C(t_j) \tag{18}$$

subject to

$$\beta_i \geq \Phi^{-1}(1 - \lambda_i)$$

$$\frac{\partial \beta_i}{\partial x} - u_i \frac{\partial F_i(x^*,c)}{\partial x} = 0 \tag{18.1}$$

$$u_i F_i(x^*,c) = 0 \tag{18.2}$$

$$u_i \geq 0 \tag{18.3}$$

$$\text{for } i = 1, 2, ..., m .$$

[4] β_i and $F(x,c)$ being differentiable at x^* are assumed.

4.3 Global Optimality Analysis

In Nonlinear Programming, no existing algorithm guarantees a globally optimal solution unless the objective function and the constraints are of certain forms. The existence check for the special forms is performed based on the following theorem [27, pp.43-44]: that the objective function is convex and the constraint functions are quasi-concave corresponds to a sufficient condition for global optimality. This ensures that the locally optimal solution implied by the Kuhn-Tucker necessary conditions is also a globally optimal solution.

Checking formulation (18) for the satisfaction of the Kuhn-Tucker sufficient condition proceeds as follows. The objective function is the sum of the individual tolerance-cost function (as in (13)). Cost models such as the ones in Table 1 are convex because the derivative of $C(t_j)$ are monotonically nondecreasing with respect to tolerances. Since the sum of convex functions is also convex, the tolerance-cost function of (13) is always convex.

Similarly, the derivatives of the reliability index function are checked for quasi-concavity. $\frac{\partial \beta}{\partial \sigma_j}$ is obtained by following the chain rule:

$$\frac{\partial \beta}{\partial \sigma_j} = \frac{\partial \beta}{\partial \beta^2} * \frac{\partial \beta^2}{\partial \sigma_j} = \frac{1}{2\beta} * \frac{\partial \beta^2}{\partial \sigma_j} \ . \tag{19}$$

For $\frac{\partial \beta^2}{\partial \sigma_j}$, rewrite β^2 as:

$$\beta^2 = (\boldsymbol{x}^*\text{-}\mu)^T \boldsymbol{V}^{-1}(\boldsymbol{x}^*\text{-}\mu) = (\boldsymbol{x}^*\text{-}\mu)^T(\boldsymbol{DRD})^{-1}(\boldsymbol{x}^*\text{-}\mu) = \boldsymbol{g}^T\boldsymbol{R}^{-1}\boldsymbol{g}$$

where $\boldsymbol{D}$ is a diagonal matrix whose the k-th diagonal element is $\frac{1}{\sigma_k}$, $\boldsymbol{R}$ is a nxn correlation matrix, and $\boldsymbol{g}=(\frac{x_1^*-\mu_1}{\sigma_1},\ldots,\frac{x_n^*-\mu_n}{\sigma_n})^T$. Then,

$$\frac{\partial \beta^2}{\partial \sigma_j} = \frac{\partial}{\partial \sigma_j} \boldsymbol{g}^T \boldsymbol{R}^{-1} \boldsymbol{g} = \frac{\partial \boldsymbol{g}^T}{\partial \sigma_j} \boldsymbol{R}^{-1} \boldsymbol{g} + \boldsymbol{g}^T \boldsymbol{R}^{-1} \frac{\partial \boldsymbol{g}}{\partial \sigma_j}$$

$$= 2 \frac{\partial \boldsymbol{g}^T}{\partial \sigma_j} \boldsymbol{R}^{-1} \boldsymbol{g} = -2\,(0,\ldots,0,\frac{x_j^*-\mu_j}{\sigma_j^2},0,\ldots,0)\, \boldsymbol{R}^{-1} \boldsymbol{g}$$

$$= -\,2 \frac{x_j^* - \mu_j}{\sigma_j^2} \left(\sum_{i=1}^{n} \hat{\rho}_{ji} \frac{x_i^* - \mu_i}{\sigma_i} \right). \tag{20}$$

where $\hat{\rho}_{ji}$ is the (i,j)-th element of matrix $\boldsymbol{R}^{-1}$. Substituting (20) into (19) results in

$$\frac{\partial \beta}{\partial \sigma_j} = -\frac{x_j^* - \mu_j}{\beta\, \sigma_j^2} \left(\sum_{i=1}^{n} \hat{\rho}_{ji} \frac{x_i^* - \mu_i}{\sigma_i} \right). \tag{21}$$

The quasi-concavity of the reliability index function with respect to standard deviation is checked under the assumption of independence among dimension variables, i.e., $\hat{\rho}_{ji}=1$ if $j=i$, and $\hat{\rho}_{ji}=0$ otherwise. Therefore, equation (21) can be rewritten as

$$\frac{\partial \beta}{\partial \sigma_j} = -\frac{(x_j^* - \mu_j)^2}{\beta\, \sigma_j^3}. \tag{22}$$

The quasi-concavity of the reliability index functions is thus shown by the nonincreasing monotonic relationship between the reliability index and the tolerances.

Since there is no restriction on the type of the design functions that can be used, equations (18.1) and (18.2) are not necessarily quasi-concave.

As a result, with arbitrary design functions, the optimal solution of formulation (18) is not guaranteed.

4.4 Algorithmic Analysis

The feasible direction method is commonly used to solve the NLP problems. The constraints of formulation (18) comprise a feasible region. A point in the feasible region corresponds to a feasible assignment of tolerance. The total cost with the assigned tolerances is obtained by evaluating the objective function at that point.

Suppose $\sigma^{(k)}$ is a point in the feasible region. A direction $d^{(k)}$ is identified such that, for a sufficiently small $\lambda>0$, the following two properties are true: i) $\sigma^{(k+1)} = \sigma^{(k)}+\lambda d^{(k)}$ is feasible, and ii) the objective value at $\sigma^{(k+1)}$ is better than the objective value at $\sigma^{(k)}$. In each iteration of the feasible direction method, having determined a feasible direction, a one-dimensional optimization problem is solved to maximize the improvement of the objective value.

If formulation (11) satisfies the Kuhn-Tucker sufficient condition and equations (18.1) and (18.2) are quasi-concave, the optimal solution of (18) is guaranteed by applying the feasible direction method. That is, if the feasible direction algorithm reaches a point that satisfies the Kuhn-Tucker necessary condition, then the corresponding tolerance assignment requires the least cost. The modified version of Zoutendijk's Method due to Topkis and Veinott [16,27] can be applied and is guaranteed to converge.

If either the formulation (11) does not satisfy the Kuhn-Tucker sufficient condition or equations (18.1) and (18.2) are not quasi-concave (or both), the sensitivity of yield is analyzed. The sensitivity of yield provides information about which tolerances are critical and helps determine the search direction in the optimization process for tolerance assignment. With respect to a tolerance t_j, it is defined as

$$s(\beta,t_j) = \frac{\partial Y(\boldsymbol{t})}{\partial t_j} . \tag{23}$$

Now that $Y(\boldsymbol{t})$ is approximated by $\Phi(\beta)$ (by step (iii) in tolerance analysis for nonlinear design functions), (23) can be approximated by $\frac{\partial \Phi(\beta)}{\partial t_j}$. (For a linear design function, $\frac{\partial \Phi(\beta)}{\partial t_j}$ is the exact yield sensitivity.)

Expanding $s(\beta,t_j)$ by the chain rule reveals the search direction:

$$s(\beta,t_j) = \frac{\partial \Phi(\beta)}{\partial \beta} \frac{\partial \beta}{\partial \sigma_j} \frac{\partial \sigma_j}{\partial t_j} = \phi(\beta) * \{- \frac{x_j^* - \mu_j}{\beta\, \sigma_j^2} \left(\sum_{i=1}^{n} \hat{\rho}_{ji} \frac{x_i^* - \mu_i}{\sigma_i} \right)\} * \frac{1}{6} \tag{24}$$

where $\hat{\rho}_{ji}$ is the (i,j) element of the inverse matrix of the correlation matrix,

x_j^* is the j-th coordinate of the expansion point, and

$\phi(\cdot)$ is the PDF of the standard normal distribution, i.e.,

$$\phi(\beta) = \frac{1}{\sqrt{2\pi}} \exp\{-\frac{\beta^2}{2}\}.$$

Substituting (22) into (24) and by assumption of independence of dimensions, the sensitivity of yield is rewritten as

$$s(\beta,t_j) = \phi(\beta) \ * \left\{- \frac{(x_j^* - \mu_j)^2}{\beta \ \sigma_j^3}\right\} * \frac{1}{6}. \tag{25}$$

Equation (25) shows that the $s(\beta,t_j)$ are always nonpositive. This means that there is a nonincreasing *monotonicity* between yield and tolerance. This monotonic property demonstrates the trade-off between tolerance and performance: the performance, which is implied by yield, increases as the tolerances are tightened.

4.5. Lagrangean Multiplier

The impact of modifying yields constraints on the manufacturing cost, i.e., the objective value, is considered in this section. This is performed as a post-optimality analysis.

The partial derivative $\frac{\partial C(\sigma_1,\ldots,\sigma_n)}{\partial \lambda_i}$ at the optimal solution $(\sigma_1^*,\ldots,\sigma_n^*)^T$ is decomposed into two parts using the chain rule:

$$\frac{\partial C(\boldsymbol{t})}{\partial \lambda_i} = \frac{\partial C(\boldsymbol{t})}{\partial q_i} \frac{\partial q_i}{\partial \lambda_i} \tag{26}$$

The second part of this decomposition, $\frac{\partial q_i}{\partial \lambda_i}$, turns out to be:

$$\frac{\partial q_i}{\partial \lambda_i} = - \frac{\sqrt{2\pi}}{\exp\left\{- \frac{q_i^2}{2}\right\}}$$

since $\lambda_i = 1-\Phi(q_i)$. The first part of (26), $\frac{\partial C(\boldsymbol{t})}{\partial q_i}$, called the "Lagrangean multiplier" in NLP has the following characteristic: if the constraint is inactive at the optimal solution, then the corresponding Lagrangean multiplier should be zero. This means that for an inactive constraint at optimum the corresponding q_i can be modified slightly without incurring additional manufacturing cost.

In case that a constraint is active at optimum, the corresponding Lagrangean multiplier should be greater than zero. It is computed as:

$$\frac{\partial C(\boldsymbol{t})}{\partial q_i} = \sum_{j=1}^{n} \frac{\partial C(\boldsymbol{t})}{\partial \sigma_j} \frac{\partial \sigma_j}{\partial q_i} = \sum_{j=1}^{n} \frac{\partial C(\boldsymbol{t})}{\partial \sigma_j} \frac{\partial \sigma_j}{\partial \beta_i} \tag{27}$$

since the i-th constraint is assumed to be active, i.e. $q_i = \beta_i$. From equation (22), $\frac{\partial \beta_i}{\partial \sigma_j} = - \frac{(x_{ij}^{*} - \mu_j)^2}{\beta_i \sigma_j^3}$, where x_{ij}^{*} is the j-th expansion point for β_i, (27) becomes

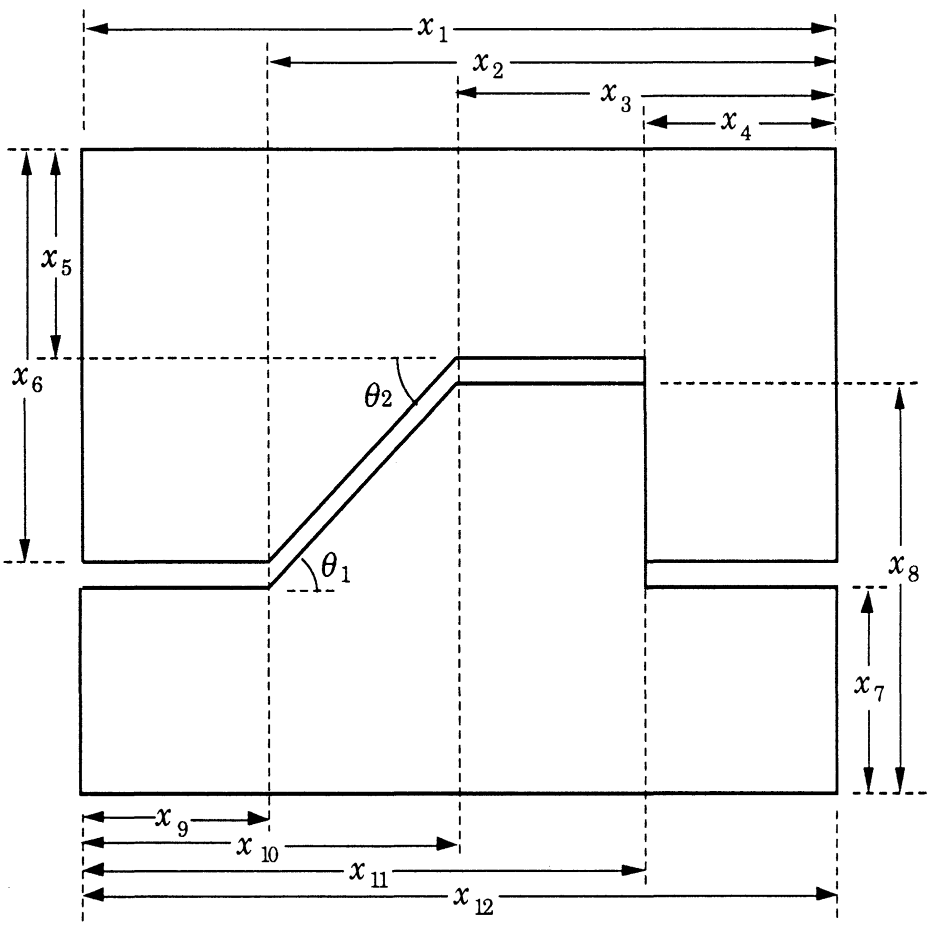
$$\frac{\partial C(\boldsymbol{t})}{\partial q_i} = \sum_{j=1}^{n} \frac{\partial C(\boldsymbol{t})}{\partial \sigma_j} \left\{- \frac{\beta_i \sigma_j^3}{(x_{ij}^{*} - \mu_j)^2}\right\}. \tag{28}$$

Based on the inverse-squared model, i.e., $C(\boldsymbol{t}) = \sum_{k=1}^{n} \left\{\frac{a_k}{(6\sigma_k)^2} + f_k\right\}$, (28) becomes

$$\frac{\partial C(\boldsymbol{t})}{\partial q_i} = \sum_{j=1}^{n} \frac{-a_j}{18\sigma_j^3} \left\{- \frac{\beta_i \sigma_j^3}{(x_{ij}^{*} - \mu_j)^2}\right\} = \sum_{j=1}^{n} \frac{a_j \beta_i}{18(x_{ij}^{*} - \mu_j)^2}.$$

5. Implementation

An assembly of two parts with twelve dimensions is shown in Figure 9(a). Six design functions, linear and non-linear, are given in Figure 9(b). Design functions $F_1(X)$ and $F_2(X)$ represent the vertical and the horizontal clearance conditions of the two parts. Design functions $F_3(X)$ and $F_4(X)$ post the restriction on the difference between angles θ_1 and θ_2 to ensure feasibility of assembly. Design functions $F_5(X)$ and $F_6(X)$ give the requirements for the size difference of those two parts.



(a)

$$F_1(X) = (x_6 - x_5) - (x_8 - x_7)$$

$$F_2(X) = (x_3 - x_4) - (x_{11} - x_{10})$$

$$F_3(X) = (x_8 - x_7)(x_2 - x_3) - (x_6 - x_5)(x_{10} - x_9) + \tan(\pi/180) * \{(x_{10} - x_9)(x_2 - x_3) + (x_8 - x_7)(x_6 - x_5)\}$$

$$F_4(X) = (x_6 - x_5)(x_{10} - x_9) - (x_8 - x_7)(x_2 - x_3) + \tan(\pi/180) * \{(x_{10} - x_9)(x_2 - x_3) + (x_8 - x_7)(x_6 - x_5)\}$$

$$F_5(X) = -x_1 + x_{12} + 0.01$$

$$F_6(X) = x_1 - x_{12} + 0.01$$

(b)

Figure 9. Examples of Linear and Nonlinear Design Functions.

The nominal dimensions are given as X^T = (50.0, 40.00125, 20.05, 9.9985, 9.9985, 30.0, 10.0, 30.0, 10.05, 30.0, 40.0, 50.0). The cost function with respect to the tolerance of each dimension is

$$C(\sigma_i) = \frac{a_i \times 10^{-3}}{(6\sigma_i)^{b_i}}.$$

The coefficients of the respective cost functions are: $a_1 = 0.2$, $a_2 = 1.0$, $a_3 = a_4 = 0.015$, $a_5 = 0.008$, $a_6 = 0.009$, $a_7 = 0.008$, $a_8 = 0.006$, $a_9 = 1.0$, $a_{10} = 0.01$, $a_{11} = 0.015$, and $a_{12} = 0.2$; and $b_1 = \ldots = b_{12} = 2.0$.

The algorithm given in this paper is implemented in PASCAL and runs on the IBM PC. For the algorithms to be practical in an interactive design environment, attention is given to speed - in particular, the computation for the Jacobian matrix. The initial tolerances are assigned in accordance to ANSI-Y14.5M for loose fit.

It took 5.25 CPU seconds to obtain the optimal solution, and the resulted tolerances are shown in Table 2; each stack-up condition is satisfied with 95% confidence if the dimensions x_1 to x_{12} are manufactured within the tolerances obtained. To cross check the algorithm, dimensions x_2 and x_9 are intentionally assigned with higher manufacturing costs, i.e., they are harder to be manufactured then the other dimensions. The tolerances computed are consistent with the manufacturing costs as much looser tolerances are assigned to dimensions x_2 and x_9.

Dimensions	Cost function coefficients		Tolerances
	a_i	b_i	
x_1	0.2	2.0	0.0093
x_2	1.0	2.0	0.6427
x_3	0.015	2.0	0.0337
x_4	0.015	2.0	0.0337
x_5	0.008	2.0	0.0010
x_6	0.009	2.0	0.0011
x_7	0.008	2.0	0.0010
x_8	0.006	2.0	0.0008
x_9	1.0	2.0	0.6433
x_{10}	0.01	2.0	0.0337
x_{11}	0.015	2.0	0.0337
x_{12}	0.2	2.0	0.0093

Table 2. Result of the example.

6. Concluding Remarks

A general framework for tolerance synthesis based on least manufacturing cost has been presented. The algorithm for probabilistic tolerancing has been developed based on the notion of feasible directions. An analytic result of yield sensitivity is used to speed up the computation of the Jacobian matrix inside the optimization loop. Compared to a previously established landmark [14], this new algorithm produces a solution with more than 10 times reduction in CPU time. Also, the post-optimality analysis of the algorithm enables a designer to verify design intention with ease.

References

[1] *Dimensioning and Tolerancing*, ANSI Standard Y14.5M-1982, American Society of Mechanical Engineering, New York, 1982.

[2] Bjorke, O., *Computer Aided Tolerancing*, Tapir Press, Norway, 1978.

[3] Charnes, A. and Cooper, W.W., "Deterministic Equivalents for Optimizing and Satisficing under Chance Constraints," *Ops. Res.*, Vol. 11, 1963, pp. 18-39.

[4] Dong, Z. and Soom, A., "Automatic Tolerance Analysis from a CAD Database," ASME Paper No. 86-DET-36, 1986.

[5] Evans, D.H., "Statistical Tolerancing: The State of the Art, Part II, Methods for Estimating Methods," *Journal of Quality Technology*, Vol. 7, No. 1, January 1975, pp. 1-12.

[6] Graybill, F.A., *An Introduction to Linear Statistical Model - Volume 1*, McGraw-Hill Book Co., Inc., New York, 1961.

[7] Greenwood, W.H. and Chase, K.H., "A New Tolerance Analysis Method for Designers and Manufacturers," *Trans. of ASME, Journal of Engineering for Industry*, Vol. 109, May 1987, pp.112-116.

[8] Grossman, D.D., "Monte-Carlo Simulation of Tolerancing in Discrete Parts Manufacturing and Assembly," Report No. STAN-CS-76-555, Computer Science Department, Stanford Univ., May 1976

[9] Hasofer, A.M. and Lind, N.C., "Exact and Invariant Second-Moment Code Format," *Journal of the Engineering Mechanics Division*, ASCE, Vol. 100, 1974, pp. 111-121.

[10] Hiller, M.J., "A Systematic Approach to the Cost Optimization of Tolerances in Complex Assemblies," *Bulletin of Mechanical Engineering Education*, Vol. 5, 1966, pp. 157-161.

[11] Huang, J. and Ostwald, P.F., "A Method for Optimal Tolerance Selection," ASME Paper No. 76-WA/DE-23.

[12] Latta, L.W., "The Assignment of Least Cost Tolerances," *Product Engineering*, Vol. 34, 1963, pp. 111.

[13] Lee, W.-J., *Tolerancing - Computation on Geometric Uncertainties*, Ph.D. Dissertation, Dept. of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan, 1988.

[14] Lee, W.-J. and Woo, T.C., "Tolerances: Their Analysis and Synthesis," Tech. Rep. No. 86-30, Dept. of Industrial and Operations Eng., Univ. of

Michigan, Ann Arbor, Michigan, 1986; to appear in *Trans. of ASME, Journal of Engineering for Industry.*

[15] Lee, W.-J. and Woo, T.C., "Optimum Selection of Discrete Tolerances," *Trans. of ASME, Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 111, No. 2, June 1989, pp.243-251.

[16] Luenberger, D.G., *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, 1965.

[17] Madsen, H.O., Krenk, S., and Lind, N.C., *Methods of Structural Safety*, Prentice-Hall, Inc., Englewood Cliffs, NJ 07632, 1986.

[18] Mansoor, E.M., "The Application of Probability to Tolerances Used in Engineering Designs," *Proc.Inst.Mech.Eng.*, Vol. 178, No. 1, 1963, pp. 29-51.

[19] Michael, W. and Siddall, J.N., "The Optimal Tolerance Assignment with Less Than Full Acceptance," *Trans. of ASME, Journal of Mechanical Design*, Vol. 104, Oct. 1982, pp. 855-860.

[20] Parkinson, D.B., "The Application of Reliability Methods to Tolerancing" *Trans. of ASME, Journal of Mechanical Design*, Vol. 104, July 1982, pp. 612-618.

[21] Peat, A.P., *Cost Reduction Charts for Designers and Production Engineers*, The Machining Publishing Co. Ltd., Brighton, 1968.

[22] Spotts, M.F., "Allocation of Tolerances to Minimize Cost of Assembly," *Trans. of ASME, Journal of Engineering for Industry*, Aug. 1973, pp. 762-764.

[23] Sutherland, G.H. and Roth, B., "Mechanism Design: Accounting for Manufacturing Tolerances and Costs in Function Generating Problems," *Trans. of ASME, Journal of Engineering for Industry*, 1975, pp. 283-286.

[24] Turner, J.U. and Wozny, M.J., "Tolerances Analysis in a Solid Modeling Environment," *Computers in Engineering*, Vol. 2, ASME, 1987, pp.169-175.

[25] Wilde, D. and Prentice, E., "Minimum Exponential Cost Allocation of Sure-Fit Tolerances," ASME Paper No. 75-DET-93.

[26] Wu, Z., Elmaraghy, W.H., and Elmaraghy, H.A., "Evaluation of Cost-Tolerance Algorithms for Design Tolerance Analysis and Synthesis," *Manufacturing Review*, Vol. 1, No. 3, October 1988, pp. 168-179.

[27] Zangwill, W.I., *Nonlinear Programming - A Unified Approach*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1969.

Appendix : Transformation to the Standard Coordinates

The transformation into the standard coordinates is accomplished by taking the following four steps [6]:

Step 1: Translate into the origin by $\boldsymbol{x}^{0} = \boldsymbol{x} - \mu$;

Step 2: Diagonalize the given covariance matrix $\boldsymbol{V}$ through the operation $\boldsymbol{P}^{T}\boldsymbol{V}\boldsymbol{P} = \boldsymbol{V}_{\boldsymbol{z}}$, where $\boldsymbol{P}$ is the orthogonal matrix;

Step 3: Orthogonally transform by $\boldsymbol{z}^{0} = \boldsymbol{P}^{T}\boldsymbol{x}^{0}$;

Step 4: Standardize by $\boldsymbol{z} = \boldsymbol{D}^{-1}\boldsymbol{z}^{0}$, where $\boldsymbol{D}\boldsymbol{D}^{T} = \boldsymbol{V}_{\boldsymbol{z}}$.

4/9/90

Woo-Jong Lee is with the Technical Center in Daewoo Motor Co., Korea. His principal research interests are in computer-aided design and manufacturing, solid modeling, and methods for analyzing and synthesizing tolerances in design and manufacturing. He received a B.S. degree in industrial engineering from Seoul National University in 1979, an M.S. degree in industrial engineering from Korea Advanced Institute of Science and Technology in 1981, and a Ph.D. in industrial and operations engineering from University of Michigan in 1988.

Tony C. Woo is Professor of Industrial and Operations Engineering at the University of Michigan, Ann Arbor, where he teaches courses in CAD/CAM. His research interest is in the development of geometric algorithms for design, manufacturing, and robotics.

Shuo-Yan Chou is a research assistant in the Department of Industrial and Operations Engineering at the University of Michigan, Ann Arbor. His research interest is in Computational Geometry and Tolerancing. He received a B.B.A. degree in industrial management science from Chen-Kung University in 1983, an M.S. degree in industrial engineering from University of Michigan in 1987, and he is currently working toward his Ph.D. degree in the Department of Industrial and Operations Engineering at University of Michigan.