GENERAL SYSTEM FOR LEAST COST TOLERANCE
ALLOCATION IN MECHANICAL ASSEMBLIES

ADCATS Report No. 90-2

Carl Blaine Andersen, graduate student
Department of Mechanical Engineering
Brigham Young University

MS Thesis sponsored by ADCATS

July 1990

ABSTRACT

This research report describes the development of an efficient method for determining the least-cost component tolerances for mechanical assemblies. It combines a new optimization technique with statistical tolerance analysis to insure lowest production cost of an assembly. Called the Equalized Gradient method, it will give designers the ability to solve complex assembly problems with any of the following features:

1) Two-dimensional assemblies described by multiple vector loops and with dependent variables eliminated.
2) Cost-vs-tolerance functions described by a power law with different exponent values for each independent dimension in an assembly.
3) Mean shift values representing biased processes, which may be different for each component in the assembly.

Using this optimization tool, the designer will be able to include available manufacturing data in the initial design stages to assure the assembleability and performance of the final product and easily refine his design as more information becomes available.

The Equalized Gradient method is general enough that it may provide efficient solutions to other unrelated nonlinear problems.
TABLE OF CONTENTS

List of Figures iv
List of Tables v
Acknowledgements vi
1. INTRODUCTION 1
   1.1 Tolerance Stacking 1
   1.2 Tolerance Allocation 2
   1.3 Thesis Objective 3
   1.4 Outline of Thesis 3
2. PREVIOUS WORK 5
   2.1 Models and Solution Techniques 5
   2.2 Lagrange Multiplier Solution of Worst Case and Statistical Models 6
   2.3 Extending to 2-D and 3-D Assemblies 8
   2.4 Estimated Mean Shift Model 10
   2.5 Comparison of Previous Work to This Thesis 12
3. CLOSED FORM ESTIMATE OF LEAST COST SOLUTION 13
   3.1 Lagrange Multiplier Solution of Estimated Mean Shift Model 13
   3.2 Closed Form Estimate of Lagrange Multiplier Solution 13
   3.3 OPTDES Comparison 14
4. NUMERICAL SEARCH TECHNIQUES TO FIND THE OPTIMUM 18
   4.1 The Optimization Problem 18
   4.2 Development of a Search Technique 19
   4.3 Simplified Technique 23
      4.3.1 Step Size Improvement 26
      4.3.2 SQP and GRG Comparison 27
   4.4 Possible Extension to Multiple Constraints 27
5. SAMPLE PROBLEMS 30
   5.1 Geometric Stack Problem 30
   5.2 One-Way Clutch Problem 32

6. CONCLUSIONS AND RECOMMENDATIONS 35
   6.1 Contributions of this Thesis 35
   6.2 Future Work 36
       6.2.1 Multiple Constraints 36
       6.2.2 Three-Dimensional Problems 36
       6.2.3 Equalized Gradient Search Method 36
       6.2.4 Development of Valid Cost-Tolerance Functions 37

References 38

Abstract 40
## List of Figures

1.1 Tolerance Analysis versus Tolerance Allocation  
2.2 Typical Cost-vs-Tolerance Relationship  
2.2 Ideal vs. Actual Assembly Distributions  
3.1 Geometric Stack Problem  
4.1 Location of Minimum Cost for a Two Part Assembly  
4.2 Location of Minimum Cost for Three Assembly Functions  
4.3 Creating a Vector Tangent to the Constraint Curve  
4.4 Projection of Negative Production Cost Gradient Vector onto Tangent Plane  
4.5 Creating Search Vector from Sum of Gradient Vectors  
4.6 Steps from Initial Value to Optimum  
4.7 Two Part Assembly with Two Constraints  
5.1 Geometric Stack Problem  
5.2 One-Way Clutch Problem  
5.3 One-Way Clutch Parameters
List of Tables

2.1 Proposed Cost vs. Tolerance Models and Nonlinear Solution Methods 6
3.1 Cost Function and Mean Shift Values for Figure 3.1 16
3.2 Cost and Tolerance Results Before and After Allocation 16
3.3 Cost and Tolerance Results Using Uniform Cost Models 17
5.1 Geometric Stack Problem Cost Function and Mean Shift Values 30
5.2 Geometric Stack Cost and Tolerance Results Before and After Allocation 31
5.3 One-Way Clutch Problem Cost Function and Mean Shift Values 33
5.4 One-Way Clutch Problem Cost and Tolerance Results Before and After Allocation 33
Acknowledgements

First and foremost I must express my great love and appreciation to my wife, Shari. It has been her support and encouragement that has seen me through my studies and made it possible to realize success in my endeavors. She has exercised immeasurable patience, understanding and wisdom while caring for our children, encouraging me and waiting for me as I worked many long hours over the past several years.

I thank my graduate advisor, Dr. Ken Chase, whose example of hard work and perseverance has shown me the key to his success. His door was always open as he willingly spent time to guide, teach and encourage me. He was the core and strength of my graduate experience.

I would also like to thank Brigham Young University and especially the Mechanical Engineering Department for providing an educational environment rich with valuable learning tools and highly qualified faculty. I am very grateful for the opportunity I had to pursue graduate studies.

I must also express my gratitude to ADCATS (the Association for the Development of Computer-Aided Tolerancing Software) for sponsoring my research.
Chapter 1
INTRODUCTION

The specification of tolerances on the dimensions of manufactured parts has a significant impact on final production cost. Tight tolerances can result in excessive process costs, while loose tolerances may lead to increased waste and assembly problems. Improper tolerance specification may also result in inferior product performance and loss of market share. The modest investments required for a thorough tolerance analysis can yield substantial benefits.

Designers use various strategies for assigning tolerances. Tolerance selection is often accomplished by applying "rules of thumb" or extracting tolerance values from similar designs. Depending on how a current assembly differs from previous designs, this may or may not result in appropriate tolerance values. Tolerances may of course be specified by considering the capability of the processes to be employed to manufacture parts. This will assure that the parts can be produced, but it does not say whether they will assemble or perform their function. Tolerance analysis of the entire assembly is required to make this determination. An assembly model must be used to estimate the effects that individual part tolerances will have on the accumulated assembly tolerance.

1.1 Tolerance Stacking

Mechanical assemblies are usually constrained by a specified tolerance on some critical clearance or resultant dimension. Component variations can accumulate or "stack up," affecting resultant assembly dimensions. Two of the most common models used for estimating tolerance accumulation are "Worst Case" analysis and "Root Sum Squares" or "Statistical" analysis (eqs 1.1). The former assumes each part to be as far from the specified dimension as allowed by the tolerance, the total error of the assembly being the linear sum of the tolerance of each part. The latter takes the square root of the sum of the squares of the individual part tolerances as the resulting assembly tolerance. It is based on the sum of the variances of normal distributions. These methods provide some rational basis for assigning tolerances.

Worst case:

\[ T_{asm} = \sum_{j=1}^{n} |T_j| \]

Statistical:

\[ T_{asm} = \left[ \sum_{j=1}^{n} T_j^2 \right]^{1/2} \]

(1.1)
1.2 Tolerance Allocation

Once an assembly tolerance has been analyzed, individual part tolerances within the assembly may be adjusted to better meet the total assembly tolerance specifications. This process of adjusting part tolerances and re-analyzing the assembly is referred to as synthesis or allocation of tolerances.

Figure 1.1 demonstrates the difference between tolerance analysis and tolerance allocation. Analysis is the process of applying an accumulation model (such as eqs 1.1) to a set of part tolerances to determine the resulting assembly tolerance. Allocation is the reverse procedure of assigning individual part tolerances so their sum meets a specified assembly tolerance. This job of allocating tolerances in an effort to meet assembly specifications can be frustrating and may require large amounts of the designer's time.

![Diagram of Tolerance Analysis and Allocation]

Figure 1.1 Tolerance Analysis versus Tolerance Allocation

Theoretically, there exists an infinite number of individual part tolerance combinations which will meet the assembly requirements. Some parts are most likely more difficult to manufacture than others. Some are probably less expensive to produce than others. It makes sense to try to increase the tolerance on the expensive parts and reduce the tolerance on the less expensive parts, while keeping the total assembly tolerance at the
design limit. This leads one to believe that an optimum in terms of cost and manufacturability most likely exists.

1.3 Thesis Objective

The objective of this thesis is to develop an efficient method for determining the necessary individual part tolerance distribution to insure lowest production cost of an assembly. Based on the Estimated Mean Shift model (Section 2.4) as the assembly model, the designer will be given the ability to solve a general problem with the following features:

1) Two-dimensional assemblies described by multiple vector loops and with dependent variables eliminated.

2) Cost functions with different exponent values for each independent dimension in an assembly.

3) Mean shift values, different for each component in an assembly.

1.4 Outline of Thesis

Chapter 2 describes several analytical methods for allocating tolerances for minimum production cost. It also describes an improved tolerance stack model called the Estimated Mean Shift model and a linear algebraic system for analyzing complex 2-D and 3-D assemblies described by vector loops.

Chapter 3 describes an attempt at solving the problem in closed form. This solution method was accurate for a large percentage of problems tested. However, as cost models become more varied within an assembly, the method may give inaccurate tolerance values resulting in a total cost of up to 15% more than the cost of the optimum tolerance allocation.

Chapter 4 describes the development of iterative search techniques intended to move from the estimated solution to the optimum. The final search technique, called the Equalized Gradient Search method, is a significant simplification of the first attempts of this development. The search technique uses the underlying principle of Lagrange multipliers, equating the normalized gradients of the assembly tolerance constraint and the assembly cost function. The Equalized Gradient Search method was able to move from the preliminary closed form estimate to the optimum in an average of five iterations. However,
it was soon discovered that this same search technique could find the optimum from an arbitrary initial estimate in an average of only seven iterations. As two additional iterations of the search are computationally simpler than the closed form estimate, it is recommended that the Equalized Gradient Search be used directly in all cases.

The nonlinear programming package OPTDES.BYU[16] was used as the benchmark for the optimum solution to each problem tested. The Equalized Gradient Search technique developed in Chapter 4 required a comparable number of iterations to find the optimum as the nonlinear programming routines available with OPTDES. The advantage to the new method is its computational simplicity. Actual CPU time required by the new Equalized Gradient method is less than that required by the OPTDES routines.
2.1 Models and Solution Techniques

Investigators have applied several techniques to the problem of tolerance allocation in an attempt to allocate individual part tolerances based on total assembly tolerance requirements. Solving the problem first requires development of an appropriate cost function. Figure 2.1 shows the general cost-versus-tolerance relationship. As a tolerance becomes small, the cost of producing a dimension to that tolerance increases. Larger tolerances are cheaper to produce.

![Tolerance vs Cost Graph](image)

Figure 2.1 Typical Cost-vs-Tolerance Relationship

Cost curves as shown in Figure 2.1 are finite. That is, there exists a limit as to how accurately a given manufacturing process may produce a required dimension. Also, a limit exists where an increase in the tolerance value no longer reduces the cost. A cost function is only valid over a specified range of tolerance values.

Table 2.1 shows the various cost-vs-tolerance functions which have been proposed and the variety of methods used to solve them. The variable T represents the tolerance in each of the models shown, and A represents any fixed costs such as material costs, tooling and setup. A comparison of most of the models and optimization methods found in Table 2.1 has been published[26].
Table 2.1 Proposed Cost vs. Tolerance Models and Nonlinear Solution Methods[4]

<table>
<thead>
<tr>
<th>Type</th>
<th>Cost Model</th>
<th>Method</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear</td>
<td>A - BT</td>
<td>linear prog</td>
<td>Edel and Auer[6]</td>
</tr>
<tr>
<td>reciprocal</td>
<td>A + B/T</td>
<td>Lagrange mult</td>
<td>Chase and Greenwood[3]</td>
</tr>
<tr>
<td>reciprocal squared</td>
<td>A + B/T^2</td>
<td>Lagrange mult</td>
<td>Spotts[23]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nonlinear prog</td>
<td>Parkinson[17]</td>
</tr>
<tr>
<td>reciprocal power</td>
<td>A/B/T^k</td>
<td>Lagrange mult</td>
<td>Sutherland and Roth[24]</td>
</tr>
<tr>
<td></td>
<td>B/T^ki</td>
<td>nonlinear prog</td>
<td>Lee and Woo[8]</td>
</tr>
<tr>
<td>exponential</td>
<td>Be^{-mT}</td>
<td>Lagrange mult</td>
<td>Speckhart[22]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>geometric prog</td>
<td>Wilde and Prentice[25]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>graphical</td>
<td>Peters[19]</td>
</tr>
<tr>
<td>expon/recip power</td>
<td>B(e^{-mT})/T^k</td>
<td>nonlinear prog</td>
<td>Michael and Siddall[10,11]</td>
</tr>
<tr>
<td>piecewise linear</td>
<td>A_i - B_iT_i</td>
<td>linear prog</td>
<td>Bjork[1],Patel[18]</td>
</tr>
<tr>
<td>empirical data</td>
<td>discrete points</td>
<td>zero-one prog</td>
<td>Ostwald and Huang[15]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>combinational</td>
<td>Monte and Datseris[12]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>branch &amp; bound</td>
<td>Lee and Woo[9]</td>
</tr>
</tbody>
</table>

2.2 Lagrange Multiplier Solution of Worst Case and Statistical Models

Several authors listed in Table 2.1 used the technique of Lagrange multipliers. Lagrange multipliers provide a versatile method for tolerance allocation. For this method a cost model and assembly function model must be chosen which have continuous first derivatives. The cost of producing an individual part dimension to a specified tolerance may be represented by:

\[ C_i = A_i + B_i T_i^{-k_i} \]  (2.1)

where \( i \) refers to a particular dimension in the assembly, \( C_i \) is the cost of producing that dimension, and \( A_i \) is a fixed cost (independent of tolerance), \( T_i \) is the dimension tolerance, \( B_i \) and \( k_i \) are the tolerance cost coefficient and exponent, each associated with dimension \( i \). This cost model was chosen because it accommodates a number of the proposed cost models shown in Table 2.1. Attempts are being made to apply this model to typical manufacturing processes. The cost of an assembly is the sum of the individual part costs:
\[ C = \sum_{j=1}^{n} (A_j + B_j T_j^{-k_j}) \]  

(2.2)

Using this model, the general Lagrange multiplier solution is as follows:

minimize: \[ C = \sum_{j=1}^{n} (A_j + B_j T_j^{-k_j}) \]

subject to: Assembly function \((A.F.) = f(T_{asm}, T_1, T_2, \ldots, T_n)\)

where \(A.F.\) may be either of equations 1.1.

The Lagrange multiplier equations are created by combining the derivatives of the cost and assembly limit functions by means of the Lagrange multiplier, \(\lambda\), and setting the sum equal to zero. The equations are then solved for the unique combination of tolerances which will result in the lowest production cost, while satisfying the assembly tolerance constraint.

\[ \frac{\partial (C)}{\partial T_i} + \lambda \frac{\partial (A.F.)}{\partial T_i} = 0 \quad (i = 1, 2, \ldots, n) \]  

(2.3)

\[ -k_i B_i T_i^{-(k_i + 1)} + \lambda \frac{\partial (A.F.)}{\partial T_i} = 0 \]

\[ \lambda = \frac{-k_i B_i T_i^{-(k_i + 1)}}{\frac{\partial (A.F.)}{\partial T_i}} \]

The Lagrange multiplier, \(\lambda\), is the same for all values of \(T_i\). This implies that \(\lambda\) may be eliminated from the equation as follows:

\[ \frac{k_i B_i T_i^{-(k_i + 1)}}{\frac{\partial (A.F.)}{\partial T_1}} = \frac{k_i B_i T_i^{-(k_i + 1)}}{\frac{\partial (A.F.)}{\partial T_i}} \quad (i = 2, 3, \ldots, n) \]  

(2.4)

Equations 2.4 may be solved for each \(T_i\) in terms of \(T_1\). The value of \(T_1\) is then
found by substituting the expressions for $T_1$ into the assembly function (eqs 1.1) and setting it equal to the specified tolerance limit. Thus equations 2.4 and the assembly functions form a system of $n$ equations for the $n$ component tolerances $T_j$. Equations 2.4 assure that the set of component tolerances have the correct ratios to produce the minimum cost and the assembly function is used to size the tolerances to the largest permissible size which meets the assembly limits.

When the cost models specified for each part have the same $k_i$ values, the solution to equations 2.4 reduces to closed form. A simple one-parameter iteration for $T_1$ is required when the $k_i$ values are not equal (i.e. differing cost curves are used for various parts in the assembly). When using the common Worst Case or Statistical models as the Assembly Function, equations 2.4 reduce to equations 2.6 and 2.8:

\[
\text{Worst Case. Assembly Function: } T_{asm} = \sum_{j=1}^{n} T_j s_j \quad (2.5)
\]

\[
T_i = \left( \frac{k_i B_i s_i}{k_i B_i s_i} \right) - \left( \frac{1}{k_i + 1} \right) * \left[ T_1 \right] \quad \left( \frac{k_i + 1}{k_i + 1} \right) \quad (i = 2, 3, \ldots, n) \quad (2.6)
\]

\[
\text{Statistical. Assembly Function: } T_{asm}^2 = \sum_{j=1}^{n} T_j^2 s_j^2 \quad (2.7)
\]

\[
T_i = \left( \frac{k_i B_i s_i^2}{k_i B_i s_i^2} \right) - \left( \frac{1}{k_i + 2} \right) * \left[ T_1 \right] \quad \left( \frac{k_i + 2}{k_i + 2} \right) \quad (i = 2, 3, \ldots, n) \quad (2.8)
\]

The $s_j$ values in each of the assembly models (eqs 2.5 and 2.7) represent the geometric sensitivity of the total assembly tolerance to variations in the nominal dimension of this particular part. By including the sensitivity, this method may be applied to 2-D and 3-D problems in linearized form.

2.3 Extending to 2-D and 3-D Assemblies

Multi-dimensional mechanical assemblies may be modeled mathematically by the
vector loop method[14]. Each vector in a loop represents a controlled part dimension in an assembly. By generating one or more vector loops, all important kinematic relationships of an assembly may be expressed as a system of nonlinear equations. These assembly equations are linearized by using a multivariate Taylor series expansion about the nominal values of the part dimensions, which is then truncated at the first-order terms. The linearized equations are used to model interaction between parts in an assembly due to small dimensional variations.

As was noted in Section 2.2, the \( s_j \) values in equations 2.5 and 2.7 represent the sensitivity of each individual part dimension (independent variables) to any specified assembly constraint (dependent variables). Independent variables are dimensioned quantities which include a specified tolerance due to manufacturing variations. Dependent variables are generally not dimensioned on a drawing, nor do they have a specified tolerance. The dimension and variation of dependent variables result from the dimensions and tolerances of the independent variables and assembly. Sensitivity variables represent this relationship. These sensitivity values may be used to express the assembly relations in matrix form. The Direct Linearization Method (DLM) developed by Marler[14] is an efficient method for deriving this matrix.

Let the system of constraint equations be expressed in the following matrix form:

\[
dH = 0 = [A] \, dX + [B] \, dU \tag{2.9}
\]

where
- \( dH \) = vector of clearance variations
- \( dX \) = vector of variations for independent variables (tolerances)
- \( dU \) = vector of variations for dependent variables
- \( [A] \) = partial derivative matrix for the independent variables
- \( [B] \) = partial derivative matrix for the dependent variables

Solving equation 2.9 for \( dU \) gives:

\[
dU = -[B]^{-1} \, [A] \, dX = [S] \, dX \tag{2.10}
\]

where \([S]\) is the sensitivity matrix.

Once the set of linearized loop equations has been solved for the assembly resultants, expressions for worst case or statistical estimates of assembly variations may be set up using equations 2.5 and 2.7. Tolerance allocation methods may then be employed similar to those methods applied to 1-D assemblies.
2.4 Estimated Mean Shift Model

The allocation methods summarized in Section 2.2 have all used the common Worst Case or Statistical models. Some researchers have noted inadequacies involving these models[2,3,13]. Some of the limitations of these common methods are:

1) The Worst Limit model results in component tolerances which are tight and costly to produce.
   2) Statistical models allow looser tolerances, but often predict higher assembly yields than actually occur.
   3) For assemblies with small numbers of components, or which have one component tolerance which is much greater than the remaining components, the common Statistical model can give component tolerances which are tighter than if computed by a Worst Limit model.
   4) Statistical models assume manufacturing variations follow a Normal or classic bell-shaped distribution, symmetrically positioned at the midpoint of the tolerance limits. They do not take into account possible skewness or bias which are common in manufactured parts. Figure 2.2 illustrates the unexpected rejects which can occur when skewness and bias are not accounted for. The components on the left side of the figure each have a normal and centered distribution. The components on the right side of the figure each have

---

Figure 2.2 Ideal vs. Actual Assembly Distributions
some skewness or shift in the mean which results in a mean shift in the assembly sum also.

Assembly tolerance models have been proposed which attempt to give the designer a greater ability to account for variations such as mean shifts. Chase and Greenwood[3] and Mansoor[13] proposed models in which the total assembly tolerance is the sum of a Worst Limit term and a Root Sum Squares term.

The model proposed by Chase and Greenwood is called the Estimated Mean Shift method because the designer may estimate the bias in the process distribution for each component in an assembly (see equation 2.11). This is done by defining the zone about the midpoint of the tolerance range over which the midpoint of the process is likely to vary. The zone is expressed as a fraction of the tolerance range using a number from 0 to 1. A small number, 0.1 or 0.2 would be chosen if the part production is closely controlled. A higher number 0.7 or 0.8 might be selected for uncertain factors such as a new vendor or a less controlled production process. For common processes the factor should be selected on the basis of prior history from the quality assurance data[4]. The assembly tolerance is calculated using equation 2.11.

\[
T_{asm} = \sum_{j=1}^{n} \left| m_j s_j T_j \right| + \left[ \sum_{j=1}^{n} \left( 1 - m_j \right)^2 s_j^2 T_j^2 \right]^{1/2}
\]  

(2.11)

where \( n \) is the number of components in the assembly, \( m_j \) is the mean shift factor for the \( j^{th} \) component. Note the special case when all the mean shifts are chosen as zero. The resulting assembly tolerance reduces to the simple statistical model. Also, if all mean shifts are chosen equal to 1.0, the method reduces to a worst limit model. The Estimated Mean Shift model covers both standard equations and can therefore simulate the entire continuum between these two extremes. However, the complexity of the Estimated Mean Shift model complicates the allocation problem such that a closed form Lagrange multiplier solution is not possible. The next chapter will describe attempts to extend the Lagrange Multiplier method for tolerance allocation to include linearized 2-D or 3-D problems with mean shifts.
2.5 Comparison of Previous Work to This Thesis

In summary, the intent of this research is to include several factors in tolerance analysis and allocation problems which have not been used by the researchers listed in Table 2.1. These are as follows:

1) Solving 2-D and 3-D problems through the use of geometric sensitivities. The researchers listed have solved only 1-D problems.

2) Solving problems which require multiple vector loops by means of linearization. Previous work has involved only single loop problems.

3) Using the Estimated Mean Shift model with mixed-exponent cost models. The work listed in Table 2.1 used only the standard Statistical and Worst Limit assembly models and non-variable exponents in the cost models.

The next chapter illustrates the attempts to simultaneously include each of these factors in a closed form solution.
Chapter 3
CLOSED FORM ESTIMATE OF LEAST COST SOLUTION

3.1 Lagrange Multiplier Solution of Estimated Mean Shift Model

Section 2.2 described the Lagrange multiplier solutions for the Worst Case and Statistical models. Because of the above noted benefits of the Estimated Mean Shift model, it was desired to apply the Lagrange multiplier technique to this method. However the first derivative of the assembly function shown in equation 3.1 still contains all of the component tolerances, which prevents a closed form solution by separation of variables.

\[
\frac{\partial \text{(A.F.)}}{\partial T_i} = \left| m_i s_i \right| + \frac{(1 - m_j)^2 s_j^2 T_i}{\sum_{j=1}^{n} (1 - m_j)^2 s_j^2 T_j^{1/2}} \quad (i = 1, 2, \ldots, n) \tag{3.1}
\]

The result is a system of \( n \) nonlinear equations which may be solved by nonlinear programming or other techniques. A simpler method for estimating the solution to this problem was sought, guided by the closed form solutions described in Section 2.2.

3.2 Closed Form Estimate of Lagrange Multiplier Solution

It was thought that a closed form Lagrange Multiplier solution using the Estimated Mean Shift model would likely resemble the solutions which use other models. Because the Estimated Mean Shift model is the sum of factors of Worst Limit and Statistical techniques, the expressions for \( T_i \) shown in equations 2.6 and 2.8 are substituted into the appropriate location in the Estimated Mean Shift model, giving:

\[
T_{asm} = \sum_{i=1}^{n} m_i s_i \left[ \frac{k_i B_i s_i}{k_i B_i s_i} \right] \frac{1}{k_i + 1} \cdot \left[ T_1 \right] \frac{k_i + 1}{k_i + 1} + \sum_{i=1}^{n} (1 - m_i)^2 s_i^2 \left[ \frac{k_i B_i s_i^2}{k_i B_i s_i^2} \right] \frac{2}{k_i + 2} \cdot \left[ T_1 \right] \frac{k_i^2 + 2}{k_i^2 + 2} \right]^{1/2} \tag{3.2}
\]
The only unknown in this equation is \( T_1 \). In this case \( T_1 \) represents any one of the individual part tolerances \( T_1, T_2, \ldots, T_n \). This equation is solved \( n \) times, once for each of the part tolerances represented by \( T_1 \). These values are then substituted back into the Estimated Mean Shift model and proportionally scaled to give the specified assembly tolerance limit.

Several problems have been solved using the technique shown in equation 3.2. These results were compared with results obtained using nonlinear programming techniques (Table 3.2). It has been found that results are very consistent with nonlinear programming results for uniform \( k_i \) and \( m_i \) values. However, the procedure becomes less accurate as the \( k_i \) and \( m_i \) values become more varied within the problem. The introduction of an additional \( m \) ratio in the coefficients has been found to improve results considerably:

\[
T_{asm} = \sum_{i=1}^{n} m_i s_i \left[ \frac{k_i B_i s_i m_i}{k_i B_i s_i m_i} \right] \frac{1}{k_i + 1} \times \left[ \frac{T_1}{k_i + 1} \right] + \sum_{i=1}^{n} \frac{(1-m_i)^2 s_i^2}{k_i B_i s_i^2 (1-m_i)^2} \frac{2}{k_i + 2} \times \left[ \frac{T_1}{k_i + 2} \right]^{1/2} \quad (3.3)
\]

Equation 3.3 produced results that in most cases were quite consistent with the nonlinear programming results obtained. Other adjustments in this equation have been attempted to increase the accuracy of this closed form solution. However, nothing was found which consistently improved results for the variety of problems tested.

3.3 OPTDES Comparison

OPTDES.BYU[16] is a nonlinear programming package which employs a variety of nonlinear methods. Figure 3.1 shows a simple three part assembly, typical of the type of problems solved by OPTDES and the Lagrange approximation shown in equation 3.3. Component dimensions \( a, b, c, d, e, \) and \( f \) are the independent variables. The dependent variables are all the \( U \) and \( \theta \) values.
The height of the cylinder axis $U_1$ is the desired assembly resultant or dependent variable. The complete analysis to this 2-D problem was published by Chase, Sorensen and Andersen[5] using the Direct Linearization Method (Section 2.3) with both Worst Case and Statistical models, but no attempt was made to allocate tolerances. A more realistic evaluation of the problem could be made using the Estimated Mean Shift model (eq 2.11).

![Figure 3.1 Geometric Stack Problem](image)

In order to use the Estimated Mean Shift model for this example, some judgements must be made about the manufacturing processes used to produce each individual part. Assume, for example the cylinder will be produced in one's own shop on good turning equipment. It is estimated that it will be quite accurate and therefore assign it a mean shift of 0.25. The "stair-step" base will be produced on a milling machine, also in our own shop, but due to the complex geometry we assign it a value of 0.7 for the mean shift. The rectangular block is an inexpensive, high volume item produced on grinding equipment by an outside vendor. This part is assigned a mean shift value of 0.5.

Table 3.1 shows the cost functions used for each of these processes and the associated mean shift values. The $A_i$, $B_i$ and $k_i$ values correspond to the general cost function (eq 2.1) and were found by curve-fitting data published by Jamieson[7].
Table 3.1 Cost Function and Mean Shift Values for Figure 3.1

<table>
<thead>
<tr>
<th>variable name</th>
<th>nominal value</th>
<th>Ai</th>
<th>Bi</th>
<th>k1</th>
<th>mean shift</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>6.62 mm</td>
<td>3.356</td>
<td>-1.506</td>
<td>-0.4</td>
<td>0.25</td>
</tr>
<tr>
<td>b</td>
<td>6.81</td>
<td>2.347</td>
<td>-0.772</td>
<td>-0.5</td>
<td>0.70</td>
</tr>
<tr>
<td>c</td>
<td>10.68</td>
<td>4.257</td>
<td>-2.573</td>
<td>-0.2</td>
<td>0.70</td>
</tr>
<tr>
<td>d</td>
<td>4.06</td>
<td>2.347</td>
<td>-0.772</td>
<td>-0.5</td>
<td>0.70</td>
</tr>
<tr>
<td>e</td>
<td>24.22</td>
<td>2.347</td>
<td>-0.772</td>
<td>-0.5</td>
<td>0.70</td>
</tr>
<tr>
<td>f</td>
<td>3.91</td>
<td>6.952</td>
<td>-10.77</td>
<td>-1.5</td>
<td>0.50</td>
</tr>
</tbody>
</table>

Table 3.2 displays the results for the geometric stack problem (Fig 3.1). The sensitivity values, $s_i$, are the relation of each of the independent variables (a, b, c, d, e and f) to the dependent variable $U_1$. Using the given tolerance values for each of the independent variables, the tolerance on dimension $U_1$ was found to be 0.449 mm. This value was used as the constrained value for the tolerance on dimension $U_1$ for the Lagrange Approximation and the OPTDES solutions.

Table 3.2 Cost and Tolerance Results Before and After Allocation

<table>
<thead>
<tr>
<th>variable name</th>
<th>nominal value</th>
<th>$U_1$ sensitivity $s_i$</th>
<th>original specs</th>
<th>Lagrange estimate methods</th>
<th>nonlinear program. optimum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>original</td>
<td>Lagrange estimate methods</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>specs</td>
<td>eq 3.2</td>
<td>eq 3.3</td>
</tr>
<tr>
<td>a</td>
<td>6.62 mm</td>
<td>1.3091</td>
<td>0.200 mm</td>
<td>0.099 mm</td>
<td>0.029 mm</td>
</tr>
<tr>
<td>b</td>
<td>6.81</td>
<td>0.2577</td>
<td>0.125</td>
<td>0.089</td>
<td>0.067</td>
</tr>
<tr>
<td>c</td>
<td>10.68</td>
<td>0.0704</td>
<td>0.350</td>
<td>0.504</td>
<td>0.136</td>
</tr>
<tr>
<td>d</td>
<td>4.06</td>
<td>0.7424</td>
<td>0.150</td>
<td>0.554</td>
<td>0.032</td>
</tr>
<tr>
<td>e</td>
<td>24.22</td>
<td>0.2729</td>
<td>0.125</td>
<td>0.036</td>
<td>0.064</td>
</tr>
<tr>
<td>f</td>
<td>3.91</td>
<td>1.0366</td>
<td>0.075</td>
<td>0.760</td>
<td>0.376</td>
</tr>
<tr>
<td>$U_1$</td>
<td>18.72</td>
<td></td>
<td>0.449</td>
<td>0.449</td>
<td>0.449</td>
</tr>
<tr>
<td>Cost of assembly</td>
<td>$17.67</td>
<td></td>
<td>$17.09</td>
<td>$16.50</td>
<td>$16.09</td>
</tr>
</tbody>
</table>

The given tolerance values are those used originally by Chase, Sorensen and Andersen[5] to evaluate the assembly resultant $U_1$ after calculating the sensitivity values. Note that using these given tolerance values with the cost models in Table 3.1 results in a relative cost of $17.67 to produce this assembly. The OPTDES tolerance allocation shows that the assembly tolerance can be maintained at 0.449 mm while decreasing the cost of producing this assembly from $17.67 to $16.09. The Lagrange estimate of this optimum solution reduced the cost by 74% of the OPTDES improvement, down to $16.50. The cost
of the assembly for the Lagrange estimate tolerance distribution is 2.5% higher than the OPTDES solution.

Problems solved using a single cost model applied to all independent variables showed the Lagrange estimate to be much closer to the nonlinear programming optimum. The variation of the cost function exponent values, $k_i$, was determined to be the determining factor influencing the accuracy of the Lagrange estimate. When these exponents are identical for an entire assembly, the Lagrange estimate is consistent with the OPTDES solution, generally within 0.5% of the total assembly cost.

Table 3.3 compares the results for the same problem while using the same cost model for each of the individual parts in an assembly. Note that using cost model 1 gives resulting costs which differ by only 0.3%. Cost model 2 resulted in total costs which differed by only 0.2%. However, a method was required which would compare favorably with the OPTDES solution in all cases.

<table>
<thead>
<tr>
<th>dim</th>
<th>Cost Model 1</th>
<th>Cost Model 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Lagrange eq 3.3</td>
<td>nonlinear optimum</td>
</tr>
<tr>
<td>a</td>
<td>0.25</td>
<td>0.067 mm</td>
</tr>
<tr>
<td>b</td>
<td>0.50</td>
<td>0.379</td>
</tr>
<tr>
<td>c</td>
<td>0.70</td>
<td>0.069</td>
</tr>
<tr>
<td>d</td>
<td>0.70</td>
<td>3.000</td>
</tr>
<tr>
<td>e</td>
<td>0.70</td>
<td>0.346</td>
</tr>
<tr>
<td>f</td>
<td>0.70</td>
<td>0.059</td>
</tr>
<tr>
<td>$U_1$</td>
<td>0.476</td>
<td>0.476</td>
</tr>
<tr>
<td>Cost</td>
<td>$14.30</td>
<td>$14.26</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>k</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost Model 1</td>
<td>3.360</td>
<td>-1.506</td>
</tr>
<tr>
<td>Cost Model 2</td>
<td>2.347</td>
<td>-0.772</td>
</tr>
</tbody>
</table>
4.1 The Optimization Problem

Figure 4.1 provides a graphical representation of the optimization problem. The assembly tolerance constraint represents an infinite number of part tolerance combinations which meet the assembly tolerance specification for this two part assembly. The cost curves depict lines of constant cost of producing the entire assembly. The optimum occurs where the assembly constraint curve reaches the closest point on the lowest cost curve.

![Diagram showing minimum cost location for a two part assembly](image)

**Figure 4.1 Location of Minimum Cost for a Two Part Assembly**

Figure 4.2 shows how the different assembly limit functions discussed in this thesis vary in two-dimensional space. The Worst Limit function (eqs 1.1) is a straight line from one tolerance limit to the other. A change in one tolerance produces an opposite change in the other. The Statistical assembly model (eqs 1.1) is one quarter of an ellipse. The assembly constraint boundary generated by the Estimated Mean Shift model naturally
lies somewhere between the first two according to the mean shift factors which are used.

\[ \frac{T_2}{T_{asm}} \]

\[ \frac{T_1}{T_{asm}} \]

Figure 4.2 Location of Minimum Cost for Three Assembly Functions

The Lagrange multiplier solution to the Worst Limit and Statistical assembly models described in Section 2.2 may also be seen in Figure 4.2. A proportionally accurate solution which is generated is a result which lies somewhere along a straight line from the origin through the point of least cost on the assembly constraint curve. Therefore a proportional change in the individual tolerance values may in one step move the result to the optimum.

4.2 Development of a Search Technique

For a simple two part assembly, the assembly limit equation could be used for a manual iterative search for the minimum. By selecting a value for \( T_1 \), solving the limit equation for \( T_2 \) and calculating the resulting cost, the optimum can be readily found using a hand calculator. But for more than three component tolerances, a computer is required along with a systematic approach.
Using the Lagrange approximation technique developed in Chapter 3 results in an individual part tolerance distribution which is generally close to the optimum. A search technique was sought which would move efficiently from the closed form estimate to the optimum. Nonlinear programming techniques determine the gradients of the constraint and cost functions in order to estimate how to move toward the location of least cost. Figure 4.3 shows a two-part assembly with the necessary gradients. The negative gradient of the production cost function, $-\nabla C$, is perpendicular to the cost curves and is directed towards decreasing cost. The gradient of the assembly tolerance limit, $\nabla L$, is perpendicular to the assembly constraint and is directed towards increasing assembly tolerance.

![Graph showing the relationship between $T_1/T_{asm}$ and $T_2/T_{asm}$ with tangent vectors and cost curves.](image)

Figure 4.3 Creating a Vector Tangent to the Constraint Curve

The search method creates a tangent vector, $V$, which is the projection of the negative cost gradient vector onto the line tangent to the constraint boundary. Figure 4.3 depicts a two part assembly, giving two dimensions. Most problems involve many more
parts, which results in a constraint boundary which is actually a multi-dimensional surface. A tangent line for a two part assembly becomes a tangent plane for a three part assembly and a multi-dimensional hyper-plane for assemblies with more than three parts. The multi-dimensional negative cost gradient vector, \(-\nabla C\), must be projected onto this hyper-plane. This makes it necessary to generate an orthogonal basis to express the hyper-plane.

The search vector, \(V\), is created as follows. The gradients of the cost (eq 4.1) and assembly constraint (eq 2.11) functions are calculated as n-dimensional vectors:

\[
\nabla C = \{ \nabla C_1, \nabla C_2, \nabla C_3, \ldots, \nabla C_n \} \\
\nabla L = \{ \nabla L_1, \nabla L_2, \nabla L_3, \ldots, \nabla L_n \} 
\]

(4.1)

where \(n\) represents the number of independent dimensions in the assembly and

\[
\begin{align*}
\nabla C_i &= \frac{\partial C}{\partial T_i} = k_i B_i T_i^{k_i - 1} \\
\nabla L_i &= \frac{\partial (A.F.)}{\partial T_i} = \left| m_i s_i \right| + \frac{(1 - m_i)^2 s_i^2 T_i}{\sum_{j=1}^{n} (1 - m_j)^2 s_j^2 T_j^2}^{1/2}
\end{align*}
\]

(4.2)

The orthogonal basis is then created with \(n-1\) vectors as shown in equations 4.3.

\[
\begin{align*}
R_1 &= \left\{ \frac{\nabla L_1}{\nabla L_1^2} , - \frac{1}{\nabla L_2} , 0, 0, \ldots, 0 \right\} \\
R_2 &= \left\{ \frac{\nabla L_1}{\nabla L_1^2 + \nabla L_2^2} , - \frac{1}{\nabla L_3} , 0, 0, \ldots, 0 \right\} \\
R_3 &= \left\{ \frac{\nabla L_1}{\nabla L_1^2 + \nabla L_2^2 + \nabla L_3^2} , - \frac{1}{\nabla L_4} , 0, 0, \ldots, 0 \right\} \\
& \quad \vdots \\
R_{n-1} &= \left\{ \ldots , - \frac{1}{\nabla L_n} \right\}
\end{align*}
\]

(4.3)

The \(n\)-dimensional basis vectors, \(R_1, R_2, \ldots, R_{n-1}\) are orthogonal (see equation...
4.4) and span the hyperplane. It is easily shown that for equation 4.3:
\[ \mathbf{R}_i \cdot \mathbf{R}_k = 0 \quad i \neq k \]  
(4.4)

\( \mathbf{R}_1 \) and \( \mathbf{R}_2 \) are shown to be orthogonal in the two-dimensional tangent plane in Figure 4.4. \( \nabla \mathbf{L} \) is orthogonal to both of the basis vectors. \( -\nabla \mathbf{C} \) is projected onto the tangent plane to create the search vector, \( \mathbf{V} \).

![Diagram of projection](image)

**Figure 4.4 Projection of Negative Production Cost Gradient Vector onto Tangent Plane**

The cost gradient vector is projected onto the tangent hyper-plane to create the tangent search vector, \( \mathbf{V} \), as shown in equation 4.5. Note that both the numerator and denominator in the brackets of equation 4.5 are dot products and this ratio is the projection of \( \nabla \mathbf{C} \) onto each basis vector \( \mathbf{R}_j \). Both equations 4.3 and 4.5 lend themselves easily to computer calculations. Simple to code, they are nonetheless computationally intensive.

\[ \mathbf{V} = \sum_{j=1}^{n-1} \left[ \frac{\nabla \mathbf{C} \cdot \mathbf{R}_j}{\mathbf{R}_i \cdot \mathbf{R}_j} \right] \mathbf{R}_j \]  
(4.5)

The tangent search vector, \( \mathbf{V} \), is used to adjust the current tolerance values. The new estimate of the part tolerance vector, \( \mathbf{T}^k \), is the sum of the old vector, \( \mathbf{T}^{k-1} \), and some
factor, $\beta$, of the search vector $V$. See equation 4.6.

$$T^k = T^{k-1} + \beta V$$  \hspace{1cm} (4.6)

After this adjustment, the tolerance values generally no longer sum to the assembly constraint value and must be scaled back to the constraint. This may be accomplished by adding a small factor, $\gamma$, of the assembly constraint gradient vector to the adjusted values. In equation 4.7, a value for $\gamma$ must be found through a one-dimensional iteration for which $T^k$ sums to the assembly constraint value using the desired assembly function.

$$T^k = T^{k-1} + \beta V + \gamma V_L$$  \hspace{1cm} (4.7)

After an updated tolerance vector, $T$, is found, the gradients, orthogonal basis and search vector must again be calculated. This method had some difficulty in moving quickly to the optimum. Many more iterations are required with this method than OPTDES.

4.3 Simplified Technique

As noted in Section 2.2, the Lagrange multiplier method makes use of the fact that the optimum solution occurs at the location where the cost gradient and assembly function gradient are colinear. This means that if these two gradients vectors are normalized (unit length), each of the individual components of the cost gradient vector will be equal in magnitude to each of the corresponding components of the assembly tolerance vector at the optimum. The Lagrange multiplier equation 2.3 may be rewritten in gradient form:

$$\nabla C + \lambda \nabla L = 0$$  \hspace{1cm} (4.8)

The Lagrange multiplier, $\lambda$, may be eliminated from the equation if the gradient vectors are normalized, adjusting them to have the same magnitude. The double bars in equation 4.9 represent unit vectors.

$$||\nabla C|| + ||\nabla L|| = 0$$  \hspace{1cm} (4.9)

In examining the search vector, $V$, generated using the projection onto the tangent hyper-plane (Section 4.1) at a certain point on the assembly constraint and the two gradient vectors at this same point, it became evident that the search vector was not moving efficiently to the location of colinear gradients. That is, the relative magnitudes of the search vector components did not represent relative differences between corresponding
individual components of the gradient vectors. It became obvious that simply summing these two normalized gradient vectors gave a more useful search vector to follow. Figure 4.5 shows how these two normalized gradient vectors are summed to give the appropriate search direction. Note that Figure 4.5 uses a negative $\nabla L$ where Figure 4.3 shows a positive $\nabla L$.

![Graph showing the creation of a search vector from the sum of gradient vectors.](image)

**Figure 4.5 Creating Search Vector from Sum of Gradient Vectors**

Note that the search vector, $\mathbf{v}$, stays on the inside of the assembly constraint curve instead of moving tangent and away from the curve. Especially for initial steps, which are generally large, the search vector can stay closer to the assembly constraint than a tangent vector does. This indicates that smaller corrections would be required to move the design back onto the constraint for large steps.

It may be noted that summing two normalized gradients in this manner gives a
resulting vector with magnitude ranging from 0 to 2. This magnitude, then, may be used as a measure of how close the design is to the optimum. A search vector of magnitude greater than one indicates two gradient vectors which are not very close to being colinear. A search vector with magnitude close to zero shows that the gradient vectors are quite close to being colinear. This magnitude may be used to determine an appropriate step size for adjusting the individual tolerances toward the optimum.

Using this method, the optimum was found in an average of five iterations after using the initial estimate described in Chapter 3. However, as Figure 4.5 shows how the search vector keeps the design close to the assembly constraint even for large steps, it was noted that an initial estimate may not be advantageous. In fact, using this method while starting from arbitrary locations changed the average number of iterations required from five to just seven. It should be noted that two iterations of this method are much simpler to calculate than the estimation procedure developed in Chapter 3.

The Equalized Gradient Search technique does not require an initial set of tolerance values which meet the assembly constraint. The method will calculate the assembly tolerance for an arbitrary set of part tolerances and proportionally scale the part tolerances onto the assembly constraint. The explanation accompanying Figure 2.1 indicates that only a specified range of tolerance values are valid for each cost function. In the event that an initial tolerance value is outside of these process limits, an average of these values is used as the initial tolerance for that dimension. Figure 4.6 shows the required search steps which are:

1) Calculating assembly tolerance value given initial individual tolerance values

2) Scaling onto the constraint boundary (depicted by vector A)

3) Calculating and summing the negative unit cost and limit functions gradients, -\|\nabla C\| and -\|\nabla L\|, to form the first search direction vector, \( \mathbf{V}_1 \)

4) Moving back onto the assembly constraint boundary in the direction of \( \nabla L \) calculated in step 3

5) Repeating steps 3 and 4 for a series of vectors \( \mathbf{V}_2, \mathbf{V}_3, \) etc. as required to achieve the optimum

Once at the assembly constraint boundary, the step towards the optimum is taken in the direction of \( \mathbf{V}_1 \) at a length which keeps the value of the assembly function within the
feasible region. The feasible region is any place underneath the assembly constraint curve. All assembly values in this region are less than or equal to the required assembly value and are therefore "acceptable" from the designers standpoint.

Note that each of the vectors $V_1$, $V_2$, and $V_3$ shown in Figure 4.6 approach the optimum from the same direction. This is done for the sake of simplicity. In actuality, the step sizes used generally cause the vectors to move past the optimum several times during the search process, but this process is difficult to show clearly in a figure.

![Figure 4.6 Steps From Initial Value to Optimum](image-url)
4.3.1 Step Size Improvement

Various methods were investigated for determining the best step size for the search vector. That is, what magnitude of $V$ would move the design closest to the optimum cost. Two ideas were similar in nature in that they attempted to use information from previous designs to approximate the curvature of the assembly constraint. The first method simply curve fit three points on the assembly constraint. The second method used two points and their gradients to do the same thing. However, in most cases, after two or three iterations of the problems tested, the search method had already moved to within five percent of the optimum cost. The calculation of approximate curvature is not considered advantageous as the design only requires refining by this time.

4.3.2 Comparison to SQP and GRG

All solutions to problems discussed and tested in this thesis have been compared to results obtained using OPTDES.BYU, a commercially available and widely used nonlinear programming software package. The principal algorithms used by OPTDES are Sequential Quadratic Programming (SQP) and Generalized Reduced Gradient (GRG). These were determined to be the best of dozens of different algorithms tested on about 100 different nonlinear problems studied in 1980 by Schittkowski[21]. For the tolerance allocation problems tested for this thesis SQP and GRG, on the average, each required a similar or greater number of global iterations as did the Equalized Gradient solution method described in Section 4.3 to solve the same problems.

4.4 Possible Extension to Multiple Constraints

The solution techniques described to this point have involved only single constraints. That is, the dimension of only one dependent variable has been specified, while the independent variables have been adjusted to meet this one requirement. The ability to handle multiple constraints on a tolerance allocation problem is a very desirable option. This would give the designer the ability to specify two or more geometric requirements on dependent variables in an assembly.

The Equalized Gradient technique currently sums the gradient of the cost function with the gradient of the single assembly constraint to form a search vector. When more
than one constraint exists, the search vector could be formed using only the closest constraint, or it may be necessary to perform this sum for each of the assembly constraints and then develop a preferred search vector. The correction would then need to move the design back to the constraint boundary which is the most binding (satisfying all other constraints).

Figure 4.7 depicts a two-part assembly with two assembly tolerance constraints. Each constraint curve actually represents a limiting boundary as to the acceptable tolerance value for the specified dependent dimension. Any result inside both of the curves would be an acceptable solution in terms of tolerance. However, the optimization technique always results in a solution which is on a constraint boundary, because the lowest production cost will occur at a maximum acceptable tolerance value. The shaded region shows the area where both assembly constraints are satisfied. The optimum occurs at the location where the two constraints reach the lowest point on the process cost curves.

Figure 4.7 Two Part Assembly with Two Constraints
Note that the assembly constraint curves intersect the axes at the reciprocal of the geometric sensitivity of the part tolerance to the assembly tolerance. Sensitivity $s_{i,j}$ represents the geometric sensitivity of part $i$ to assembly constraint $j$. The axes of Figures 4.1 and 4.2 are labeled with 0.5 and 1.0, which are the correct reduced values when the sensitivities are 1.0. This condition occurs when the tolerances of the two parts vary in the identical direction in space.

With a single constraint, the individual tolerance values are scaled to move the design back onto the constraint boundary. In the case of multiple constraint boundaries, each of the constraint boundaries will need to be evaluated when scaling in order to determine which is most binding at the new location. Considering the simplicity of the current method, this technique should not be difficult to develop.

It should be noted that when working with the cost functions, it is not important which curve fitting equation is used. The relationships are fit over a relatively small region. The importance lies in the curvature of the equation; the cost must increase with decreasing tolerance and vice versa. For the equation used in this thesis (eq 2.1), the required curvature may be obtained by ensuring that the coefficient and the exponent of the tolerance are of opposite sign. Equation 2.1 is written with a positive coefficient and a negative exponent to give this idea.

It should also be noted that additional constraints may include not only specified dimensions within an assembly, but also form and feature tolerances (ANSI-Y14.5). However, this requires that cost-tolerance functions be available.
Chapter 5
SAMPLE PROBLEMS

5.1 Geometric Stack Problem

The geometric stack problem shown in Figure 5.1 was presented in Chapter 3 as an example of how tolerance analysis is performed. Here the same starting values are used for tolerance allocation. Table 5.1 shows the basic geometry, cost functions and mean shift values of the problem. Table 5.2 gives the comparative results of the allocation procedures.

![Geometric Stack Problem Diagram]

Figure 5.1 Geometric Stack Problem

<table>
<thead>
<tr>
<th>variable name</th>
<th>nominal value</th>
<th>cost parameters</th>
<th>mean shift</th>
<th>process limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>6.62 mm</td>
<td>3.356, -1.506, -0.4</td>
<td>0.25</td>
<td>0.02 mm - 1.0 mm</td>
</tr>
<tr>
<td>b</td>
<td>6.81</td>
<td>2.347, -0.772, -0.5</td>
<td>0.70</td>
<td>0.03 - 1.0</td>
</tr>
<tr>
<td>c</td>
<td>10.68</td>
<td>4.257, -2.573, -0.2</td>
<td>0.70</td>
<td>0.01 - 1.0</td>
</tr>
<tr>
<td>d</td>
<td>4.06</td>
<td>2.347, -0.772, -0.5</td>
<td>0.70</td>
<td>0.01 - 1.0</td>
</tr>
<tr>
<td>e</td>
<td>24.22</td>
<td>2.347, -0.772, -0.5</td>
<td>0.70</td>
<td>0.01 - 1.0</td>
</tr>
<tr>
<td>f</td>
<td>3.91</td>
<td>6.952, -10.77, -1.5</td>
<td>0.50</td>
<td>0.01 - 1.0</td>
</tr>
</tbody>
</table>
Table 5.2 Geometric Stack Cost and Tolerance Results Before and After Allocation

<table>
<thead>
<tr>
<th>variable name</th>
<th>nominal value (mm)</th>
<th>nominal sensitivity</th>
<th>tolerance specs</th>
<th>Equalized Gradient optimum</th>
<th>nonlinear program. optimum</th>
<th>normalized cost sensitivity values</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>6.62</td>
<td>1.3091</td>
<td>0.200 mm</td>
<td>0.026 mm</td>
<td>0.044 mm</td>
<td>0.47857</td>
</tr>
<tr>
<td>b</td>
<td>6.81</td>
<td>0.2577</td>
<td>0.125</td>
<td>0.065</td>
<td>0.050</td>
<td>0.12598</td>
</tr>
<tr>
<td>c</td>
<td>10.68</td>
<td>0.0704</td>
<td>0.350</td>
<td>0.956</td>
<td>0.539</td>
<td>0.04443</td>
</tr>
<tr>
<td>d</td>
<td>4.06</td>
<td>0.7424</td>
<td>0.150</td>
<td>0.010*</td>
<td>0.010*</td>
<td>0.30019</td>
</tr>
<tr>
<td>e</td>
<td>24.22</td>
<td>0.2729</td>
<td>0.125</td>
<td>0.045</td>
<td>0.038</td>
<td>0.15110</td>
</tr>
<tr>
<td>f</td>
<td>3.91</td>
<td>1.0366</td>
<td>0.075</td>
<td>0.353</td>
<td>0.369</td>
<td>0.80011</td>
</tr>
<tr>
<td>U₁</td>
<td>18.72</td>
<td></td>
<td>0.449</td>
<td>0.449</td>
<td>0.449</td>
<td></td>
</tr>
</tbody>
</table>

|                   |               |                      | Iterations | Cost       | CPU time   |
|                   |               |                      | 5          | $ 17.67    | 0.10 sec   |
|                   |               |                      | 7          | $ 16.02    | 0.93 sec   |

* Process limit

It may be noted that results for some of the individual tolerances differ by a large percentage. For example, part c has allocated values of 0.956 and 0.539 using the two methods. This is due to the low magnitude of the cost sensitivity (last column) for variable c. The tolerance of c can be increased dramatically without affecting the assembly tolerance much, but reducing the cost. Dimension f has the largest cost sensitivity value and therefore changes in this tolerance have a large influence on the total cost. Note that the resulting allocated tolerance for part f differs by only 4% using the two methods. Using the cost sensitivity values, a designer may do some refining and rounding especially on those parts with low cost sensitivity at the optimum. The two resulting total costs, $16.02 and $16.09, differ by less than one half percent. Each may be considered adequately "optimum." The tolerance values calculated and shown here are likely to be more accurate than they can actually be manufactured. Also note the significant difference in CPU time required by the two methods. Some of this difference may be attributed to the fact that OPTDES must use a series of small perturbations in order to determine derivatives, whereas the derivatives are hard-coded into the Equalized Gradient technique.

An additional advantage to the Equalized Gradient Method is the fact that the designer may be provided with the calculated gradient vectors. These vectors show which tolerances have the greatest influence on the cost of the assembly and which tolerances have the greatest influence on the resulting assembly tolerance. With this information the
designer may then be able to make well advised refinements of individual part tolerance values.

5.2 One-Way Clutch Problem

Figure 5.2 shows a cross-section of a one-way clutch assembly. This cross-section is symmetrical about two axes and the problem may therefore be modeled using just one fourth of the cross-section as shown in Figure 5.3. The angle $\theta_1$ is critical to proper operation of the clutch.

![Figure 5.2 One-Way Clutch Problem](image)

The least cost distribution of part tolerances was found for this problem using the tolerance on this angle as the required assembly tolerance. Table 5.3 shows the basic geometry, cost functions and mean shift values of the problem. Table 5.4 gives the comparative results of the allocation procedures.
Table 5.3 One-Way Clutch Problem Cost Function and Mean Shift Values

<table>
<thead>
<tr>
<th>variable name</th>
<th>nominal value</th>
<th>cost parameters</th>
<th>mean shift</th>
<th>process limits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$A_i$ $B_i$ $k_i$ $m_i$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>27.65 mm</td>
<td>8.855 -3.195 -0.33 0.70</td>
<td>0.02 mm 1.0 mm</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>50.80</td>
<td>-2.211 5.796 0.51 0.50</td>
<td>0.03 1.0</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>11.43</td>
<td>7.424 -20.64 -1.9 0.70</td>
<td>0.008 1.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.4 One-Way Clutch Problem Cost and Tolerance Results Before and After Allocation

<table>
<thead>
<tr>
<th>variable name</th>
<th>nominal value</th>
<th>sensitivity</th>
<th>$\theta_1$</th>
<th>tolerance values</th>
<th>normalized cost sensitivity values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>original specs Equalized Gradient program. optimum</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>27.65 mm</td>
<td>0.2084</td>
<td>0.100 mm</td>
<td>0.100 mm 0.099 mm</td>
<td>0.00888</td>
</tr>
<tr>
<td>b</td>
<td>50.80</td>
<td>0.2068</td>
<td>0.025</td>
<td>0.030* 0.030*</td>
<td>0.99996</td>
</tr>
<tr>
<td>c</td>
<td>11.43</td>
<td>0.4153</td>
<td>0.010</td>
<td>0.010 0.010</td>
<td>0.00096</td>
</tr>
</tbody>
</table>

$\theta_1$ 7.0°

| Iterations | 3 | 5 |
| Cost       | $49.94 | $31.14 | $31.31 |
| CPU time   | 0.04 sec | 0.31 |

* Process limit
Note that the results for the Equalized Gradient method and the nonlinear programming method are again comparable, differing in total cost by only half of a percent. Again there is a large difference in the amount of CPU time required to achieve the optimum.

It should also be noted that process limits affected the solutions to both the geometric stack and one-way clutch problems. This indicates that other possibilities should be investigated. Cost-tolerance curves which represent the same process with different tolerance values and cost curves for different processes over the same tolerance ranges (when another process may be used) should be used to determine an overall least cost assembly. One should strongly suspect that alternatives would advantageous in the one-way clutch problem due to the cost sensitivity values. A value of 0.99996 for part b indicates that the assembly cost in influenced almost entirely by this part at the optimum.
Chapter 6
CONCLUSIONS AND RECOMMENDATIONS

6.1 Contributions of this Thesis

This thesis presents a general search technique for finding least cost component tolerances. The Equalized Gradient Search method gives the designer the following capabilities for assembly modeling:

1) 2-D or 3-D assemblies.

2) Multiple loops with kinematic constraints (based on the vector loop method of geometric representation).

3) Mean shift factors for each individual dimension within an assembly.

4) Mixed cost functions based on process selection for each individual dimension in an assembly.

In the past, tolerance allocation methods have only been applied to simple one-dimensional assemblies with perfectly symmetric distributions. Tolerance allocation methods may now be extended to more complex, multiple-dimension mechanical assemblies. The designer has the ability to apply manufacturing data to each individual part in an assembly, when the data is available. Lack of information about one part no longer penalizes the entire analysis and subsequent tolerance allocation procedure on an entire assembly. The designer can choose appropriate mean shift vectors to account for process uncertainties.

Through automating the tolerance allocation process, the designer of manufactured assemblies will be able to explore many more designs quickly. Instead of spending time making adjustments to find one design which meets specifications, the designer will instantly be provided with the least cost design which meets requirements. Time may be spent in valuable pursuits such as investigating the least cost tolerance distributions using a variety of alternative manufacturing processes for applicable parts in an assembly.

The Equalized Gradient method offers an efficient alternative to the standard available nonlinear programming algorithms. It not only compares favorably to standard methods in terms of global iterations required, but through its simplicity, it also requires
less actual CPU time. This technique may easily be implemented in any computer code, requiring only about 250 lines of code.

6.2 Future Work

6.2.1 Multiple Constraints

The problems solved in Chapter 5 of this thesis have involved only single constraints. Section 4.3 described the constraints and considerations involved in applying this technique to multiple constraints, including form and feature tolerances. The ability to handle multiple constraints on a tolerance allocation problem is a very desirable option for the designer. This should likely be the next step of development for the research presented in this thesis.

6.2.2 Three-Dimensional Problems

The number of dimensions has little affect on the use of the Equalized Gradient Search method developed in this thesis. All of the research herein is based on the accurate development of a sensitivity matrix which relates all dependent variables in an assembly to all independent variables. Marler's[14] work dealt with generation of this matrix for two-dimensional assemblies and was used as the basis for this thesis. Robison[20] developed another procedure which generates a sensitivity matrix for three-dimensional assemblies. This method may be all that is required to extend the results of this thesis to three-dimensional problems.

6.2.3 Other Equalized Gradient Applications

The Equalized Gradient method developed in this thesis may have application as a general nonlinear programming technique. Some alterations may be required for cases when the gradients of the two functions are oriented the same instead of opposite as in the tolerance allocation problem.

Chase, Greenwood, Loosli, and Haugland[4] have developed a method of automated process selection. This allows the designer to search among various processes
for a particular part. Combining this method with the Equalized Gradient technique would give the designer added convenience.

6.2.4 Development of Valid Cost-Tolerance Models

Table 2.1 provides a summary of many of the cost-tolerance models which have been used by researchers. Though many models are available, not much data has been published detailing cost-tolerance relationships. The functions used in this thesis are the results of curve fitting data from the only source which was found. Much data is needed for describing cost-tolerance relationships for both standard tolerances as well as form and feature tolerances.
References


