



A Slope-Based Method for Least Cost Tolerance Allocation

Moiz Y. Nagarwala

Valeo Engine Cooling, 2669 Bond Street, Rochester Hills, MI 98309-3519

P. Simin Pulat* and Shivakumar Raman

School of Industrial Engineering, The University of Oklahoma, 202 West Boyd, Suite 124, Norman, OK 73019

Received 30 April 1995; accepted in revised form 27 September 1995

Abstract: In this paper, a new methodology is presented to solve the tolerance allocation–process selection problem simultaneously. The problem is modeled using discrete and continuous variables and is transformed into a model with only continuous variables by defining an efficient tolerance-cost curve for each component. Since the efficient tolerance-cost curve is neither convex nor concave, nonlinear programming methodologies cannot be directly applied. The tolerance-cost curve is piecewise linearly approximated and an efficient methodology is developed to solve the problem. The method starts with a solution which minimizes the objective function value but is not feasible. The infeasibility is iteratively reduced in a way that the increase in the objective function value is minimal. Computational analysis indicates that the method is very robust and requires negligible CPU time.

1. Introduction

Tolerance design requires a thorough understanding of manufacturing process selection and economics of processing, thus making it an important area of Concurrent Engineering. Specification of tolerances to mechanical assemblies has been widely studied in literature. The research on tolerance specification can be divided into two groups: tolerance analysis and tolerance allocation. Tolerance analysis is concerned with the calculation of the final tolerance from the information given on the component tolerances. Wu et al. [1] discuss eight tolerance analysis models cited in the literature. Tolerance allocation, on the other hand, is concerned with allocating component tolerances while observing the total assembly tolerance in a way to minimize total manufacturing cost. Generally, mathematical models are developed for the problem and solved using optimization techniques. Mathematical models are mathematical representations of the problem indicating the overall objective (which is generally the minimization of the total cost) and constraints of the problem. Review of literature indicates that mathematical models have been developed for simplified problems and these models have been solved using existing optimization techniques. Speckhart [2], Ostwald and Huang [3], Patel [4], Chase and Greenwood [5], and Lee and Woo [6] discuss tolerance allocation models that focus on the minimization of the total cost subject to either a constraint on the mean assembly tolerance requirement or a

constraint on the variance of the assembly tolerance distribution. Wu et al. [1] also discuss some of these models. Peters [7], Michael and Siddall [8,9], and Parkinson [10–12] demonstrate mathematically more complicated tolerance allocation models through examples. Many optimization models for the tolerance allocation problem tend to omit process selection constraints. However, process selection dictates the achievable tolerance range and affects the tolerance-cost curves. Prior selection of processes for each component of the assembly biases the allocation of tolerances and hence, may not lead to the overall optimum. Few researchers have studied the tolerance allocation problem with simultaneous process selection. However, most of the techniques developed for the problem concentrate on the changes in the structure of the underlying mathematical model due to the inclusion of the 0–1 variables for process selection. Hence, the existing approaches either use branch-and-bound technique to achieve the global optimum or heuristics such as simulated annealing or univariate search technique to achieve a local minimum solution. These approaches are discussed in more detail in the next section. In this paper, we consider the tolerance allocation with process selection problem and present an approach that is different from the existing techniques. We transform the problem which is a mixed integer optimization problem into a continuous optimization problem by defining a new cost-tolerance curve for each component and present a simple, yet efficient heuristic for its solution. The heuristic provides as good a solution as the ones generated by the existing approaches yet in almost no time.

*Author to whom correspondence should be addressed.

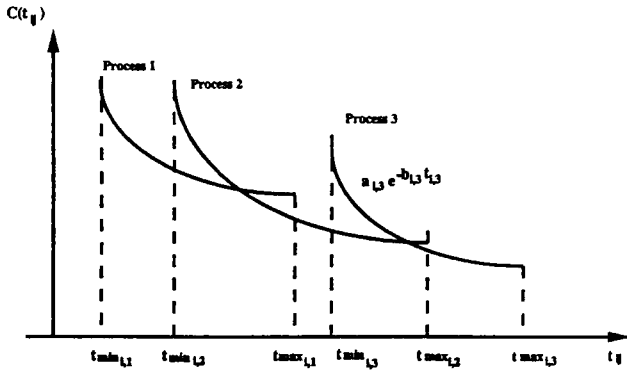


Figure 1. Process dependent tolerance-cost curves for component i .

Section 2 describes the tolerance-process selection problem in more detail and briefly discusses the existing methodologies. Section 3 presents the new algorithm. Computational performance of the algorithm is discussed in Section 4 followed by conclusions in Section 5.

2. Tolerance-Process Selection Problem

The tolerance-process selection problem addresses the selection of the best possible combination of processes for the different components of an assembly and allocation of tolerances to these components so that the required assembly tolerance is met with minimum processing cost. The phrase “the best combination of processes” refers to the combination which leads to minimum total cost tolerance allocation.

Consider an assembly with n components. Suppose, for each component i , there exists $p(i)$ many processes, any one of which can be used to manufacture the component. Assume an exponential tolerance-cost curve whose parameters depend on the process being considered. This is a generalized form of cost-tolerance curve used by many researchers in the past [2,13]. Let $t_{min,ij}$ and $t_{max,ij}$ be the minimum and the maximum tolerances achievable on component i using process j . Figure 1 illustrates process dependent tolerance-cost curves where t_{ij} is the tolerance assigned to component i manufactured using process j , and $c(t_{ij})$ is the corresponding cost. Let t_a be the specified assembly tolerance.

Define x_{ij} as:

$$x_{ij} = \begin{cases} 1 & \text{if process } j \text{ is selected for component } i \\ 0 & \text{otherwise} \end{cases}$$

The tolerance-process specification problem can then be mathematically modeled as:

$$\text{Min } Z = \sum_{i=1}^n \sum_{j=1}^{p(i)} x_{ij} a_{ij} e^{-b_{ij} t_{ij}} \quad (1)$$

subject to

$$\sum_{i=1}^n \sum_{j=1}^{p(i)} x_{ij} t_{ij}^2 \leq t_a^2 \quad (\text{statistical model}) \quad (2)$$

OR

$$\sum_{i=1}^n \sum_{j=1}^{p(i)} x_{ij} t_{ij} \leq t_a \quad (\text{worst case model}) \quad (3)$$

$$t_{min,ij} \leq t_{ij} \leq t_{max,ij} \quad \text{for all } i = 1, \dots, n \quad (4)$$

$$j = 1, 2, \dots, p(i)$$

$$\sum_{j=1}^{p(i)} x_{ij} = 1 \quad \text{for all } i = 1, 2, \dots, n \quad (5)$$

$$t_{ij} \geq 0 \quad \text{for all } i = 1, \dots, n$$

$$j = 1, \dots, p(i)$$

$$x_{ij} = 0 \text{ or } 1 \quad \text{for all } i = 1, \dots, n$$

$$j = 1, \dots, p(i)$$

The objective function Z gives the total processing cost for a given process combination and tolerance allocation. Constraints (2) and (3) indicate that the tolerance stack-up should not violate the tolerance requirement on the assembly. One can consider either (2) or (3) depending on the selected model (statistical, worst case). Constraints (4) indicate achievable tolerance range for each component and process combination. Constraints (5) ensure that only one process is selected for each component. (1) to (5) is a mixed-integer nonlinear programming model since t_{ij} s are continuous and x_{ij} s are binary decision variables and constraints (1) and (2) are nonlinear.

Since the above model contains integer decision variables, due to its combinatorial nature, any procedure adopted to generate the optimal solution will have exponential complexity, that is, the time required to reach optimality increases exponentially as the number of integer variables increases. Chase et al. [14] present three methods: exhaustive search, univariate search and sequential quadratic programming (SQP) to solve the tolerance-process selection problem. The exhaustive search method generates all feasible process combinations and uses a nonlinear programming technique to determine the optimal component tolerances for each process combination. The process combination that leads to the least total cost defines the optimal solution. While the exhaustive search technique achieves optimality, it is not practical since the number of process combinations increases exponentially as a function of the total number of

processes being considered. The univariate search method, on the other hand, fixes a process for each of the last $(n - 1)$ components and determines the best process for the first component by enumerating over the possible choices and solving a nonlinear programming problem to determine the optimal tolerances and the associated cost for each choice. Then, the process is fixed for the first component, and the last $(n - 2)$ components varied for the second component, and similarly, the best process is determined for the second component. The methodology repeats itself until the best process is determined for the n th component. An improved univariate search method is performed by Loosli [15] which repeats the optimization cycle until no improvement on any of the components can be made, that is, when a local minimum solution is reached. Although the univariate search method produces good results, it still has to solve a series of nonlinear programming problems to reach the result. Since the enumeration is not exhaustive, the resulting solution need not be the global minimum. Therefore, the time spent in optimization is not well justified. Similarly, the SQP technique relaxes the 0,1 restriction on x_{ij} s and considers them as continuous variables between 0 and 1. It solves an optimization problem with the intent of driving x_{ij} s to 0 or 1, which is not always possible. Zhang and Wang [16] report problems related to use for SQP for this problem.

Simulated annealing technique which has been shown to successfully solve discrete optimization problems is used by Cagan and Kurfess [17] to solve the tolerance-process optimization problem. The two simulated annealing approaches presented differ in the way that one solves an optimization problem for tolerances and the other varies tolerances randomly in the neighborhood of the old values. Zhang and Wang [16] have shown that the simulated annealing algorithm produces global or near-global solutions for moderate size tolerance-process selection problems. However, we believe that the algorithm still has to go through some process combinations which may be dominated by other combinations since all processes for all components are considered at all times.

The procedure presented in this paper generates an efficient tolerance-cost curve for each component and eliminates process selection for each component and, hence, eliminates the generation of process combinations. Its only limitation is that it assumes that there exists at least one process which can achieve tolerance $t_{min} \leq t_i \leq t_{max}$ where t_{min} and t_{max} are the minimum and maximum tolerances achievable considering all the processes combined. A further salient feature is that a nonlinear program does not need to be solved at each step which saves considerable time and computational effort.

3. The Slope-Based Method

As explained by Bjorke [18], a cost model for a combined

process means a plot of the manufacturing cost as a function of machining precision (tolerance) under the assumption that different processes yield different precision. For any given tolerance, there is only one manufacturing process that yields the lowest possible cost. A cost model for the combined process is shown in Figure 2. The cost function contains discontinuities at T_{23} and T_{12} "line up." The points of discontinuities define the tolerances at which two processes are equivalent as far as cost is concerned. The intervals between the break points define the economical range of individual processes. Process 1 should be used when the required tolerance is greater than T_{12} , process 2 should be used for a tolerance in the range between T_{12} and T_{23} , and process 3 should be used for a tolerance narrower than T_{23} . Thus, as shown by the solid curve in Figure 2, a single continuous and nondominating tolerance-cost function is obtained for each component. Similar types of curves exist for all other components in the assembly. This tolerance-cost function is neither concave nor convex.

After having obtained the above-mentioned tolerance-cost function for each component, these cost functions are approximated by a sequence of linear functions as indicated in Figure 3. To obtain higher precision, the linear function should be as close as possible to the original cost-tolerance curve. This can be achieved by linear approximation of the cost-tolerance curve over smaller intervals. Next, the slopes of each of the linear segments of the cost-tolerance curves for each component are calculated and the range of tolerance over which this slope is valid (i.e., the range over which each linear function is defined) is noted.

A pseudo-code for the slope-based method can be described as:

1. Begin by assigning maximum (very high) tolerances (t_i)

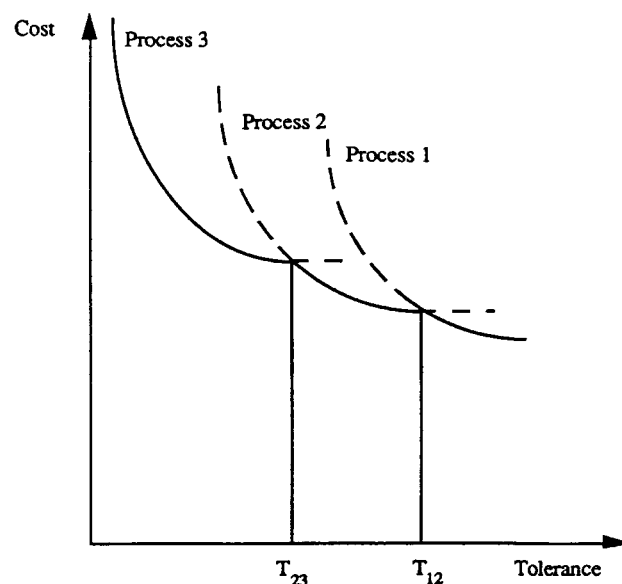


Figure 2. Cost model for a combined process.

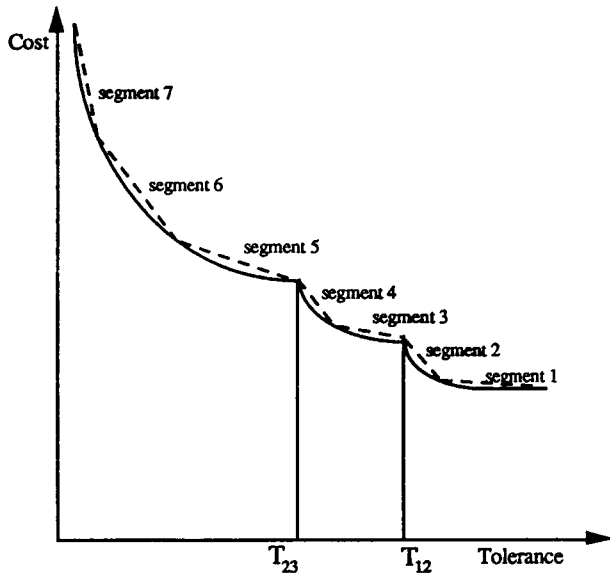


Figure 3. Sequence of linear approximations of the tolerance-cost curve of Figure 2.

- on each component using the minimum cost segment (i.e., segment #1)
- 2. Do the following while infeasibility exists (the solution is infeasible if $\sum t_i^? > t_a^?$ in case of statistical type of constraint or if $\sum t_i > t_a$ in case of worst-case type of constraint, and the infeasibility is $\alpha = \sum t_i^? - t_a^?$ for statistical type of constraint or $\alpha = \sum t_i - t_a$ for worst-case constraint).
 - 2.1 From among the tolerance-cost segments for each component, find the segment which gives the least increase in cost per unit decrease in tolerance.
 - 2.2 For the segment selected in 2.1, calculate the increase in cost that would result if the tolerance on this component were to be reduced by an amount equal to the range Δ over which that segment is defined. The infeasibility would decrease by amount $\delta = t_i^? - (t_i - \Delta)^2$ for statistical constraint or by an amount $\delta = \Delta$ for worst-case constraint; if $\delta > \alpha$ then calculate the cost increases for the corresponding reduction in tolerance that would result in reducing the infeasibility by an amount $\delta = \alpha$.
 - 2.3 For each of the remaining components, calculate the corresponding increase in cost that would result if the tolerance on each of these components were to be reduced by an amount that would lead to the same reduction in infeasibility (amount δ).
 - 2.4 For the component that leads to the least increase in cost, decrease the tolerance by the amount determined in either of steps 2.2 or 2.3 and select the corresponding tolerance-cost segment.
- 3. Even though the feasibility is achieved, from the present tolerance-cost segments for each component, select the one which would give the least increase in cost per unit

decrease in tolerance. For the corresponding component, calculate the increase in cost that would result if the tolerance were to be further reduced by an amount that would lead to another cost-tolerance segment. Also, for each of the remaining components, calculate the decrease in cost that would result if the tolerance on them were to be increased such that an exact feasible solution were to be obtained. If the net change in cost is negative, then the corresponding increase and decrease in tolerance on the selected components should be carried out. This process is repeated until there is no cost saving resulting from the above moves.

4. Computational Analysis and Discussion

Exhaustive Search, Univariate Search, Simulated Annealing (SA), and the Slope-Based (SB) methods were coded for comparisons. The optimization module in the first three methods used complex search technique [19]. The SA algorithm of Cagan and Kurfess [17] was found to give best results with annealing parameters set to $T = 100$, $f = 0.5$, and $iter = 10$; where T is the temperature, f is the reduction factor, and $iter$ is the maximum number of iterations to be carried out without any improvement in the objective function value. The SB method assumed the statistical type of constraint (constraint 2) on the assembly tolerance.

Our experimental results indicated that all methods exhibited similar performance for smaller size problems. As the number of components and the number of processes for each component was increased, the methods differed in the quality of the first solution generated. The SB method performed better than univariate search and simulated annealing methods with a CPU time only a fraction of the CPU time required for the other two methods.

For an assembly with twelve components and the number of processes

$$P(i) = \begin{cases} 6 & i = 1,2,4 \\ 5 & i = 3,5, \dots, 12 \end{cases}$$

for each component, Table 1 displays data for special assembly tolerance, the parameters of the exponential function, and the minimum and maximum tolerances achievable by each process for each part. The exhaustive search was terminated for the problem due to excessive CPU time. The slope-based method, simulated annealing, and univariate search method results are shown in Figure 4. The SB method performed better than the other two methods for the entire tolerance range. However, the univariate search method also generated very good solutions. The average CPU time varied between 2 to 15 seconds for the SB method, 20 to 140 seconds for the univariate method, and 1 to 73 seconds for the SA method. The authors recognize the fact that the SA version coded may not be the same as the one used by Cagan and Kurfess [17] since the parameters

Table 1. Input data.

Part #		Process #					
		1	2	3	4	5	6
1 and 4	coefs.— <i>a</i>	0.817	0.664	0.22	0.274	0.92	1.04
	— <i>b</i>	51.38	749.87	60.99	333.8	3360.47	3990.2
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	0.0001
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	0.0002
2	coefs.— <i>a</i>	0.35	0.547	0.64	2.29	5.25	5.683
	— <i>b</i>	63.08	211.62	310.58	1902	3976.15	4373.02
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	0.0001
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	0.0002
3	coefs.— <i>a</i>	0.466	2.187	1.565	5.783	5.693	
	— <i>b</i>	67.86	583.4	374.05	2008.09	1967.07	
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	
5 and 9	coefs.— <i>a</i>	0.51	0.71	0.84	5.31	9.82	
	— <i>b</i>	48.87	159.39	265.22	2568.8	4107	
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	
6 and 10	coefs.— <i>a</i>	1.51	2.24	3.534	4.77	7.68	
	— <i>b</i>	114.032	246.04	531.42	906.79	2097.12	
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	
7 and 11	coefs.— <i>a</i>	1.486	4.137	9.81	6.95	8.98	
	— <i>b</i>	125.49	480.96	994.06	563.496	1203.24	
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	
8 and 12	coefs.— <i>a</i>	0.526	13.465	4.13	4.186	8.68	
	— <i>b</i>	117.607	1198.11	459.656	476.12	2299.18	
	tol. range— <i>t</i> min	0.003	0.0016	0.0008	0.0004	0.0002	
	— <i>t</i> max	0.006	0.003	0.0016	0.0008	0.0004	

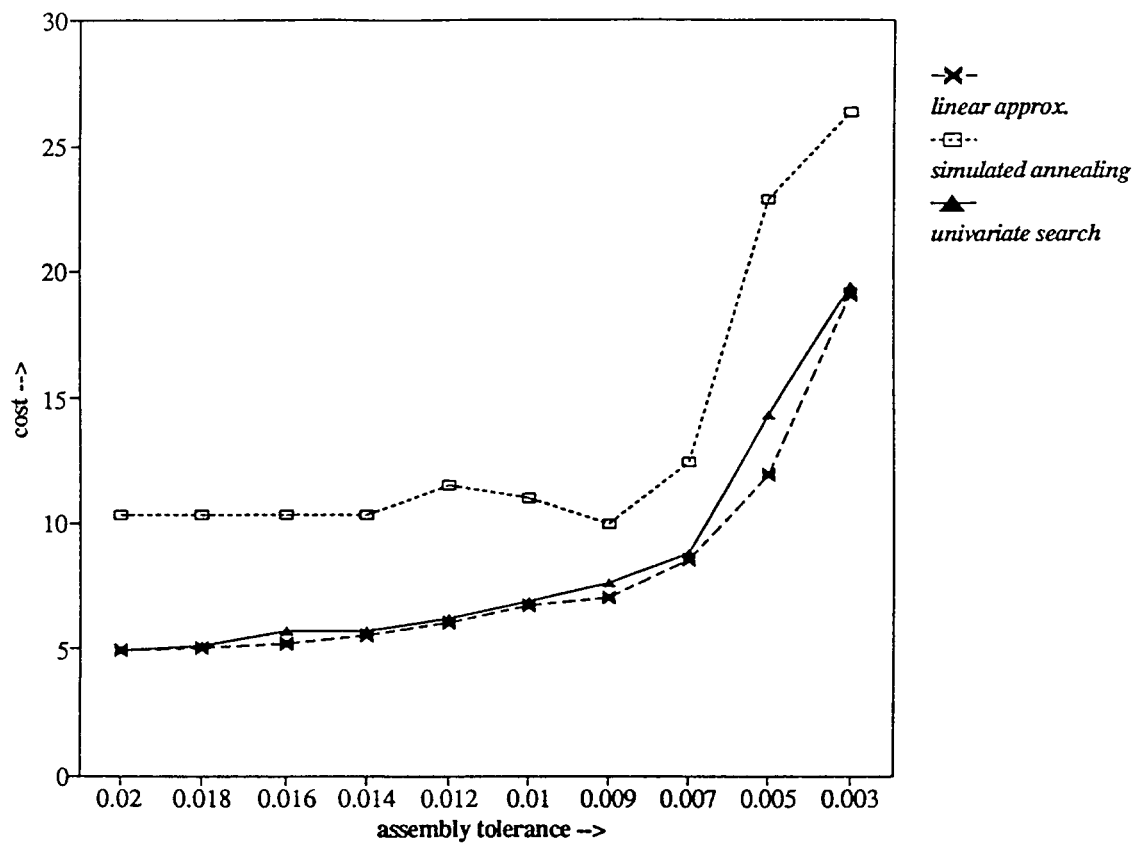


Figure 4. Cost vs. assembly tolerance for the example problem.

of the SA method were not explicitly stated in their paper. For this reason, we also ran the friction wheel example given by Cagan and Kurfess using the slope-based method code and compared with the SA results reported by Cagan and Kurfess. This example has four components forming an assembly and three different processes for each component. The processes have reciprocal cost-tolerance relationships. A worst-case stack-up of tolerances is assumed. Figure 5 shows component tolerances allocated by the algorithm as a function of the assembly tolerance. Table 2 contains detailed results. As the assembly tolerance (output tolerance) changes, processes allocated to components and allocated tolerances change. Initially, all four components exhibit tight tolerances. As the assembly tolerance is increased, tolerances on components 1, 2, 3, and 4 are increased as shown in the figure. Cagan and Kurfess have a similar figure obtained by the SA algorithm. Figure 6 gives the total cost value for each solution as a function of the assembly tolerance. The cost function is smooth except for tight tolerances. The cost curve is also very similar to the cost curve published in Figures 8, 9, 10, and 11 in their paper [17]. The CPU time for the SB method varied between a second to two seconds. The objective of the computational analysis was to verify that the SB method is simple to implement and that it generates very good solutions in negligible time. Therefore,

we used the SA approach for verification purposes rather than for comparisons of solution quality.

5. Conclusions

A new methodology (the slope-based method) is presented to solve the tolerance allocation process selection problem. The method is very simple and efficient. It differs from existing techniques in handling the process selection. A new tolerance-cost curve is defined using the process tolerance-cost curves for each component of the assembly. The new tolerance-cost function can be referred to as the efficient (or nondominating) tolerance-cost curve since each point on the cost curve refers to a unique process which is the most cost-effective process with the associated tolerance value. The procedure starts with $t_i = t_{max}$ for all $i = 1, \dots, n$. The solution may be infeasible, but it sets a lower bound on the total cost value. The procedure then reduces in feasibility with minimal increase in the total cost value. Once the feasibility is achieved, component tolerance values may be modified in a way that the total tolerance (stack-up) value equals the assembly tolerance, but leads to a reduction in the total cost value.

The SB method is verified against the univariate search

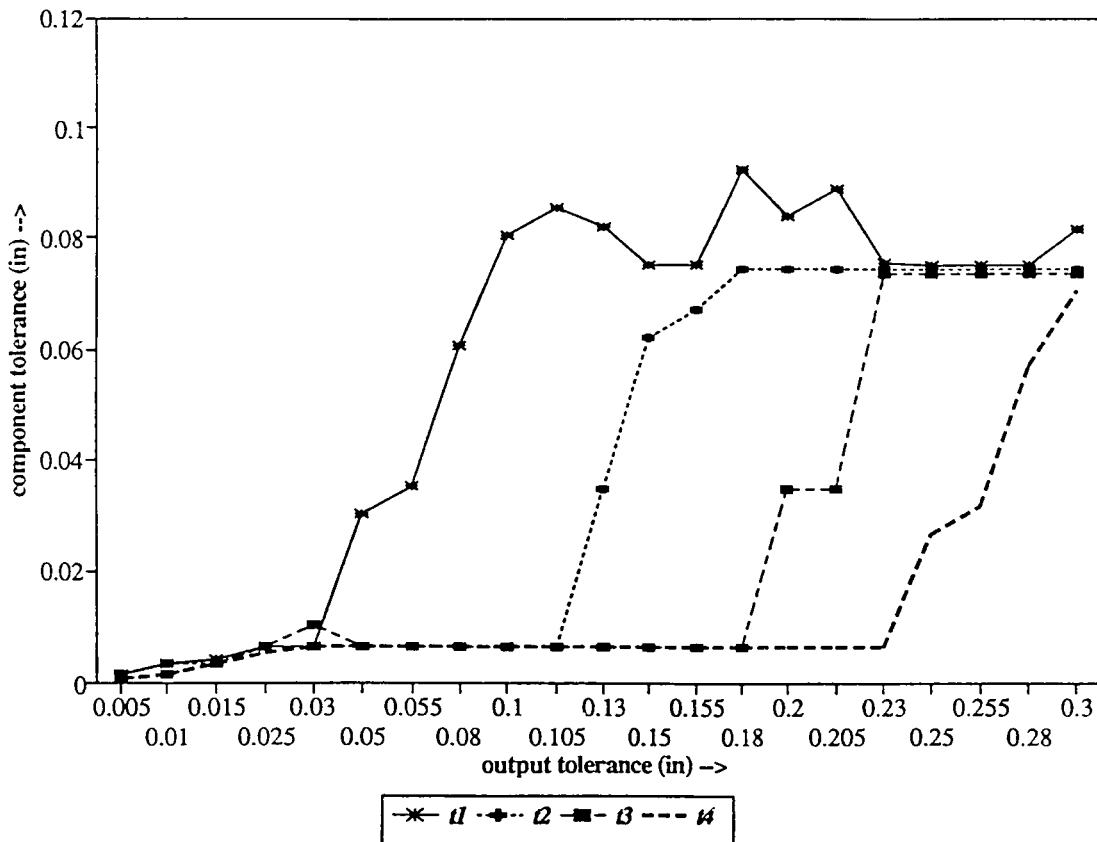


Figure 5. Component tolerance vs. output tolerance using the slope-based method.

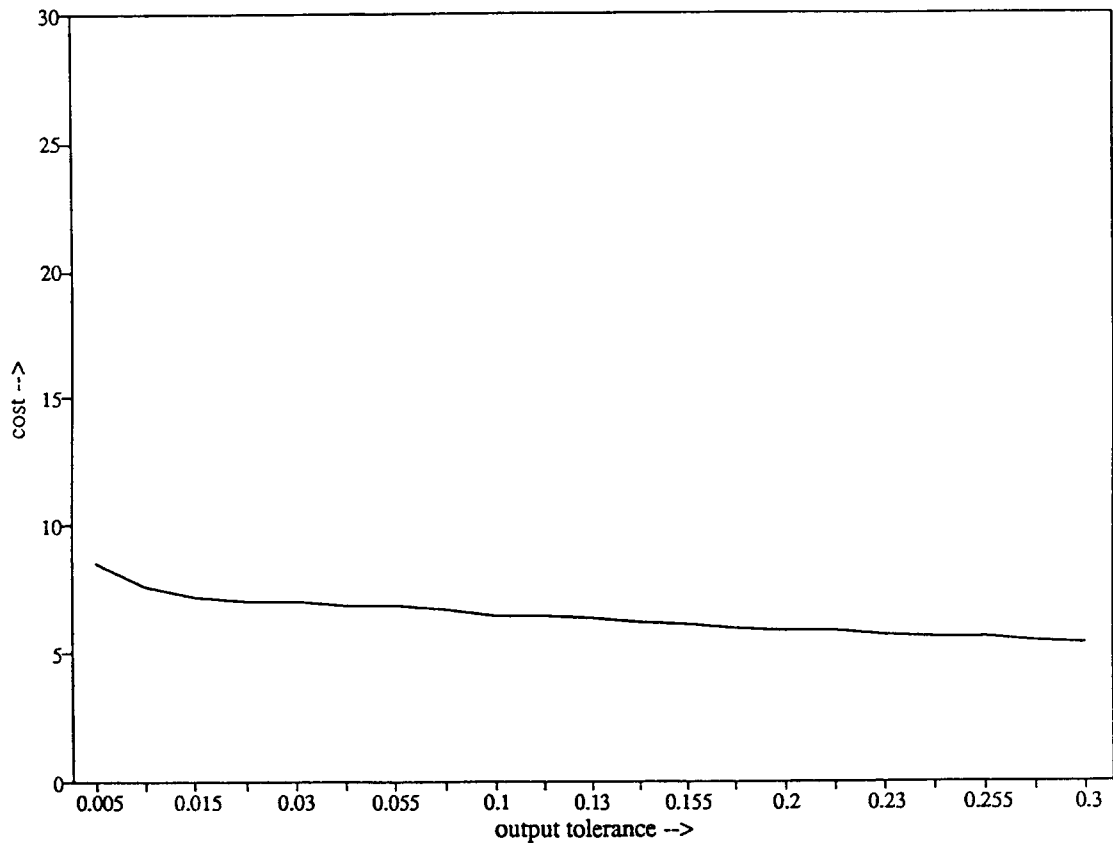


Figure 6. Optimal cost vs. output tolerance using the slope-based method.

Table 2. Solutions generated by the slope-based method.

T_s	Process Comps 1 through 12	Tolerances						Total Cost	Time (cpu secs)
		Comp 1	Comp 2	Comp 3	Comp 4	Comp 5	Comp 6		
2.00E-02	1-1-1-1-1-1-1-1-1-1-1-1	3.00E-03	6.00E-03	6.00E-03	5.57E-03	6.00E-03	6.00E-03	6.00E-03	2
1.80E-02	1-1-1-1-1-1-1-1-1-1-1-1	3.00E-03	3.00E-03	6.00E-03	3.00E-03	3.00E-03	3.00E-03	5.00	3
1.40E-02	1-1-1-1-1-1-1-2-1-1-1-1	3.00E-03	3.00E-03	3.00E-03	3.00E-03	3.00E-03	3.00E-03	5.51	3
1.60E-02	1-1-1-1-1-1-1-1-1-1-1-1	3.00E-03	3.00E-03	3.00E-03	3.00E-03	3.00E-03	3.00E-03	5.17	3
1.20E-02	1-1-1-1-1-1-1-1-1-1-1-1	3.00E-03	3.00E-03	3.00E-03	3.00E-03	3.00E-03	3.00E-03	6.04	4
1.00E-02	1-2-1-1-2-1-1-1-2-1-1-1	3.00E-03	1.60E-03	3.00E-03	3.00E-03	3.00E-03	1.60E-03	6.73	3
9.00E-03	3-2-1-2-2-1-1-1-2-1-1-1	8.00E-04	1.60E-03	3.00E-03	2.70E-03	1.60E-03	1.60E-03	7.05	4
7.00E-03	4-3-2-4-3-2-2-1-3-2-2-1	4.00E-04	8.00E-04	1.93E-03	4.00E-04	1.02E-03	1.02E-03	8.58	6
5.00E-03	4-3-3-4-3-2-2-2-3-2-2-2	4.00E-04	8.00E-04	8.00E-04	4.00E-04	8.00E-04	8.00E-04	11.98	9
3.00E-03	4-4-3-4-3-3-3-4-3-3-3-4	4.00E-04	6.32E-04	8.00E-04	8.00E-04	8.00E-04	8.00E-04	19.10	9
1.00E-03	6-5-5-6-4-5-5-4-4-5-5-4	1.00E-04	2.00E-04	2.00E-04	1.00E-04	4.00E-04	4.00E-04	40.97	14
9.00E-04	6-5-5-6-4-5-5-5-4-5-5-4	1.00E-04	2.00E-04	2.00E-04	1.00E-04	4.00E-04	4.00E-04	43.86	15

T_s	Tolerances						Total Cost	Time (cpu secs)
	Comp 7	Comp 8	Comp 9	Comp 10	Comp 11	Comp 12		
2.00E-02	6.00E-03	6.00E-03	6.00E-03	6.00E-03	6.00E-03	6.00E-03	4.88	2
1.80E-02	6.00E-03	6.00E-03	6.00E-03	6.00E-03	6.00E-03	6.00E-03	5.00	3
1.40E-02	6.00E-03	3.00E-03	3.00E-03	5.44E-03	5.94E-03	3.00E-03	5.51	3
1.60E-02	5.44E-03	4.69E-03	3.00E-03	6.00E-03	6.00E-03	6.00E-03	5.17	3
1.20E-02	4.08E-03	3.00E-03	3.00E-03	4.08E-03	4.68E-03	3.00E-03	6.04	4
1.00E-02	3.20E-03	3.00E-03	2.99E-03	3.20E-03	3.20E-03	3.00E-03	6.73	3
9.00E-03	3.12E-03	3.00E-03	1.60E-03	3.05E-03	3.13E-03	3.00E-03	7.05	4
7.00E-03	2.97E-03	3.00E-03	1.06E-03	1.88E-03	2.97E-03	3.00E-03	8.58	6
5.00E-03	1.60E-03	2.27E-04	8.00E-04	1.60E-03	1.60E-03	2.59E-03	11.98	9
3.00E-03	1.40E-03	7.21E-04	8.00E-04	8.00E-04	1.44E-03	7.21E-04	19.10	9
1.00E-03	2.00E-04	4.00E-04	4.00E-04	3.74E-04	2.00E-04	4.00E-04	40.97	14
9.00E-04	2.00E-04	2.64E-04	4.00E-04	2.00E-04	2.00E-04	4.00E-04	43.86	15

and the simulated annealing methods. The SB method performs as well as or better than the existing methods for large-size problems. It achieves the global solution or finds a near-global solution for the problems tested. The strength of the method lies in the fact that global or near-global optimal solutions were obtained in fairly negligible computational time for all of the test problems. The new method can be used in the real world to solve large-size problems (like an automotive assembly) wherein the assembly is made up of a large number of components, each having a number of alternate manufacturing processes to select from.

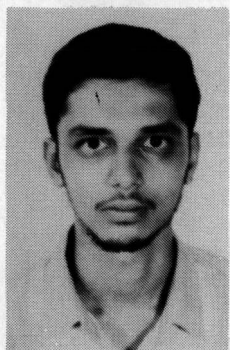
The method can be modified to solve general tolerance-cost curves where the tolerance-cost curves corresponding to the feasible processes for each component may be non-overlapping.

References

1. Wu, Z., W. H. Elmaraghy and H. A. Elmaraghy. 1988. "Evaluation of Cost Tolerance Algorithms for Design Tolerance Analysis and Synthesis," *Manufacturing Review, ASME*, 1(3): 168-179.
2. Speckhart, F. H. 1972. "Calculation of Tolerance Based on a Minimum Cost Approach," *Trans. ASME, Journal of Engineering for Industry*, pp. 447-453.
3. Ostwald, P. F. and J. Huang. 1977. "A Method for Optimal Tolerance Selection," *Trans. ASME, Journal of Engineering for Industry* (August):558-565.
4. Patel, A. M. 1980. "Computer-Aided Assignment of Manufacturing Tolerances," *Proceedings - Design Automation Conference*, Minneapolis: ACM IEE, June, 80:129-133.
5. Chase, K. W. and W. H. Greenwood. 1988. "Design Issues in Mechanical Tolerance Analysis," *Manufacturing Review*, 1(1): 50-59.
6. Lee, W. J. and T. C. Woo. 1989. "Optimum Selection of Discrete Tolerances," *Trans. ASME, Journal of Mechanisms, Transmissions, and Automation in Design*, 111:243-251.
7. Peters, P. 1970. "Tolerancing the Components of an Assembly for Minimum Cost," *Trans. ASME, Journal of Engineering for Industry*, 92(3):677-682.
8. Michael, W. and J. N. Siddall. 1981. "The Optimization Problem with Optimal Tolerance Assignment and Full Acceptance," *Trans. ASME, Journal of Mechanical Design*, 103(4):842-848.
9. Michael, W. and J. N. Siddall. 1982. "The Optimal Tolerance Assignment with Less Than Full Acceptance," *Trans. ASME, Journal of Mechanical Design*, 104:855-860.
10. Parkinson, D. B. 1982. "The Application of Reliability Methods to Tolerancing," *Trans. ASME, Journal of Mechanical Design*, 104:612-618.
11. Parkinson, D. B. 1984. "Tolerancing of Component Dimensions in CAD," *Computer Aided Design: CAD*, 16(1):25-32.
12. Parkinson, D. B. 1985. "Assessment and Optimization of

- Dimensional Tolerances," *Computer Aided Design: CAD*, 17(4):191-199.
13. Dong, Z. and A. Soom. 1990. "Automatic Optimal Tolerance Design for Related Dimension Chains," *Manufacturing Review*, 3(4):262-271.
 14. Chase, K. W., W. H. Greenwood, B. G. Loosli and L. F. Hauglund. 1990. "Least Cost Tolerance Allocation for Mechanical Assemblies with Automated Process Selection," *Manufacturing Review*, ASME, 3(1):49-59.
 15. Loosli, B. G. 1987. "Manufacturing Tolerance Cost Minimization Using Discrete Optimization for Alternate Process Selection," M.S. thesis, Brigham Young University, Provo, Utah.
 16. Zhang, C. and H. P. Wang. 1993. "The Discrete Tolerance Optimization Problem," *Manufacturing Review*, ASME, 6(1): 60-71.
 17. Cagan, J. and T. R. Kurfess. 1992. "Optimal Tolerance Allocation over Multiple Manufacturing Alternatives," *Advances in Design Automation*, ASME, 2:165-172.
 18. BJORKE, O. 1992. *Computer-Aided Tolerancing*. New York, NY: ASME Press.
 19. Reklaitis, G. V., A. Ravindran and K. M. Ragsdell. 1983. *Engineering Optimization Methods and Applications*. Wiley-Interscience.

Moiz Y. Nagarwala



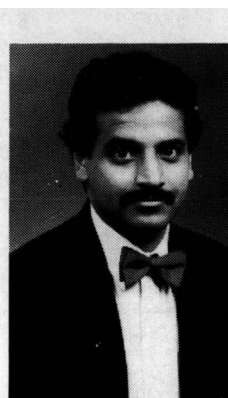
Moiz Y. Nagarwala has a Bachelor's degree in Mechanical Engineering from the College of Engineering, Pune and has completed his Master of Science degree in Industrial Engineering from the University of Oklahoma, Norman. He is currently working with Valeo Engine Cooling as Applications Engineer, primarily involved with the design and selection of cooling systems for automotive applications. He is also working toward his MBA degree from Wayne State University.

P. Simin Pulat



Dr. Simin Pulat is an Associate Professor in the School of Industrial Engineering at the University of Oklahoma. She has a B.S. degree in Industrial Engineering from Middle East Technical University, Turkey, M.S. and Ph.D. degrees in Operations Research from North Carolina State University, Raleigh, N.C. Her main research interests include optimization techniques and integration of optimization techniques to design and manufacturing problems. She is a full member of Institute of Industrial Engineers and Institute of Operations Research and Management Science.

Shivakumar Raman



Shivakumar Raman is an Associate Professor and Graduate Liaison of Industrial Engineering at the University of Oklahoma (OU). Dr. Raman received his B.E. (Mechanical Engineering) from the Shivaji University, India, his M.S. (Mechanical Engineering) from the University of Texas at Arlington and his Ph.D. (Industrial Engineering) from the Pennsylvania State University. He is a member of ASEE, ASME, IIE, SAE, and SME and an elected member of NAMRI/SME and Alpha Pi Mu. He is a recipient of the Clyde A. Sluhan Outstanding Young Manufacturing Engineer Award from SME (1994), Ralph R. Teetor Educational Award from SAE (1995) and two Engineering Excellence Distinguished Lecturer Awards from the College of Engineering at OU (1993-1994 and 1994-1995). He teaches courses in manufacturing processes and computer aided manufacturing and conducts research in modeling for machining, integrated manufacturing, tolerances, and automated inspection.