The Visual Computer

Tolerances in computer-aided geometric design

Joshua U. Turner[1] and Michael J. Wozny[2]

[1] IBM Corporation, P.O. Box 390, Poughkeepsie, NY 12602, USA
[2] Center for Interactive Computer Graphics, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, USA

In the design of discrete part shapes, the specification of tolerance constraints can have major consequences for product quality and cost. Traditional methods for tolerance analysis and synthesis are time-consuming, and have limited applicability. This paper presents the results of research into the use of solid modeling technology for the automated solution of tolerancing problems. A linear programming method is presented for the solution of tolerance analysis problems on a worst-case basis. A Monte Carlo method is presented for both worst-case and statistical tolerance analysis. Both methods automatically derive all necessary geometric relationships from a solid model of the assembly. Example problems are solved using the experimental GEOTOL geometric design system.

Key words: Tolerances – Geometric modeling – Solid modeling – Computer-aided design

1 Introduction

When an engineer designs an assembly, he specifies the overall shape and dimensions of each part. The part specification is often referred to as the "nominal" part. It is an idealization, since it is recognized that no possible manufacturing process is capable of producing the part exactly as specified, without variation. Therefore tolerances are also specified, which establish limits on the allowable variation. The tolerance constraints are intended to control part variation closely enough that any assembly of in-tolerance parts will satisfy the relevant design constraints. Thus the tolerance constraints provide an indirect means of enforcing the design constraints.

The assignment of actual values to the tolerance limits has a major influence on the overall cost and quality of the assembly. If the tolerances are too "tight" then the individual parts will cost more to make. If the tolerances are too "loose" then an unacceptable percentage of assemblies may be rejected, or require rework.

The process of allocating tolerances to an assembly is referred to as "tolerance synthesis." The process of checking the tolerances of verify that all design constraints will be met is referred to as "tolerance analysis." Traditional methods for tolerance analysis and synthesis are time-consuming, and error prone. For complicated problems they are usually infeasible.

Recently, there has been a growing interest in the use of solid modeling technology to provide a common geometric data base for design, analysis, and manufacturing. Already solid modeling systems have been used to automate many engineering tasks.

This paper gives results to date of a research project undertaken as a joint activity between Rensselaer Polytechnic Institute and IBM, devoted to the automated solution of tolerancing problems. The basic approach has been to use solid modeling technology to assess the effect of each possible part geometric variation on the tolerance constraints and design constraints of the assembly. Thus all required geometric relationships are automatically derived from a solid model. Automated methods for tolerance analysis based on linear programming and Monte Carlo procedures are presented here. These methods are illustrated using an experimental geometric design system, GEOTOL, developed at RPI and at IBM. Automated methods for tolerance synthesis are presented in Turner [16].

Several previous attempts have been made to use a geometric model as a basis for automated toler-

The Visual Computer (1987) 3:214–226

ancing. Hillyard [8] outlined a proposal for automated tolerance analysis. Work at MIT, such as Lin [11], and Light and Gossard [12], extended some of Hillyard's ideas. Grossman [6] described a basic approach to tolerance analysis based on Monte Carlo simulation. Requicha [15] gave a theory of tolerances based on offset surfaces. None of this work led to an operational capability for automated tolerance analysis or synthesis. Several packages [4], [17], [18], [1] were developed for tolerance analysis, which however do not appear to make use of a complete geometric model. Instead, it appears that the designer is required to identify the significant dimensions affecting to given design constraint, and to specify their relationships. Finally, there is a large body of literature on tolerancing methods that are either entirely manual or computer extensions of manual methods. Examples are Fortini [5], Bjorke [2], and Parkinson [13].

2 Definitions

Much of the conceptual confusion to be found in prior work on tolerances stems from a lack of well-defined terminology. Apart from the material on solution basis, the definitions given in this section are new to this work, and are intended to provide a unifying conceptual framework for discussion.

2.1 Solution basis

When universal interchangeability is an assembly requirement, tolerance problems must be solved on a *worst-case* basis. This means that all possible combinations of in-tolerance parts must result in an assembly that satisfies the design constraints. However in most cases the likelihood of a worst-case combination of parts is very low. When tolerance problems are solved on a *statistical* basis, manufacturing costs are reduced by loosening up the tolerances, and accepting a calculated risk that the design constraints may not be satisfied 100 per cent of the time. By assuming a probability distribution for each toleranced measurement, it is possible to determine the likelihood that the specified design limits will be exceeded. Effectively, a reject rate is determined for the assembly. A non-zero reject rate may be preferable to an increase in individual part manufacturing costs due to tighter tolerances. Both the worst-case and the statistical approaches are important in practice.

2.2 Design variables

The process of assigning tolerances to an assembly of parts begins with the establishment of design constraints. A design constraint is a limit (or pair of limits) that is explicitly specified by a designer over some aggregate geometric property of an assembly. For instance, the clearance between two parts in an assembly is an aggregate property determined by the individual part dimensions. Likewise, the volume of a container is an aggregate property determined by the dimensions of the container. Design constraints specify acceptable limits for such aggregate properties. [1] If an actual instance of an assembly of parts violates one of these design constraints it will be unacceptable.

A *design variable* will represent the variation from nominal of such an aggregate property. The designer will be viewed as indirectly identifying one or more design variables, whenever a design constraint is specified. The design constraint establishes limits on the allowable value of the design variable. [2]

For any given instance of the physical assembly of parts, the constrained aggregate properties can be measured, and the value of each design variable can be determined. An assembly will be considered "in-design" if all of its design variables fall within the limits specified by the design constraints.

Figure 1 shows a design variable called out on a simple part. D_1 measures variations in the area of the part.

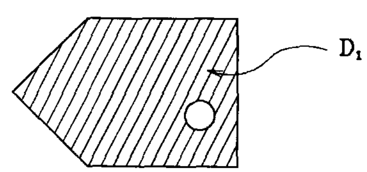

Fig. 1. Example of a design variable

[1] In more restricted contexts some authors refer to design constraints as "sum tolerances," or "assembly tolerances"

[2] The term "design variable" is sometimes used in the context of design optimization theory to refer to one of the parameters of a design, such as the value of a dimension, which may be treated as a variable for purposes of design optimization. That usage and the present usage are not related

2.3 Tolerance variables

A tolerance constraint is a limit (or pair of limits) that is explicitly specified by a designer over some one-dimensional geometric property of a single part. By implication, the toleranced property is assumed to be capable of direct control by the manufacturing engineer.[3] For instance, tolerances may be established on the position and diameter of a cylindrical hole, or on the distance between two parallel planar faces.

The tolerances assigned by the designer have no intrinsic significance. They are specified as an indirect means of enforcing the design constraints. For instance, the clearance between two parts in an assembly can only be controlled by setting limits on the dimensions of each of the individual parts. Likewise, the volume of a container can only be controlled by setting limits on the dimensions of the container.

A *tolerance variable* will represent the variation from nominal of some toleranced property. As with design variables, the designer will be viewed as indirectly identifying one or more tolerance variables whenever a tolerance constraint is specified. The tolerance specification establishes limits on the allowable values of the tolerance variable.

For any given instance of the physical part, the toleranced property can be measured, and the value of the associated tolerance variable can be determined. A part will be considered "in-tolerance" if all of its tolerance variables fall within their tolerance limits. Tolerance analysis and synthesis procedures must assure that all (or most) combinations of in-tolerance parts result in an in-design assembly.

Figure 2 gives an example of several tolerance variables called out on a simple part. T_1 measures variations in the distance of the left vertex from the right edge. T_2 measures variations in the angle between two sides. T_3 measures variations in the position of a hole. (The nominal positon of the hole must be established with reference to datum features of the part, for instance two of its perpendicular sides. This is not shown.)

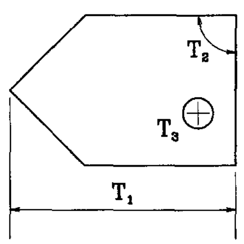

Fig. 2. Example of several tolerance variables

[3] This definition corresponds to what some authors refer to as a *component* tolerance

2.4 Variational model

If the nominal design model of a mechanical part (as embodied in a CAD system data base, for example) is compared with an actual instance of the physical part, the nominal model may be recognized as an abstraction of certain part geometric properties. This abstraction explicitly models certain characteristics, while others are ignored. For the solution of tolerancing problems, instead of a single abstract model of a nominal part, it is necessary to model certain types of part variations, thus defining a *variational class* of part instances.

2.5 Model variables

A variational model will be constructed from the nominal model, by introducing specified types of variations. Each allowable variation will be represented by a *model variable*. A model variable is a real-valued measure of the extent to which some elementary geometric property of a given part varies from nominal. The variational model will be comprised of the nominal part model augmented by a specified collection of model variables.

This paper will consider variations in part size, in relative orientation of part features, and in location of part features.[4] These variations are a sufficient characterization of actual manufactured parts for most purposes.

There are many possible strategies for defining model variables. Figure 3 illustrates one approach. This figure shows a complete set of model variables for a simple part, for a variational class comprising size, orientation, and position variations. In this example, the first ten model variables are paired about the outer boundary of the part. For exam-

[4] These, together with form variations, are the major classes of variations addressed by the ANSI dimensioning and tolerancing standard [3]. Considerations relative to form variations are presented in Turner [16]

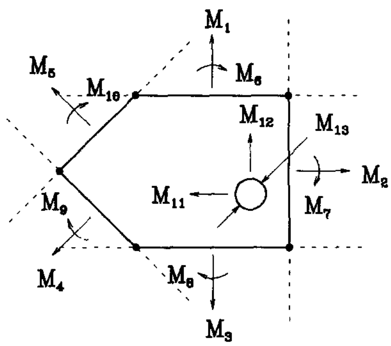

Fig. 3. Example of a set of model variables

ple, M_1 and M_6 induce variations in the coefficients of the line equation of the top edge of the part. M_1 induces variations in the position of the line, and M_6 induces variations in the slope. The vertex coordinates are implicitly determined by the line intersections. M_{11} and M_{12} induce variations in the position of the circular feature, and M_{13} induces variations in its diameter.

A given strategy for associating model variables with the nominal model effectively defines a variational class. For any given element of such a class, the values of the tolerance variables and design variables may be computed. Thus, the tolerance variables and design variables are effectively functions of the model variables.

While the tolerance variables and design variables are all functions of the model variables, it is ultimately the functional dependence of the design variables on the tolerance variables that is of concern to the designer. These functional relationships are illustrated in Fig. 4. Note that the function labeled **h** may not exist if insufficient tolerances are specified.

Since all three types of variables measure variations from nominal, they all take on values of zero for an assembly of nominal parts.

2.6 Example

Figure 5 shows an assembly consisting of two rectangular parts enclosed by a bracket. A design constraint specifies that the clearance between the rightmost rectangular part, and the inner edge of the bracket should vary from nominal by no more than 0.3 mm. A design variable is associated with the variation in the clearance, and the design constraint is expressed as:

$$-0.3 \leq D_1 \leq 0.3$$

To enforce this constraint, tolerance limits of ± 0.1 mm are applied to three of the part dimensions. Tolerance variables are associated with the variations in these dimensions, and the tolerance constraints may be expressed as:

$$-0.1 \leq T_1 \leq 0.1$$
$$-0.1 \leq T_2 \leq 0.1$$
$$-0.1 \leq T_3 \leq 0.1$$

Note that all these variables measure only variations from nominal – the nominal dimensions are not included.

Model variables can be specified to establish a variational model for each of the parts. To simplify the example, suppose that all form and orientation variations are disregarded (all edges remain either horizontal or vertical), and that only size variations acting in the horizontal direction are allowed. Figure 6 shows a collection of five model variables, which are sufficient to characterize the permitted variations. In this figure, the five model variables should be interpreted as follows. The left side of

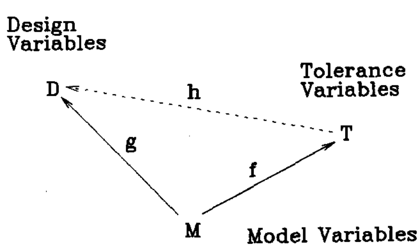

Fig. 4. Summary of relationships among the three variable types

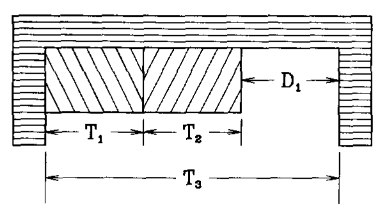

Fig. 5. A simple assembly

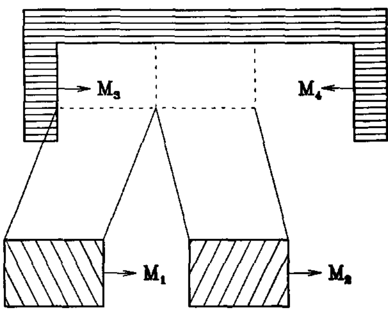

Fig. 6. Associated model variables

each part establishes a frame of reference for the part. Each model variable measures the variation from nominal of one of the other sides of its part relative to this frame of reference. Assigning a positive value to a given model variable causes a translation of the corresponding side in the direction of the arrow.

The tolerance and design variables may be expressed as functions of the five model variables:

$T_1 = M_1$
$T_2 = M_2$
$T_3 = -M_3 - M_4$
$D_1 = -M_1 - M_2 - M_3 - M_4$
(M_5 has no effect.)[5]

By further restricting the problem, it is possible to use a graph to illustrate the relationships among these variables. Specifically, if all variations in the bracket are ignored, that is if M_3, M_4, and M_5 are all assumed equal to zero, then the remaining model variables, M_1 and M_2, may be treated as independent dimensions of a vector space. The tolerance limits establish bounds on an "in-tolerance" region of this space. Similarly, the design limits establish an "in-design" region. These bounds are shown in Fig. 7. If all other variations are ignored, then there is a one-to-one correspondence between the points of this vector space and the set of all possible assembly instances.

The in-tolerance region of the diagram is that portion of the vector space admitted by the tolerance constraints. So the point-set defined by the in-tolerance region gives a mathematical representation

[5] As already explained, each of the three types of variables measures only the variation from nominal. Thus, for instance, the equation for D_1 states that any increase in M_1 M_2 M_3 or M_4 causes a decrease in D_1

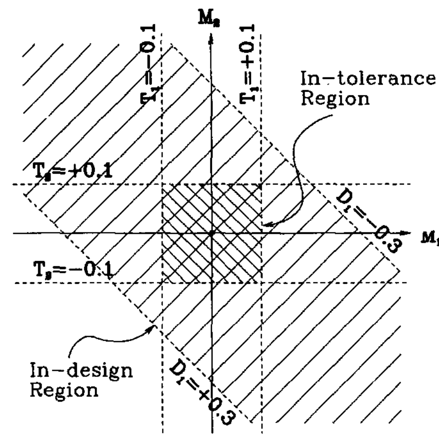

Fig. 7. Vector space of model variations

of the tolerance specification. Vector space representations have been constructed for a variety of representative tolerance types (Turner [16]).

Similarly, the in-design region gives a mathematical representation of the design constraint. Since the in-tolerance region is contained within the in-design region, the tolerance limits are adequate to satisfy the design constraint on a worst-case basis.

3 Tolerance analysis methods

From a formal viewpoint the tolerance analysis problem is as follows: given a specification of the tolerance constraints and design constraints for an assembly, find the relationship between the corresponding in-tolerance region and in-design region. For a *worst-case* tolerance analysis, the in-tolerance region must fall entirely within the in-design region. For a *statistical* tolerance analysis, a probability distribution is assumed for the vector space of model variations. The probability associated with that portion of the in-tolerance region falling outside the in-design region is computed. This probability must not exceed a previously established limit.

This section presents a linear programming method and a Monte Carlo method for automated tolerance analysis. Both methods are constructive: first, values are assigned to the model variables and used to construct geometric models of the corresponding part instances; next, the parts assembly sequence is simulated to construct a model of an assembly instance; finally, the values of the toler-

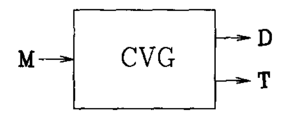

Fig. 8. Function of constructive variational geometry

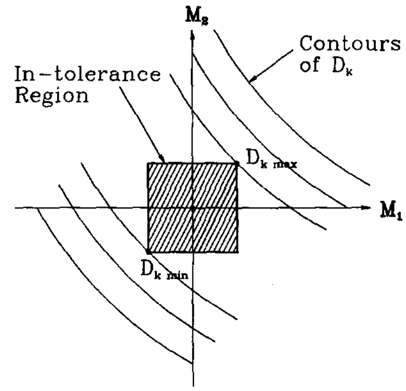

Fig. 9. Tolerance analysis by optimization methods

ance variables and design variables are measured from the assembly instance. A constructive geometric design system with variational capabilities is used to perform these operations. Such a design system will be referred to as providing capabilities for "Constructive Variational Geometry" (CVG). The schematic given in Fig. 8 summarizes the CVG function.

The GEOTOL geometric design system provides a working CVG capability. GEOTOL can generate model variations in the size, orientation, and position of part features. It supports a variety of both dimensional (plus-minus) tolerances, and the recently-developed ANSI geometric tolerances [3]. The implementation of the CVG function in the GEOTOL system is described in Turner [16].

The two methods presented in this section may be described as automatic. The designer must specify the tolerance constraints and design constraints, but explicit functional expressions for the tolerance variables and design variables are not required. All necessary geometric relationships are derived from the geometric model without designer guidance.

3.1 Tolerance analysis by optimization methods

One approach to the solution of the worst-case tolerance analysis problem is to find the actual limits of variation of each design variable permitted by the specified tolerance constraints. The actual limits of each design variable must fall within the specified limits for that variable.

The determination of the actual limits of variation of a design variable may be expressed as a constrained optimization problem. Figure 4 illustrated the functional dependency of design variables and tolerance variables on the model variables. In the terminology of Hillier and Lieberman [7], the model variables may be viewed as the *decision variables* of an optimization problem. The tolerance limits define constraints on the tolerance variables, and hence on the model variables. The design variables are considered one at a time, the goal being to find both the maximum and the minimum possible value for each design variable. The design variable is the *objective function* of the optimization problem. Figure 9 illustrates this graphically. Once both extremes of a given design variable have been determined, they can be compared against the specified limits. The tolerance specification is acceptable if and only if

$$d_{k_L} \le D_{k_{min}} \le D_{k_{max}} \le d_{k_U}. \qquad (1)$$

Here d_{k_L} and d_{k_U} are the specified design limits on D_k while $D_{k_{min}}$ and $D_{k_{max}}$ are the actual limits of variation of D_k for the given tolerance limits. D_k is given by

$$D_k = g_k(\mathbf{M}). \qquad (2)$$

The constraints on the feasible region are given by the tolerance limits:

$$t_{i_L} \le T_i \le t_{i_U} \quad i=1, \ldots, m$$

where

$$T_i = f_i(\mathbf{M}). \qquad (3)$$

3.2 Tolerance analysis by linear programming methods

Since the tolerance constraints permit only small variations in the model variables, the design variables and tolerance variables are usually approximately linear. If this is the case, then Eqs. (2) and

(3) may be linearized.[6] (In the GEOTOL system these linearizations are obtained using numerical methods.) Thus the following equations are obtained:

$$D_k = \nabla D_k \cdot \mathbf{M} \quad (4)$$

$$T_i = \nabla T_i \cdot \mathbf{M}. \quad (5)$$

Using these equations, the tolerance analysis problem may be solved using linear programming.

After the linear programming problem has been solved, a sensitivity analysis can be performed to determine the relative contribution of each of the tolerances.

If the tolerance variables and design variables are not linearizable, then it may be possible to apply nonlinear optimization methods.

3.3 Tolerance analysis by the Monte Carlo method

The Monte Carlo method is applicable whether the design variables are linearizable or not. Either a worst-case or a statistical analysis may be performed. This method operates by generating a large number of assembly instances. Each instance corresponds to a point in the in-tolerance region. The instance is checked to determine whether it also falls in the in-design region. Linearizations are required for the tolerance variables, but the design variables are measured from the varianced model. Unlike the optimization methods, already discussed, the Monte Carlo method allows all of the specified design variables to be analyzed simultaneously.

At each iteration of the method, the following steps are performed:

1. A point is selected within the in-tolerance region. For a worst-case analysis, this point should be at either the upper or lower limit of as many tolerances as possible. Therefore, a *corner-point* of the in-tolerance region is selected. For a statistical analysis, the selected point is statistically distributed within the in-tolerance region.
2. The model variables are set equal to the coordinates of this point.
3. The corresponding assembly instance is simulated, and the design variables are measured.
4. Statistics are collected.

The above procedure is repeated for many iterations. The actual number of instances to run is determined to achieve a given level of confidence in the results. The sample distribution of the design variable can be used as a gauge of the stability of the results.

It may perhaps be argued that for a large complex assembly, the dimensionality of the sample space is so great that a Monte Carlo analysis, even one based on a large number of assembly instances, does not carry a high degree of reliability as a predictor of actual manufacturing experience. Nevertheless a Monte Carlo analysis may act to focus attention on potential problem areas. Actually, in manufacturing practice, a prototype build-and-test run is often used for this purpose. The Monte Carlo method achieves the same end without the time and cost of parts fabrication.

4 GEOTOL implementation of CVG

The preceding section mentioned the role of Constructive Variational Geometry (CVG) in the tolerance analysis methods: once values have been assigned to the model variables, the CVG capability is used to evaluate the tolerance variables and design variables. First, a varianced model is constructed for each of the parts in the assembly. Next, the part models are combined to form a varianced assembly model. Finally, the tolerance variables and design variables are measured from the varianced assembly model.

The first step in this process is to apply the model variables to the individual parts. The GEOTOL system implements model variables corresponding to size, orientation, and position variations, in a manner similar to that suggested by Fig. 3. Size and orientation variations are applied as variations to the surface equations of the part faces. Position variations are applied to features of position. New edges and vertices are computed at the intersections of the varianced faces.

Once the model variables have been applied to the individual part models, the parts are assembled. To simulate the assembly process, a feature-based method is used to define the relative positions of the parts in the assembly. The GEOTOL system implements a positioning strategy in which each new part is positioned relative to a frame of refer-

[6] Although these functions can be highly nonlinear, and will usually involve square roots, sines, and cosines, the tolerance constraints limit the functions to a small region about the nominal, in which they are approximately linear

ence established by existing part features.[7] The modeling of feature relationships is a recent innovation. Lieberman and Wesley [10], and Lee and Gossard [9] have also done work in this area.

5 Examples

Several examples will be used to illustrate the use of the GEOTOL system for the solution of tolerance analysis problems by the linear programming and the Monte Carlo methods. The first example is a one-dimensional tolerance analysis problem that can also be solved by hand. However, in the second example, tolerances interact in all three dimensions, and a hand solution might require many hours of laborious calculations. The third example illustrates the analysis of a larger problem from a real product.

5.1 One-dimensional example

The model shown in Fig. 10, represents an assembly consisting of a U-shaped part with two small rectangular parts stacked against the left side of the U. A design constraint (bottom) has been specified for the clearance between the right side of the rightmost block, and the right side of the U-shape. Tolerances (top) have been specified for the pertinent part dimensions.

It is not necessary for the designer to specify any relationship between the given tolerance constraints and the design constraint. In fact, there may be many other tolerance and design constraints specified in addition to the ones shown here. The GEOTOL system will automatically determine which tolerance constraints have an influence on each design constraint.

When the variational class is limited to size variations, the linear programming method immediately computes values of ± 0.03 as the actual limits of variation of the design variable.

When the Monte Carlo method is applied, there are only two worst-case possibilities for each of the toleranced measurements (upper and lower limits of size). Thus, there are $2 \times 2 \times 2 = 8$ different possible assembly instances. A worst-case tolerance analysis shows that the design variable associated with the design constraint takes on values of -0.03, -0.01, $+0.01$, or $+0.03$ for these assembly instances. The results of the Monte Carlo analysis are shown in Fig. 11. The histogram shows the distribution of the design variable for the actual assembly instances sampled. The difference between actual maximum and minimum values of the design variable is divided into equal intervals. The histogram shows the frequency with which actual values of the design variable fall within each interval. The dashed lines show the values of the specified design limits.

[7] Relative positioning is also used to position features of position, such as holes and bosses, within a single part. Feature position is defined relative to a datum frame of reference using the method given in the ANSI geometric tolerancing standard [3]

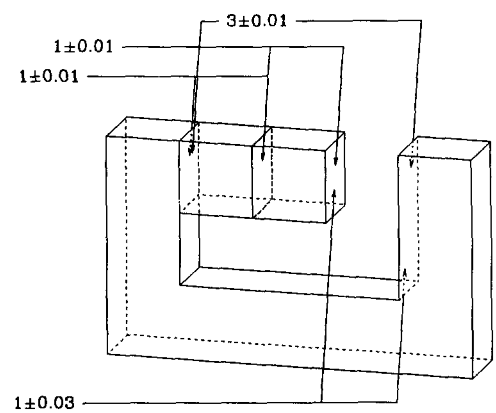

Fig. 10. Simple one-dimensional tolerance analysis problem

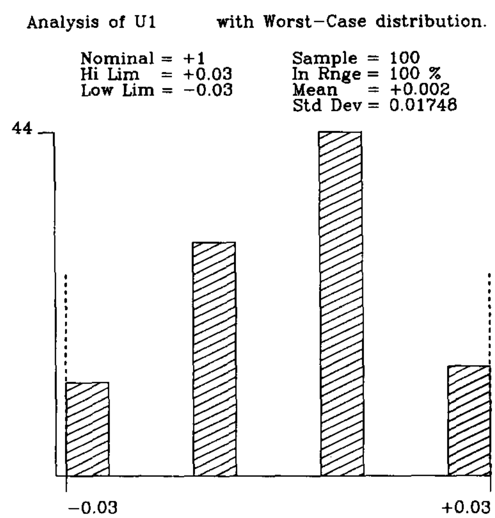

Fig. 11. Worst-case analysis of U-shape. Size variations only

In addition to generating the individual parts, the GEOTOL system must position each part in sequence relative to its adjacent parts, using the actual part features as reference.

The assembly can also be analyzed under statistical assumptions of uniform or normal distributions for each toleranced part. Figures 12 and 13 give these results. For both statistical analyses the sample distribution is assumed to be normal, and plotted using the computed sample mean and standard deviation.

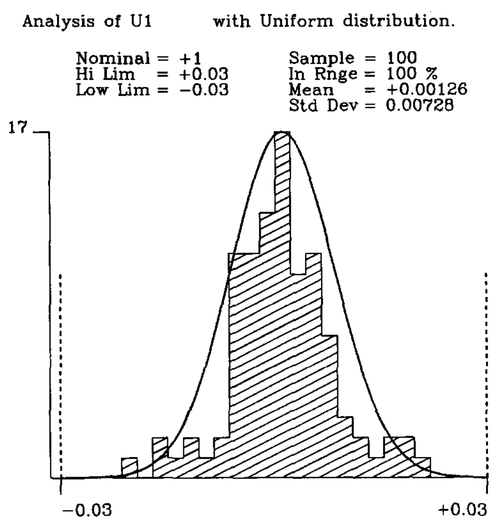

Fig. 12. Analysis of U-shape with assumed uniform distributions

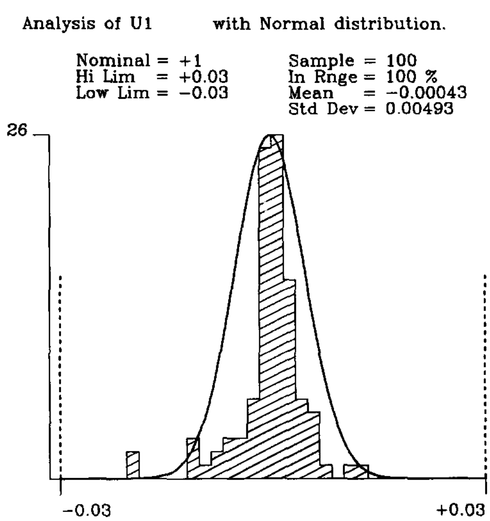

Fig. 13. Analysis of U-shape with assumed normal distributions

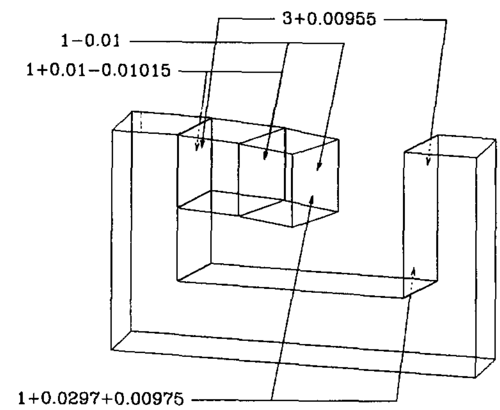

Fig. 14. Example of the effects of nonlinearities

The latter analysis shows that under the assumption of normal distributions, the tolerance limits might be relaxed considerably.

When the variational class is extended to incorporate orientation variations, nonlinearities become significant. When the linear programming algorithm is run with both size and orientation variations enabled, the actual limits of variation of the design variable are found to be:

$D_{min} = -0.0307$

$D_{max} = +0.0297.$

Figure 14 shows an instance drawn from this analysis, illustrating one of these extremes (variations are exaggerated). The numbers following the nominal dimensions are the actual values of the toleranced measurements. Since nominally parallel faces need not be parallel in the varianced model, there are two values for some of the toleranced measurements – a low value, and a high value. Nonlinearities in the tolerance variables have allowed some of the tolerance limits to be exceeded. The Monte Carlo method gives similar results.

It should be noted that this particular problem can also readily be solved by manual methods, since all of the measurements line up along a single dimension. Indeed, the worst-case results are obvious by inspection.

5.2 Three-dimensional example

A simple three-dimensional example with some surprising subtleties is found in the cuboid shown in Fig. 15. The width, height, and depth have been

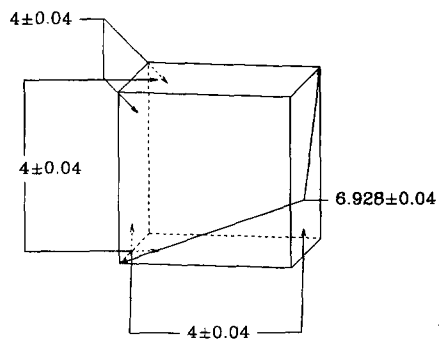

Fig. 15. Simple three-dimensional tolerance analysis problem

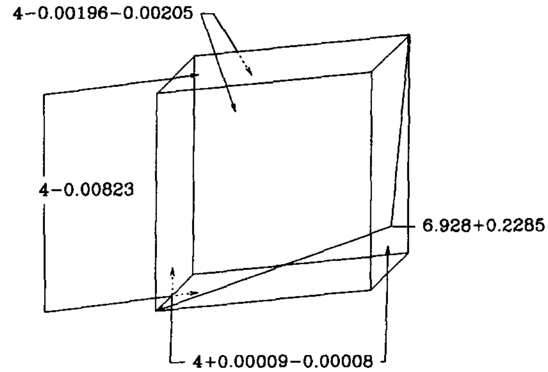

Fig. 16. Illustration that original design was undertoleranced

given equal dimensions, and 1% tolerances of ± 0.04. A design constraint has been specified on the distance between the two vertices at the endpoints of one of the major diagonals. Design limits of ± 0.04 have been imposed. (To motivate the design constraint, take the cuboid as a design for a packing crate, and assume that a shaft is to be packed along the diagonal. The design constraint is intended to assure that the shaft will fit snugly.)
If only size variations are generated, then the linear programming method gives values of ± 0.06928 as the actual worst-case limits of the associated design variable. The Monte Carlo method gives the same results. These results can be verified by realizing that if all three dimensions are taken to their lower limits, that is, reduced by 1%, then the diagonal will also be reduced by 1%, or 0.06928 (similarly for the upper limits). It is clear that under worst-case assumptions, the tolerance limits on the three dimensions are too loose to satisfy the specified design constraint of ± 0.04.
If orientation variations are generated as well as size variations, then the GEOTOL tolerance analysis exposes the fact that the design is actually is undertoleranced, and that there is no effective control over the variation in the diagonal. As illustration, Fig. 16 shows an instance of the cube that satisfies all the stated size tolerances, but that exhibits a variation of $+0.2285$ in the diagonal – over five times the specified design limit. Such variation is possible because the size tolerances, as defined in the ANSI geometric tolerancing standard [3], do not control the variation in the angle formed by adjacent faces. So long as opposite faces remain approximately parallel, the part remains in-tolerance. (The two-dimensional analog to such a part is rhomboid.) To establish control over the variation in the diagonal, the designer must specify perpendicularity tolerances between some of the adjacent faces. Therefore three perpendicularity tolerances of 0.04 were added to the model.[8] The perpendicularity tolerances are shown in Fig. 17. Now an analysis exploiting both size and orientation variations can be performed. The linear programming method gives worst-case limits of

$$D_{\min} = -0.1374$$
$$D_{\max} = +0.1397.$$

The Monte Carlo method gives

$$D_{\min} = -0.1360$$
$$D_{\max} = +0.1439.$$

The discrepancy between the two methods is due to nonlinearities.
So when perpendicularity tolerances were incorporated and both size and orientation variations were modeled, the actual limits of variation were found to be about twice the limits found earlier with only size variations modeled. In principle it would be possible for an engineer to work out a formula showing how variations in the three size dimensions and the three toleranced angles affect the diagonal. Once such a formula had been worked out, it could be used to determine the limits of varia-

[8] Three perpendicularity tolerances are sufficient provided one tolerance applies in each dimension. However this allows some angles more freedom of variation than others. To ensure equal treatment, the other possible perpendicularity tolerances could be specified. This illustrates a strength of the vector space treatment – additional tolerances simply provide additional constraints on the in-tolerance region. Although in a traditional sense the model is overtoleranced, no difficulty arises

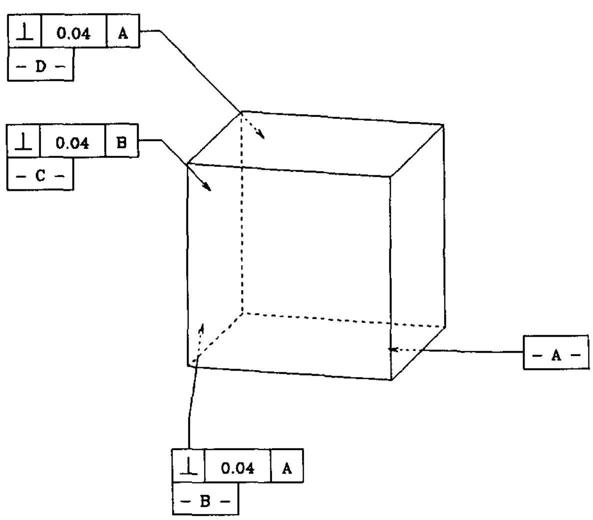

Fig. 17. Added perpendicularity tolerances

tion. However, even for this simple problem, the development of such a formula is infeasible. In larger problems, even recognizing which dimensions affect a given design constraint can be difficult. Therefore the ability to perform such an analysis without the prior hand derivation of the requisite geometric relationships is a significant benefit to the engineer.

5.3 Example from engineering practice

Figure 18 shows an example of an assembly from an actual product. This is a (simplified) model of a bus bar assembly from the IBM 3090 system. The six bus bars conduct current from the system's power supplies to the electronics modules. The bus bars attach to a clamp (clamp base shown at left). In addition, there are assembly constraints represented by the part shown at right. (This part is a simplification, taking the place of a number of distinct mating parts.) The principal design constraint for this assembly is that it assembles without interference. The assembly was modeled, and all specified tolerances were applied. A choice of strategies is available as to how to model the design constraint. One possibility would be to define a design constraint, with associated limits, for the clearance between each hole-boss pair. But since all of the fits in this particular assembly are clearance fits, it is possible to define a single design

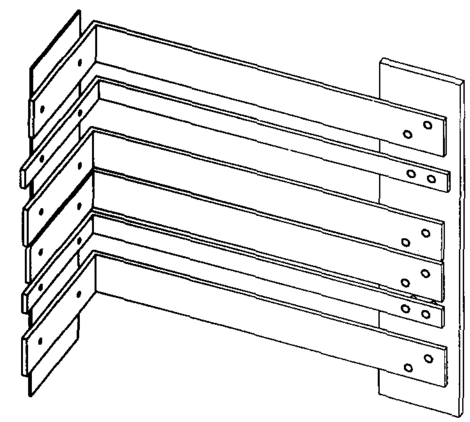
Fig. 18. Bus bar assembly

constraint on the overall fit of the entire assembly. This gives rise to a binary design variable, which evaluates to 1 if all the parts assemble without interference and to 0 otherwise.

Perpendicularity tolerances were entirely missing in the original part drawings. Thus, as the previous example showed, the design is undertoleranced, and at least three perpendicularity tolerances are required per part, to fully tolerance the design. In most cases, the omission of these tolerances is based on assumptions about process characteristics. For instance, four right angles are revealed by taking a cross-section of one of the bus bars anywhere along its principal dimension. Variations in these angles have no effect on the design constraint. It may be assumed that any reasonable manufacturing process will produce acceptable angles. But if angular variation is not constrained, GEOTOL detects an undertoleranced design, and will not proceed. Considering the frequency of this type of situation in design practice, it would be helpful if the designer were able to specify a blanket angularity control to be used in the absence of an explicit tolerance specification.[9]

Despite the above remarks, the omission of a perpendicularity tolerance on the major bend in each of the bus bar parts appears to be an error. Variation in this angle could have a significant effect on assemblability. A discussion with the designers

[9] Such a blanket tolerance should not be confused with the so-called "default" tolerance frequently used in current drafting practice. The default tolerance only applies to features that are explicitly dimensioned. That is, a default tolerance is an explicit tolerance applied indirectly through explicit dimensions

revealed that they had assumed sufficient flexibility to permit bending the bars at assembly time.
Because of the above omissions, the analysis was run without orientation variations. A worst-case analysis of 100 instances showed that 100% assembled successfully. To fully validate the tolerance specifications, the missing perpendicularity tolerance on the major bend should be supplied, and orientation variations in that direction should be enabled.

5.4 Computer time used

Theoretical results as to computational complexity are given in Turner [16]. However a summary of the computer times for the preceding examples is also instructive. For both the linear programming and the Monte Carlo methods, the analysis step is preceded by an initialization step in which linearizations are computed for the design variables and tolerance variables. The following table shows the actual computer times used. These times were derived from the execution of the examples on a conventional uniprocessor machine (the IBM 4381).

Example	Initialization	Linear Programming	Monte Carlo (per instance)
U-Shape (size only)	2.31 s	0.71 s	1.32 s
Cuboid	1.83 s	1.69 s	0.81 s
Bus Bars	5 min 33 s	–	2 min 12 s

The per-instance execution time for the Monte Carlo analysis of the bus bar assembly is quite high. However as yet little has been done to optimize the performance. In addition, there is strong evidence that the algorithms can be structured to take advantage of vector processing computer architectures.

6 Summary

This paper has presented a unifying terminology for the interpretation of tolerance problems. Model variables are used to represent a variational class of part instances. Tolerance variables and design variables are used to measure variations in quantities constrained by the designer. A tolerance specification may be expressed as an in-tolerance region in a vector space defined by the model variables.
It was shown that the worst-case tolerance analysis problem may be formulated as an optimization problem and solved using linear programming. Alternately, a Monte Carlo method may be applied. The Monte Carlo method allows for nonlinear design variables and supports statistical as well as worst-case analysis, but is generally more expensive than the linear programming method. Both methods may be described as automatic, since all requisite geometric relationships are derived without designer guidance.
Principal areas for further research are as follows. First, significant nonlinearities can arise in tolerance analysis problems. It appears that it may be possible to address these nonlinearities using the method of successive linear programs [14] or other nonlinear methods. Second, models for worst-case and statistical tolerance synthesis were developed in Turner [16]. It appears that these models can be solved using convex programming methods, but experience is needed.
By contrast with the most mature areas of engineering analysis and synthesis, the automation of tolerancing problems is in its infancy. However, we believe this work has demonstrated the potential for a comprehensive automated tolerancing capability based on solid modeling technology, and has established promising directions for further work.

Acknowledgements are due to the IBM Corporation for its support of the research described herein, and particularly to Messrs. William Beausoleil, Gilbert Curl, and Floyd Skutnik.

References

1. AVSS (Assembly Variation Simulation System) (1986) Marketing Literature, John Deere, John Deere Road, Moline, Illinois 61265 (March 1986)
2. Bjorke O (1978) Computer-aided tolerancing. Tapir Publishers, Trondheim, Norway
3. Dimensioning and tolerancing (1983) ANSI Y14.5M-1982, American Society of Mechanical Engineers, New York
4. Emanuel R (1985) Establishing mechanical tolerances with CAD/CAM. Computer-aided engineering 4(7):76–80
5. Fortini ET (1967) Dimensioning for interchangeable manufacture. Industrial Press, New York
6. Grossman D (1976) Monte Carlo simulation of tolerancing in discrete parts manufacturing and assembly. Stanford Artificial Intelligence Laboratory Memo AIM-280 and Stanford University Computer Science Report No. STAN-CS-76-555. (May 1976)
7. Hillier FS, Lieberman GJ (1980) Introduction to Operations Research, Holden-Day, Oakland, California
8. Hillyard RC (1978) Dimensions and tolerances in shape design. PhD Dissertation, University of Cambridge, UK

9. Lee K, Gossard DC (1985) A hierarchical data structure for representing assemblies: part 1. Computer-Aided Design 17(1):15–19
10. Lieberman LI, Wesley MA (1977) AUTOPASS: An automatic programming system for computer controlled mechanical assembly. IBM J Res Dev 21(4):321–333
11. Lin VC, Gossard DC, Light RA (1981) Variational geometry in computer-aided design. SIGGRAPH '81 Conf Proc (August 1981), Comput Graph 15(3):171–177
12. Light R, Gossard D (1982) Modification of geometric models through variational geometry. Computer-Aided Design 14(4):209–214
13. Parkinson DB (1985) Assessment and optimization of dimensional tolerances. Computer-Aided Design 17(4): 191–199
14. Reklaitis GV, Ravindran A, Ragsdell KM (1983) Engineering optimization. John Wiley, New York
15. Requicha AAG (1983) Toward a theory of geometric tolerances. Int J Robotics Res 2(4):45–60
16. Turner JU (1987) Tolerances in computer-aided geometric design. PhD Dissertation, Rensselaer Polytechnic Institute (May 1987)
17. VSAS (Variation Simulation Analysis Software) (1984) Marketing Brochure, Applied Computer Solutions, 100 Maple Park Blvd., Suite 116, St. Clair Shores, Michigan 48080
18. The VSAS (Variation Simulation Analysis Software) Report (1985) Marketing Newsletter, Applied Computer Solutions, 100 Maple Park Blvd., Suite 116, St. Clair Shores, Michigan 48080, No 1 (November 1985)

Michael J. Wozny *(S'58–A'62) joined Rensselaer Polytechnic Institute, Troy, NY, in 1977 to establish the Center for Interactive Computer Graphics. He is currently on leave to the National Science Foundation as Division Director for the Design, Manufacturing, and Computer-Integrated Division. His previous appointments include Purdue University, Oakland University, GM Research labs, NASA Electronics Research Center and NSF. Dr. Wozny has served on a number of advisory boards, (OTA, CAD/CAM ALERT, WORKSTATION ALERT, Who's Who in Computer Graphics), was Chairman of a National Academy of Sciences panel which prepared a briefing document "Research Opportunities for Design and Manufacturing" was a member of the IEEE Computer Society Publications Boards, and a former Director of NCGA. He was the first Editor-in-Chief of IEEE* Computer Graphics and Applications, *and presently serves on the Editorial Boards of* The Visual Computer *and* IEEE Proceedings. *He is a Director of the Computer Graphics Society and a Director of two companies.*

Joshua Turner *is with the Solid Modeling Technology Department of IBM Corp., in Poughkeepsie, N.Y. His principal research interests are in computer-aided design, solid modeling, computer graphics, and methods for automating the use of tolerances in design.*
Turner received a Ph. D. in computer and systems engineering from Rensselaer Polytechnic Institute in 1987. He received an M.S.E. in computer and information science from the University of Pennsylvania in 1980, and a B.A. from Haverford College in 1972.
He is a member of IEEE, ACM, and ASME.