ABSTRACT

The essence of this research is the reconciliation and standardization of feature fitting algorithms used in Coordinate Measuring Machine (CMM) software and the development of Inspection Maps (i-Maps) for representing geometric tolerances in the inspection stage based on these standardized algorithms. The i-Map is a hypothetical point-space that represents the substitute feature evaluated for an actual part in the inspection stage.

The first step in this research is to investigate the algorithms used for evaluating substitute features in current CMM software. For this, a survey of feature fitting algorithms available in the literature was performed and then a case study was done to reverse engineer the feature fitting algorithms used in commercial CMM software. The experiments proved that algorithms based on least squares technique are mostly used for GD&T inspection and this wrong choice of fitting algorithm results in errors and deficiency in the inspection process. Based on the results, a standardization of fitting algorithms is proposed in light of the definition provided in the ASME Y14.5 standard and an interpretation of manual inspection practices. Standardized algorithms for evaluating substitute features from CMM data, consistent with the ASME Y14.5 standard and manual inspection practices for each tolerance type applicable to planar features are developed.

Second, these standardized algorithms developed for substitute feature fitting are then used to develop i-Maps for size, orientation and flatness tolerances
that apply to their respective feature types. Third, a methodology for Statistical Process Control (SPC) using the I-Maps is proposed by direct fitting of i-Maps into the parent T-Maps. Different methods of computing i-Maps, namely, finding mean, computing the convex hull and principal component analysis are explored. The control limits for the process are derived from inspection samples and a framework for statistical control of the process is developed. This also includes computation of basic SPC and process capability metrics.
Dedicated to my father Mani, mother Saraswathy and sister Saradha whose love and support made me what I am and let me pursue my dreams. Special mention to my roommates Bharani, Vasu, Raghu, MNA, Sridhar and friends back home Chandru, SRB, Barbi, Mureli, Radhu, Ramprasad and Pattabi on whom I can always count for support.
ACKNOWLEDGEMENTS

My most sincere appreciation and gratitude go to my committee chair and advisor, Dr. Jami Shah who has been a very patient and inspiring mentor and guide in my professional life. Without his contribution and support this work would not have been possible. I would like to give a special mention to Dr. Davison, who was always willing to spend time with me, answer my questions and helped me all along.

I would like to thank Dr. Rajan for sharing his expertise on optimization algorithms and Dr. Farin for his time serving as part of my committee. I would like to mention the valuable assistance offered by Zoltan Farkas for CMM measurements.

I would also like to thank students in the Design Automation Laboratory – Naresh Kumar, Manikandan Mohan, Adam Dixon, Patrick Clasen, Bradley Gaul, Nadeem Khan, Srinath Balaji – for their assistance. They were great colleagues to have and were always ready to share their thoughts, think about and help me with my ideas.

I would also like to thank former students Amir Mujezinović, Gaurav Ameta, Saurabh Bhide, Zhengshu Shen who did preliminary work on the mathematical model with geometric tolerances.

I wish to acknowledge the financial support for this work provided by the National Science Foundation Grant #CMMI-0700878.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td>xvii</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xix</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Geometric Dimensioning and Tolerancing</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Dimensional Metrology</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Problem Statement</td>
<td>6</td>
</tr>
<tr>
<td>2 LITERATURE REVIEW ON TOLERANCE MODELING AND TOLERANCE ANALYSIS</td>
<td>8</td>
</tr>
<tr>
<td>2.1 Mathematical Modeling of Tolerances</td>
<td>8</td>
</tr>
<tr>
<td>2.1.1 Parametric Models</td>
<td>9</td>
</tr>
<tr>
<td>2.1.2 Offset Zone Models</td>
<td>9</td>
</tr>
<tr>
<td>2.1.3 Variational Surfaces Models</td>
<td>10</td>
</tr>
<tr>
<td>2.1.4 Kinematic Models</td>
<td>10</td>
</tr>
<tr>
<td>2.1.5 Degree of Freedom Models</td>
<td>11</td>
</tr>
<tr>
<td>2.1.6 Other Models</td>
<td>11</td>
</tr>
<tr>
<td>2.2 Requirements of a Mathematical Model</td>
<td>12</td>
</tr>
<tr>
<td>2.3 Tolerance Maps</td>
<td>13</td>
</tr>
<tr>
<td>2.4 Tolerance Analysis</td>
<td>13</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>2.4.1 Tolerance Charts..............................................................16</td>
<td></td>
</tr>
<tr>
<td>2.4.2 Parametric Tolerance Analysis.............................................19</td>
<td></td>
</tr>
<tr>
<td>2.4.2.1 Linearized Tolerance Analysis..........................19</td>
<td></td>
</tr>
<tr>
<td>2.4.2.2 Non-Linear Tolerance Analysis.......................20</td>
<td></td>
</tr>
<tr>
<td>2.4.3 Parametric Approach Using CAD or Abstracted Geometry.........................21</td>
<td></td>
</tr>
<tr>
<td>2.4.3.1 Direct Constraint Model in CAD.................22</td>
<td></td>
</tr>
<tr>
<td>2.4.3.2 Abstracted Feature Parameter Model.............22</td>
<td></td>
</tr>
<tr>
<td>2.4.4 Vector Loop or Kinematic Methods..........................24</td>
<td></td>
</tr>
<tr>
<td>2.4.5 Variation Zone Based Analysis..........................25</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>TOLERANCE MODELING AND ANALYSIS WITH TOLERANCE MAPS..............................27</td>
</tr>
<tr>
<td>3.1</td>
<td>Tolerance Modeling with T-Maps.............................................27</td>
</tr>
<tr>
<td>3.2</td>
<td>Areal Coordinates..............................................................28</td>
</tr>
<tr>
<td>3.3</td>
<td>Tolerance Maps......................................................................29</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Tolerance Map for a Planar Round Face...............30</td>
</tr>
<tr>
<td>3.3.1.1</td>
<td>Size Tolerance on Planar Round Face........30</td>
</tr>
<tr>
<td>3.3.1.2</td>
<td>Size and Orientation on Round Face........35</td>
</tr>
<tr>
<td>3.3.1.3</td>
<td>Size and Form on Planar Round Face........36</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Tolerance Map for a Planar Rectangular Face........38</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Tolerance Map for an Axis.............................................40</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>3.4</td>
<td>T-Map Based Tolerance Analysis</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Tolerance Accumulation</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Worst Case and Statistical Analysis</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Worst Case and Statistical Analysis</td>
</tr>
<tr>
<td>4</td>
<td>TOLERANCE MAP CATALOG</td>
</tr>
<tr>
<td>4.1</td>
<td>T-Map Catalog</td>
</tr>
<tr>
<td>4.2</td>
<td>T-Maps for Planar Circular Features</td>
</tr>
<tr>
<td>4.3</td>
<td>T-Maps for Planar Rectangular Features</td>
</tr>
<tr>
<td>4.4</td>
<td>T-Maps for Planar Triangular Features</td>
</tr>
<tr>
<td>4.5</td>
<td>T-Maps for FOS Features</td>
</tr>
<tr>
<td>4.6</td>
<td>T-Maps for Axis Elements</td>
</tr>
<tr>
<td>4.7</td>
<td>T-Maps for Mid-Plane Elements</td>
</tr>
<tr>
<td>4.8</td>
<td>T-Maps for Circular and Cylindrical Features</td>
</tr>
<tr>
<td>5</td>
<td>ANALYSIS OF METROLOGY PRACTICES</td>
</tr>
<tr>
<td>5.1</td>
<td>Dimensional Metrology</td>
</tr>
<tr>
<td>5.2</td>
<td>Definition of Tolerance Classes in the ASME</td>
</tr>
<tr>
<td></td>
<td>Size Tolerance</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Orientation Tolerance</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Parallelism Tolerance</td>
</tr>
<tr>
<td>5.2.2.1</td>
<td>Perpendicularity Tolerance</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>6.4.1.2 Planar Datums Subject to Size Variations</td>
<td>136</td>
</tr>
<tr>
<td>6.4.1.3 Non-Planar Datums Subject to Size Variations</td>
<td>138</td>
</tr>
<tr>
<td>6.4.2 Fitting Datum Features</td>
<td>138</td>
</tr>
<tr>
<td>6.4.2.1 Deficiencies in Current Fitting Methods</td>
<td>139</td>
</tr>
<tr>
<td>6.4.2.2 Wobbling or Unstable Datum Features</td>
<td>139</td>
</tr>
<tr>
<td>6.4.3 Fitting Algorithms for Individual Tolerance Classes</td>
<td>140</td>
</tr>
<tr>
<td>6.4.3.1 Size Tolerances</td>
<td>141</td>
</tr>
<tr>
<td>6.4.3.2 Orientation Tolerances</td>
<td>141</td>
</tr>
<tr>
<td>6.4.3.3 Form Tolerances</td>
<td>142</td>
</tr>
<tr>
<td>6.4.3.4 Position Tolerances</td>
<td>142</td>
</tr>
<tr>
<td>6.4.3.5 Runout Tolerances</td>
<td>143</td>
</tr>
<tr>
<td>6.5 GIDEP and NIST Alert on Errors in CMM Inspection Algorithms</td>
<td>145</td>
</tr>
<tr>
<td>6.6 Case Study on Commercial CMM Inspection Software</td>
<td>146</td>
</tr>
<tr>
<td>6.6.1 Size Tolerance</td>
<td>148</td>
</tr>
<tr>
<td>6.6.2 Perpendicularity Tolerance</td>
<td>149</td>
</tr>
<tr>
<td>6.6.3 Parallelism Tolerance</td>
<td>151</td>
</tr>
<tr>
<td>6.6.4 Flatness Tolerance</td>
<td>152</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>6.7</td>
<td>155</td>
</tr>
<tr>
<td>Reconciliation of Fitting Algorithms w.r.t GD&amp;T standards And Manual Inspection Practices</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>158</td>
</tr>
<tr>
<td>NORMATIVE GUIDELINES FOR FEATURE FITTING ALGORITHMS IN CMM INSPECTION</td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>158</td>
</tr>
<tr>
<td>Standardization of Fitting Algorithms</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>158</td>
</tr>
<tr>
<td>Size Tolerance</td>
<td></td>
</tr>
<tr>
<td>7.2.1</td>
<td>158</td>
</tr>
<tr>
<td>Standard Definition</td>
<td></td>
</tr>
<tr>
<td>7.2.2</td>
<td>159</td>
</tr>
<tr>
<td>Interpretation of Manual Inspection Practices</td>
<td></td>
</tr>
<tr>
<td>7.2.3</td>
<td>159</td>
</tr>
<tr>
<td>Fitting Algorithm for Size Tolerance on a Planar Face Coupled with Orientation</td>
<td></td>
</tr>
<tr>
<td>7.2.4</td>
<td>160</td>
</tr>
<tr>
<td>Fitting Algorithm for Size Tolerance on a FOS Feature without Orientation</td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td>161</td>
</tr>
<tr>
<td>Parallelism Tolerance</td>
<td></td>
</tr>
<tr>
<td>7.3.1</td>
<td>161</td>
</tr>
<tr>
<td>Standard Definition</td>
<td></td>
</tr>
<tr>
<td>7.3.2</td>
<td>161</td>
</tr>
<tr>
<td>Interpretation of Manual Inspection Practices</td>
<td></td>
</tr>
<tr>
<td>7.3.3</td>
<td>162</td>
</tr>
<tr>
<td>Fitting Algorithm for Parallelism Tolerance on Planar Feature</td>
<td></td>
</tr>
<tr>
<td>7.3.4</td>
<td>164</td>
</tr>
<tr>
<td>Fitting Algorithm for Parallelism Tolerance on FOS Feature</td>
<td></td>
</tr>
<tr>
<td>7.4</td>
<td>166</td>
</tr>
<tr>
<td>Perpendicularity Tolerance</td>
<td></td>
</tr>
<tr>
<td>7.4.1</td>
<td>166</td>
</tr>
<tr>
<td>Standard Definition</td>
<td></td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>7.4.2</td>
<td>Interpretation of Manual Inspection Practices.....166</td>
</tr>
<tr>
<td>7.4.3</td>
<td>Fitting Algorithm for Angularity Tolerance on</td>
</tr>
<tr>
<td></td>
<td>Planar Feature......................................................167</td>
</tr>
<tr>
<td>7.4.4</td>
<td>Fitting Algorithm for Angularity Tolerance on</td>
</tr>
<tr>
<td></td>
<td>FOS Feature.........................................................168</td>
</tr>
<tr>
<td>7.5</td>
<td>Angularity Tolerance.......................................................171</td>
</tr>
<tr>
<td>7.5.1</td>
<td>Standard Definition..............................................171</td>
</tr>
<tr>
<td>7.5.2</td>
<td>Interpretation of Manual Inspection Practices....172</td>
</tr>
<tr>
<td>7.5.3</td>
<td>Fitting Algorithm for Perpendicularity Tolerance on</td>
</tr>
<tr>
<td></td>
<td>Planar Feature......................................................172</td>
</tr>
<tr>
<td>7.5.4</td>
<td>Fitting Algorithm for Perpendicularity Tolerance on</td>
</tr>
<tr>
<td></td>
<td>FOS Feature.............................................................173</td>
</tr>
<tr>
<td>7.6</td>
<td>Flatness Tolerance...........................................................175</td>
</tr>
<tr>
<td>7.6.1</td>
<td>Standard Definition..............................................175</td>
</tr>
<tr>
<td>7.6.2</td>
<td>Interpretation of Manual Inspection Practices....175</td>
</tr>
<tr>
<td>7.6.3</td>
<td>Fitting Algorithm for Flatness Tolerance............176</td>
</tr>
<tr>
<td>7.7</td>
<td>Position Tolerance...........................................................177</td>
</tr>
<tr>
<td>7.7.1</td>
<td>Standard Definition..............................................177</td>
</tr>
<tr>
<td>7.7.2</td>
<td>Interpretation of Manual Inspection Practices....177</td>
</tr>
<tr>
<td>7.7.3</td>
<td>Fitting Algorithm for Position Tolerance</td>
</tr>
<tr>
<td></td>
<td>Based on Center Plane of the Slot.........................177</td>
</tr>
</tbody>
</table>

xiii
### CHAPTER 7

#### 7.7.4 Fitting Algorithm for Position Tolerance

Based on Side Surfaces of the Slot

---

### CHAPTER 8

#### 8.1 Tolerances in Inspection Stage

---

#### 8.2 Inspection Map

---

#### 8.3 Inspection Map for Size Tolerance on

- **Planar Circular Face**

---

#### 8.4 Inspection Map for Size Tolerance on

- **Planar Rectangular Face**

---

#### 8.5 Orientation Tolerance Independent of Size

- **8.5.1 T-Map for Orientation Tolerance applied to**
  - **Planar Face Independent of Size**

- **8.5.2 I-Map for Orientation Tolerance applied to**
  - **Planar Face Independent of Size**

---

#### 8.6 I-Maps for Flatness Tolerance

- **8.6.1 I-Map for Flatness Tolerance applied to**
  - **Planar Face Coupled with Size**

- **8.6.2 I-Map for Flatness Tolerance applied to**
  - **Planar Face Independent of Size**

---

### CHAPTER 9

#### 9. STATISTICAL QUALITY CONTROL USING INSPECTION MAPS
<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>Quality in Manufacturing</td>
</tr>
<tr>
<td>9.2</td>
<td>Statistical Quality Control</td>
</tr>
<tr>
<td>9.3</td>
<td>Control Charts</td>
</tr>
<tr>
<td>9.3.1</td>
<td>Shewhart Control Charts</td>
</tr>
<tr>
<td>9.3.1.1</td>
<td>$\bar{x}$ and R Chart</td>
</tr>
<tr>
<td>9.3.1.2</td>
<td>$\bar{x}$ and S Chart</td>
</tr>
<tr>
<td>9.4</td>
<td>Statistical Process Control Using I-Maps</td>
</tr>
<tr>
<td>9.4.1</td>
<td>SPC Using I-Maps for Size Tolerance on Planar Circular Face</td>
</tr>
<tr>
<td>9.4.1.1</td>
<td>Construction of T-Map</td>
</tr>
<tr>
<td>9.4.1.2</td>
<td>Construction of Control Limits</td>
</tr>
<tr>
<td>9.4.1.3</td>
<td>Fitting of I-Maps within the T-Map and Control Limits</td>
</tr>
<tr>
<td>9.4.1.4</td>
<td>Process Capability Index</td>
</tr>
<tr>
<td>9.4.2</td>
<td>SPC Using I-Maps for Size Tolerance on Planar Rectangular Face</td>
</tr>
<tr>
<td>9.4.3</td>
<td>SPC Control limits by offsetting T-Map</td>
</tr>
<tr>
<td>9.4.4</td>
<td>SPC Using I-Maps for Orientation Tolerance on Planar Circular and Rectangular Faces</td>
</tr>
<tr>
<td>9.4.5</td>
<td>Percentage Non-Conformance</td>
</tr>
<tr>
<td>9.5</td>
<td>Principal Component Analysis with i-Maps</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>Page</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>9.5.1 Computing Principal Components</td>
<td>232</td>
</tr>
<tr>
<td>9.5.2 PCA Analysis</td>
<td>234</td>
</tr>
<tr>
<td>9.6  i-Maps from convex hulls</td>
<td>235</td>
</tr>
<tr>
<td>10 CONCLUSION</td>
<td>237</td>
</tr>
<tr>
<td>10.1 Future Work</td>
<td>238</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>240</td>
</tr>
</tbody>
</table>

APPENDIX

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>MATLAB IMPLEMENTATION</td>
</tr>
<tr>
<td>B</td>
<td>T-MAP LIBRARY</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>sample tolerance chart ................................................................. 18</td>
</tr>
<tr>
<td>4.1</td>
<td>T-Maps for planar circular features ....................................................... 50</td>
</tr>
<tr>
<td>4.2</td>
<td>T-Maps for planar rectangular features ..................................................... 53</td>
</tr>
<tr>
<td>4.3</td>
<td>T-Maps for planar triangular features ........................................................ 60</td>
</tr>
<tr>
<td>4.4</td>
<td>T-Maps for FOS features ........................................................................ 61</td>
</tr>
<tr>
<td>4.5</td>
<td>T-Maps for axis elements ........................................................................ 62</td>
</tr>
<tr>
<td>4.6</td>
<td>T-Maps for mid-plane elements ................................................................ 69</td>
</tr>
<tr>
<td>4.7</td>
<td>T-Maps for circular and cylindrical features ............................................. 71</td>
</tr>
<tr>
<td>5.1</td>
<td>Comparison of conventional and coordinate metrology using CMM.............. 118</td>
</tr>
<tr>
<td>6.1</td>
<td>Table summarizing the Chebyshev one-sided and two-sided fits .......... 135</td>
</tr>
<tr>
<td>6.2</td>
<td>Flatness tolerance results of case study ................................................ 154</td>
</tr>
<tr>
<td>6.3</td>
<td>Perpendicularity tolerance results of case study .................................... 154</td>
</tr>
<tr>
<td>6.4</td>
<td>Parallelism tolerance results of case study ............................................ 154</td>
</tr>
<tr>
<td>9.1</td>
<td>Values of constant $A_2$ in the R chart .................................................... 214</td>
</tr>
<tr>
<td>9.2</td>
<td>Values of constants $D_3$ and $D_4$ in the R chart ................................... 215</td>
</tr>
<tr>
<td>Table</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>9.3. Data for Principal Component Analysis</td>
<td>232</td>
</tr>
<tr>
<td>9.4. Mean subtracted data for Principal Component Analysis</td>
<td>232</td>
</tr>
<tr>
<td>9.5. Covariance Matrix for the mean subtracted data</td>
<td>232</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Schematic representation of tolerance classes and subclasses</td>
</tr>
<tr>
<td>2.1</td>
<td>Different configurations for maximum and minimum stackup conditions in an assembly</td>
</tr>
<tr>
<td>2.2</td>
<td>Non-linear tolerance analysis via Monte-Carlo simulation</td>
</tr>
<tr>
<td>2.3(a)</td>
<td>Abstracted feature parameter method of tolerance analysis in a simple assembly – A simple two part assembly with GD&amp;T</td>
</tr>
<tr>
<td>2.3(b)</td>
<td>Abstracted feature parameter method of tolerance analysis in a simple assembly – Model creation and feature variation</td>
</tr>
<tr>
<td>2.3(c)</td>
<td>Abstracted feature method of tolerance analysis in a simple assembly – Assembly constraint satisfaction and measurement computation</td>
</tr>
<tr>
<td>3.1</td>
<td>Areal coordinates in a triangle</td>
</tr>
<tr>
<td>3.2(a)</td>
<td>A diametral cross-section of a round bar with size tolerance ‘t’</td>
</tr>
<tr>
<td>3.2(b)</td>
<td>A point map that represents half the planes in one diametral cross-section of the tolerance zone of Fig. 3.2(a)</td>
</tr>
<tr>
<td>3.3</td>
<td>The T-Map for size tolerance on a planar circular face</td>
</tr>
<tr>
<td>3.4</td>
<td>A diametral half section of the T-Map in Fig. 3.3 and its two diconical subsets t” and a vertical line of dimension (t – t”) representing the truncation due to the orientational tolerance t</td>
</tr>
<tr>
<td>3.5</td>
<td>A solid model of the T-Map for size tolerance and orientation tolerance applied together on a planar circular face</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>3.6</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>A diametral half section of the T-Map in Fig. 3.3 and its two diconical subsets of dimension $t'$ and $(t - t')$.</td>
</tr>
<tr>
<td>3.7</td>
<td>38</td>
</tr>
<tr>
<td></td>
<td>The trade-off between the internal subsets for form and range of positions within the T-Map.</td>
</tr>
<tr>
<td>3.8</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>The end face of the rectangular bar with size tolerance showing the exaggerated tolerance zone and the different basis planes that define the T-Map for the feature.</td>
</tr>
<tr>
<td>3.9</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>The T-Map for the rectangular bar for a tolerance zone as defined in Fig. 3.8.</td>
</tr>
<tr>
<td>3.10</td>
<td>41</td>
</tr>
<tr>
<td></td>
<td>A cylindrical tolerance-zone for an axis bounded by circles $\overline{C}$ and $C$ with diameters equal to the position tolerance $t$ and $\ldots$ are basis lines.</td>
</tr>
<tr>
<td>3.11</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>Two dimensional cross sections of the 4-D T-Map for the tolerance zone defined by Fig. 3.10.</td>
</tr>
<tr>
<td>3.12</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>An assembly of two cylindrical bars with size tolerance applies on the two end faces. The basis planes that define the T-Maps for the individual bars are also shown.</td>
</tr>
<tr>
<td>3.13</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>Solid model of the accumulation T-Map for the assembly shown in Fig. 3.12.</td>
</tr>
<tr>
<td>3.14</td>
<td>47</td>
</tr>
<tr>
<td></td>
<td>Steps to perform worst case and statistical tolerance analysis.</td>
</tr>
<tr>
<td>5.1</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>Size Tolerance on a feature of Size.</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5.2</td>
<td>Parallelism tolerance on a planar surface</td>
</tr>
<tr>
<td>5.3</td>
<td>Perpendicularity tolerance on a planar surface</td>
</tr>
<tr>
<td>5.4</td>
<td>Angularity tolerance on a planar surface</td>
</tr>
<tr>
<td>5.5</td>
<td>Position tolerance on two holes</td>
</tr>
<tr>
<td>5.6</td>
<td>Explanation for position tolerance on a slot w.r.t center plane and side surfaces</td>
</tr>
<tr>
<td>5.7</td>
<td>Explanation for position tolerance on a hole</td>
</tr>
<tr>
<td>5.8</td>
<td>Form tolerance applied on a planar surface</td>
</tr>
<tr>
<td>5.9</td>
<td>Straightness tolerance applied on a cylindrical bar</td>
</tr>
<tr>
<td>5.10</td>
<td>Datum Reference Frame (DRF) defined by three datum planes</td>
</tr>
<tr>
<td>5.11</td>
<td>Primary datum of a part made to contact the surface plate</td>
</tr>
<tr>
<td>5.12</td>
<td>Using hole blocks for simulating datums</td>
</tr>
<tr>
<td>5.13</td>
<td>Using L-blocks for simulating datums</td>
</tr>
<tr>
<td>5.14</td>
<td>Using pin gauges to simulate axis datums</td>
</tr>
<tr>
<td>5.15</td>
<td>Using pin gauges and V-blocks to simulate axis datums</td>
</tr>
<tr>
<td>5.16</td>
<td>Measuring size tolerances using screw gauge</td>
</tr>
<tr>
<td>5.17</td>
<td>Measurement of the width of a slot using parallels</td>
</tr>
<tr>
<td>5.18</td>
<td>Measuring size using gage blocks and size indicators</td>
</tr>
<tr>
<td>5.19</td>
<td>Measurement of parallelism tolerance using surface plates and dial indicator</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5.20</td>
<td>Dial indicator being used to measure parallelism on a planar surface</td>
</tr>
<tr>
<td>5.21</td>
<td>Measurement of parallelism tolerance on FOS feature using dial indicator</td>
</tr>
<tr>
<td>5.22</td>
<td>Fixing DRF for perpendicularity tolerance using surface plates and dial indicator</td>
</tr>
<tr>
<td>5.23</td>
<td>Measuring perpendicularity tolerance on a planar surface using surface plates, simulated datums and dial indicator</td>
</tr>
<tr>
<td>5.24</td>
<td>Measuring perpendicularity tolerance on a planar surface using pin gauges as simulated datums and dial indicator</td>
</tr>
<tr>
<td>5.25</td>
<td>Measuring perpendicularity tolerance on a slot using dial indicator</td>
</tr>
<tr>
<td>5.26</td>
<td>Measuring perpendicularity tolerance on a slot using dial indicator and pin gauges as simulated datums</td>
</tr>
<tr>
<td>5.27</td>
<td>Measurement of angularity tolerance on a planar surface using sine plate and dial indicator</td>
</tr>
<tr>
<td>5.28</td>
<td>Measurement of position tolerance of a slot by measuring the size of the slot using screw gauges</td>
</tr>
<tr>
<td>5.29</td>
<td>Measurement of position tolerance of a slot using special mounting fixtures with holding chucks, height gauge and dial indicators</td>
</tr>
<tr>
<td>5.30</td>
<td>Measurement of flatness tolerance on a planar surface using inclinometers and dial indicators</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>5.31</td>
<td>Measurement of flatness tolerance on a planar FOS using dial indicators</td>
</tr>
<tr>
<td>5.32</td>
<td>Measurement of flatness tolerance directly using dial indicator</td>
</tr>
<tr>
<td>5.33</td>
<td>Measurement of the accuracy of trammels using dial indicators</td>
</tr>
<tr>
<td>5.34</td>
<td>Measurement of flatness using trammels and dial indicators</td>
</tr>
<tr>
<td>5.35</td>
<td>Measurement of flatness tolerance on an angular planar face using gage blocks, surface plane and dial indicator</td>
</tr>
<tr>
<td>5.36</td>
<td>Measurement of flatness tolerance on a rectangular FOS using dial indicators</td>
</tr>
<tr>
<td>6.1</td>
<td>Different objective functions result in different substitute features for the same set of data points</td>
</tr>
<tr>
<td>6.2</td>
<td>Schematic of fitting substitute features in CMM software</td>
</tr>
<tr>
<td>6.3</td>
<td>Fitting for a slot used as a datum</td>
</tr>
<tr>
<td>6.4</td>
<td>Chebyshev Two-Sided fit for Form Tolerance</td>
</tr>
<tr>
<td>6.5</td>
<td>Maximum Inscribed (MI) Circle Fitting</td>
</tr>
<tr>
<td>6.6</td>
<td>Maximum Circumscribed (MC) Circle Fitting</td>
</tr>
<tr>
<td>6.7</td>
<td>Isometric views of the part used in the case study</td>
</tr>
<tr>
<td>7.1</td>
<td>Fitting for size tolerance on a prismatic FOS feature</td>
</tr>
<tr>
<td>7.2</td>
<td>Fitting for parallelism tolerance on a planar feature</td>
</tr>
<tr>
<td>7.3</td>
<td>Fitting for parallelism tolerance on a FOS feature</td>
</tr>
<tr>
<td>7.4</td>
<td>Fitting for perpendicularity tolerance on a planar feature</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>7.5</td>
<td>Fitting for perpendicularity tolerance on a FOS feature</td>
</tr>
<tr>
<td>7.6</td>
<td>Fitting for angularity tolerance on a planar feature</td>
</tr>
<tr>
<td>7.7</td>
<td>Fitting for angularity tolerance on a FOS feature</td>
</tr>
<tr>
<td>7.8</td>
<td>Fitting for flatness tolerance on a planar face</td>
</tr>
<tr>
<td>7.9</td>
<td>Fitting for position tolerance based on the center plane of a slot</td>
</tr>
<tr>
<td>7.10</td>
<td>Fitting for position tolerance based on side surfaces of a slot</td>
</tr>
<tr>
<td>8.1</td>
<td>Tolerance Zone for Size Tolerance on Planar Circular Face</td>
</tr>
<tr>
<td>8.2</td>
<td>CMM data points measured on a planar circular face</td>
</tr>
<tr>
<td>8.3</td>
<td>Cross section of the tolerance zone of a planar circular face with the CMM data points and the basis planes of the T-Map</td>
</tr>
<tr>
<td>8.4</td>
<td>Substitute feature fit to the CMM data points – Case 1</td>
</tr>
<tr>
<td>8.5</td>
<td>I-Map for size tolerance on planar circular face inside parent T-Map – Case 1</td>
</tr>
<tr>
<td>8.6</td>
<td>Substitute feature fit to the CMM data points – Case 2</td>
</tr>
<tr>
<td>8.7</td>
<td>I-Map for size tolerance on planar circular face inside parent T-Map – Case 2</td>
</tr>
<tr>
<td>8.8</td>
<td>Tolerance Zone for Size Tolerance on Planar Rectangular Face with measured CMM data points</td>
</tr>
<tr>
<td>8.9</td>
<td>Tolerance Zone for Size Tolerance on Planar Rectangular Face with measured CMM data points and substitute feature – Case 1</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>8.10</td>
<td>I-Map for size tolerance on a planar rectangular face inside parent T-Map – Case 2</td>
</tr>
<tr>
<td>8.11</td>
<td>Tolerance Zone for Size Tolerance on Planar Rectangular Face with measured CMM data points and substitute feature – Case 2</td>
</tr>
<tr>
<td>8.12</td>
<td>I-Map for size tolerance on a planar rectangular face inside parent T-Map – Case 2</td>
</tr>
<tr>
<td>8.13</td>
<td>Tolerance zone for parallelism tolerance applied on a planar surface</td>
</tr>
<tr>
<td>8.14</td>
<td>Tolerance zone for angularity tolerance applied on a planar surface</td>
</tr>
<tr>
<td>8.15</td>
<td>Tolerance zone for parallelism tolerance applied on a planar surface exaggerated to show the basis planes and the coordinate system</td>
</tr>
<tr>
<td>8.16</td>
<td>T-Map for orientation tolerance applied on a planar face</td>
</tr>
<tr>
<td>8.17</td>
<td>I-Map for orientation tolerance applied on a planar face overlaid on the parent T-Map</td>
</tr>
<tr>
<td>8.18</td>
<td>I-Maps for flatness tolerance applied on a planar circular face overlaid on the parent T-Maps for three different cases</td>
</tr>
<tr>
<td>8.19</td>
<td>I-Map for flatness tolerance applied on a planar surface overlaid on the parent T-Map</td>
</tr>
<tr>
<td>9.1</td>
<td>Fish Bone Diagram</td>
</tr>
<tr>
<td>9.2</td>
<td>Sample control charts</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>9.3</td>
<td>I-Maps within the control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for size tolerance on a planar circular face – Sample 1</td>
</tr>
<tr>
<td>9.4</td>
<td>I-Maps within the control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for size tolerance on a planar circular face – Sample 2</td>
</tr>
<tr>
<td>9.5</td>
<td>I-Maps within the overall control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for size tolerance on a planar circular face</td>
</tr>
<tr>
<td>9.6</td>
<td>I-Maps within the overall control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for size tolerance on a planar rectangular face</td>
</tr>
<tr>
<td>9.7</td>
<td>SPC control limits defined by offsetting the parent T-Map</td>
</tr>
<tr>
<td>9.8</td>
<td>Overall control limits defined by $\mu$ and $\sigma$ of the samples and the tolerance limits defined by the T-Map for orientation tolerance on a planar rectangular face</td>
</tr>
<tr>
<td>9.9</td>
<td>Eigen vectors of the covariance matrix and mean subtracted data plotted together</td>
</tr>
</tbody>
</table>
1.1 Background:

The Industrial revolution at the turn of the last century gave birth to mass production which substantially reduced cost of production, increased revenue for companies and brought technological advancements. The highly competitive industrial world mandates businesses to explore every possible method to reduce cost, eliminate waste, improve efficiency and increase profits. At the same time, one cannot compromise on the quality of the products produced and quality becomes the most important attribute that can make or break a company’s fortune. A successful and efficient mass manufacturing process should yield products that have very little variability. This results in parts that are interchangeable, thus increasing assemblability.

Although manufacturing technology has advanced greatly, it is still not possible to produce perfect and ideal parts. Since there is always a finite error associated with manufacturing, it is necessary to define the amount of acceptable error. Tolerances can be defined as the acceptable limits of dimensional variation that are allowed on a part. The use of tolerances facilitates mass production and also governs the quality and variability of parts produced by a manufacturing process. Tolerances play a major role in determining the production cost because they dictate the selection of machinery, tools, and fixtures for achieving the desired accuracy in manufacturing. Thus, tolerances that are usually fixed at the
design stage have a cascading effect throughout the production cycle. It becomes imperative that the designer understands the importance of tolerances and communicates the dimensional requirements of the part effectively for mass production to be successful. The methods for representing tolerances and tolerance measurement methods have evolved over time to be consistent with each other. For any representation to be successful, it has to be conveyed in a manner such that it can be understood by design, manufacturing and inspection personnel. For this reason, standards are established to provide a consistency in how tolerances are applied. [19]

1.2 Geometric Dimensioning and Tolerancing:

There are two techniques for representing tolerances – conventional tolerancing and geometric tolerancing. In conventional tolerancing, the tolerances are specified as plus minus values and are associated with a dimension corresponding to some feature. However, the conventional methods of tolerancing have some inherent deficiencies. The variation of a dimension is controlled only in the direction in which it is applied. The absence of datum reference frames in conventional tolerancing results in incorrect measures of variations. Also, finer variations such as orientation, form and profile could not be applied directly to features using conventional methods of tolerancing. As a result, rejection and rework rates were quite high with conventional tolerancing methods. This necessitated the development of Geometric Dimensioning and Tolerancing methods, commonly abbreviated as GD&T.
The principles of GD&T and their interpretation are described in the ASME Y14.5 1994 and 2009 standards [1, 2]. The standard provides guidelines for designers to represent the tolerances in a way that best describes the intended functionality of the product. The symbols described in the standard acts as a clear and consistent medium of communication between the designers, machinists, industrial engineers, tool designers and inspectors. GD&T represents each tolerance as a boundary called the tolerance zone within which all the variations of the tolerated feature should lie. The shape of the tolerance zone depends on the class of tolerance and the geometry of the feature being tolerated. GD&T symbols provide designers with a standardized tool to specify tolerances that reflect the intended functionality of the part and to communicate the functional relationship between different features of a part. In addition to communicating the functional intent, the datums along with tolerance zones also determine how a particular feature must be inspected. The standard specifies six classes of tolerances and each of these classes, except for the size, have one or more subclasses. Figure 1.1 gives a schematic description of the tolerance classes defined in the ASME Y14.5 1994 standard.

The ASME standard allows designers to specify the exact limits to variations in shape and position of a feature, achieving greater control than with conventional tolerancing. Adoption of GD&T in the past few decades has allowed designers greater control on dimensions and greatly improved part acceptance rate and increased process efficiency. In a nutshell GD&T has resulted in increased
performance and functionality of products without making compromises on producability. However, the standard has evolved more as an art than science and comes across as a set of rules that are derived from decades of practical experience. This gives rise to some ambiguities in its interpretation and hence a need for developing a mathematical model that represents the tolerances at design, manufacturing and inspection stages and that conforms to the principles laid down by the standard arises.

1.3 Dimensional Metrology:

The objective of dimensional metrology is two pronged. First, to check manufactured parts for tolerance conformance and second, to evaluate process capability. A phenomenal increase in quality consciousness has resulted in the past couple of decades seeing a great leap in the hardware and software capabilities used in metrology. Coordinate Measuring Machines (CMM) are being used to automate the process of inspection and this has tremendously increased inspection capabilities. These machines can inspect several parts in the time needed to inspect one part using conventional inspection techniques. This has also improved the confidence levels in predicting quality at the inspection stage. This increase in the quality is because of the capability of CMMs to use larger sample sets more frequently. One thing that is common to all these advances in the inspection process is that measurements made on a part are essentially communicated in terms of a cloud of points.
Fig 1.1 Schematic Representation of tolerance classes and subclasses [1, 2]
The responsibility of hardware, whether it costs $1000 or $10 million ends with measuring a set of points and few numbers associated with them. The software that is associated with the CMM is usually what does the bulk of the ‘intelligent’ work that can be used for making inferences. We do not get the features of interest directly such as planes, lines, circles, cylinders etc., but only a cloud of points from the CMM machine.

1.4 Problem Statement:

Feature fitting algorithms are coded in data analysis software which take the inspection data points as input and output the parameters that define the substitute geometry. The challenge in creating these fitting algorithms is how to precisely and accurately approximate the actual geometry of manufactured features. A variety of feature fitting algorithms based on Least Squares and Chebyshev algorithms are available. We now have these three important entities – the ASME Y14.5 standard, the traditional methods of measurement and the algorithms used in CMMs.

It is very important that these three entities should be consistent with each other. We know that the first two agree with each other since they evolved together. But the third entity, the CMM computational algorithms, developed from a different branch in mathematics found an application in metrology. There is a need for standardizing these computational algorithms in order to produce a unique substitute feature in conformance with the ASME standard. Finally, there is also a need to represent these substitute features that represent actual
manufactured parts using a mathematical model. Such a mathematical model for representation of tolerances in the inspection stage would pave way for a tool that can perform comprehensive dimensional inspection and statistical process evaluation.
CHAPTER 2

LITERATURE REVIEW ON TOLERANCE MODELING AND TOLERANCE ANALYSIS

2.1 Mathematical Modeling of Tolerances:

Conventional methods of tolerance specifications permit only one degree of freedom for the target entity. They do not provide ways to represent form variations on parts. This is one of the main reasons for the development of GD&T [1, 2] as an alternative to the conventional methods. In GD&T, allowed variations are defined within the tolerance zone and the target entity has several degrees of freedom within the tolerance zone. The location of the tolerance zone is determined by the basic or the nominal dimension. The size and other characteristics of the zone are determined by the tolerance values of size, form and orientation. The GD&T standard also allows interaction between different classes of tolerances. This permits modeling of stack ups accurately between target entities.

Several attempts have been made to develop a mathematical model for representing tolerances. These past attempts can generally be grouped into five categories. These are parametric models, offset zone models, variational surfaces, kinematic models and Degree of Freedom (DOF) models. [8]. A brief description of the various tolerances models and their deficiencies are mentioned below.
2.1.1 Parametric Models:

Parametric modeling of tolerances is derived from the principles of parametric CAD. In parametric cad, the shape and size of an entity is represented by a set of explicit dimensions and constraints. Solving a set of simultaneous equations from these dimensions and constraints gives the value of dependant dimensions. Similarly, tolerances are represented as plus minus variations of parameters [73, 74, 75]. The parametric equations can be used for point-to-point tolerance analysis but a zone based tolerance analysis is not possible. Also, they have limited applications suited only for 2-D profiles and polyhedral 3-D objects. They also cannot represent finer tolerances such as form and Datum Reference Frames (DRF) have no meaning in this type of representation and they cannot handle directed datum-target relations.

2.1.2 Offset Zone Models:

In the offset zone models, the tolerance zone is modeled as a Boolean of bodies having the maximum and minimum values derived by offsetting the object based on the applied tolerances. This results in a representation of the tolerance zone in terms of the boundary surfaces of the part [71, 72]. The problem with this approach is that it cannot be applied to represent variations individually. Neither does it model the interactions within the tolerance zone. The dependence of this model on the boundary surfaces of the part makes it incompatible with tolerances that apply to mid-planes or axis elements. This model also does not consider datum precedence of the DRF.
2.1.3 Variational Surfaces Models:

Variational surfaces model approach calculates the coefficients of the surfaces by changing the values of the variables that define the original model. The values of these variables are computed based to the tolerance values. Each surface is calculated independently of others. Surface variations can handle form tolerances using higher degree surfaces or surface triangulation. However, the model is not compatible with the ASME Standard and has topological problems. For example, it is not possible to maintain tangency and incidence conditions among vertices, edges and surfaces of a solid. Also, variational surfaces cannot model tolerances applied to derived features such as axes or mid-planes.

2.1.4 Kinematic Models:

The kinematic model of representing tolerances is derived from the transformational matrix approach for tolerance stack-up analysis using mechanisms. In this model, each tolerance class is represented as a combination of kinematic joints. Bodies are modeled in terms of joints and links. The relationship between a target and its datum features are represented by kinematic links. These combinations are used to estimate geometric variations caused by the applied tolerances. The tolerance analysis is based on vector additions and first order partial derivatives of analyzed dimensions [70]. Although this model can represent all the tolerance classes specified in the standard, the complete GD&T related information cannot be stored. For example, this model cannot enforce the Rule 1 of the standard [1]. Also, there are inherent disadvantages such as the
datum reference frames cannot be validated and the tolerance analysis is not zone based in the Kinematic models.

2.1.5 Degree of Freedom (DOF) Models:

DOF models are a collection of models developed by different set of people individually. These models use spatial degrees of freedom to represent geometric tolerances. These models treat geometric entities as rigid bodies with degrees of freedom and geometric relations between the entities are treated as constraints on these degrees of freedom. [76, 77, 78, 79] These along with the related model known as the Technologically and Topologically Related Surfaces (TTRS) have the capability to validate DRFs and tolerance types. Also developed was the concept of minimum geometric datum elements (MGDE). This model is a part of the minimum system of datum reference frames required for each type of geometric tolerance. However, these models suffer from deficiencies such as the inability to incorporate form tolerances, non-conformance to Rule 1 of the standard [1], absence of datum precedence and material modifier effects.

2.1.6 Other Models:

Giordano et al. [68] created a model based on a set of inequalities which are used to represent tolerance-zones. The inequalities are mapped to a geometric deviation space. This is a similar concept to the Tolerance-Map, but it is limited by the number of independent parameters needed to represent a line or plane. An assembly is modeled as clearances in joints between parts or deviations between features. The model effectively handles the interaction of tolerances in an
assembly with the Minkowski sum of deviation spaces. They have also modeled projected tolerance-zones and material conditions. Apart from these models, the metrology community has also been active in an attempt to “mathematize” the standards. However, these attempts to come up with a comprehensive mathematical model have not addressed the complete range of tolerances, their intricacies and the applications possible with these tolerances.

2.2 Requirements of a Mathematical Model:

A comprehensive mathematical representation is important not just for coming up with an unambiguous model for tolerances. It is also the basis from the perspective of computer modeling of tolerances and automation of tolerance analysis. Unless the rules, classification and the symbols defined in the ASME standard are defined mathematically, it is impossible to do a faithful computer modeling of tolerances. [11] Wu et. al describes the characteristics needed for a computer model for GD&T. They are completeness, compatibility, computability and validity. Completeness implies that the model should represent all the tolerance classes mentioned in the ASME standard including DRFs and material modifier conditions. Compatibility means the model should conform to the principles laid down by the standards and be consistent with engineering practices. Computability means that the model should be understandable to the computers and enable the automation of GD&T reasoning and tolerance analysis with minimum human interaction. Validity is the ability to detect incorrect GD&T specifications and to correct them. So for automation of tolerance modeling and
analysis, each tolerance class should be interpreted in terms of the controlled entity type, the nature of geometric and dimensional variations, applicable datum reference frames and its corresponding metric relations.

2.3 Tolerance Maps:

Based on the above principles, Davidson and Shah [3] introduced a new mathematical model for geometric tolerances consistent with the ASME standard that differentiates between all classes of tolerances, follows the datum precedence of the DRFs and can represent the stack up of tolerances in an assembly. Central to the model proposed by Davidson and Shah is the Tolerance-Map (T-Map). The T-Map is a hypothetical Euclidean volume of points, the shape, size and internal subsets of which represent all possible variations in size, form and orientation of a target feature. The T-Map is a convex set of points in which each point uniquely maps to a variation of a target feature in the tolerance-zone. The model has been under development since 1998 and previous work has been done by several graduate students.

2.4 Tolerance Analysis:

Tolerance Analysis is the process of checking the extent and the nature of variation of a particular dimension or geometric feature of interest for the given GD&T scheme. This is usually done by the design engineer to endure that the form and fit of related parts and assemblies will satisfy the intended function. This can also throw light on some of the problems and wrong allocation of tolerances giving a chance for the designer to rectify any potential flaws that could turn out
to be expensive once the product has reached the shop floor. This is a very important step in the design cycle that must be communicated to the manufacturing personnel.

Tolerance analysis can be performed only if the meaning of the individual tolerancing specifications is understood. The actual step of determining the cumulative variation between two or more features cannot take place if there is no clear understanding of the GD&T specifications. In other words, tolerance analysis can answer questions such as the following:

- Will there be interference between the two mating parts? What will be the maximum and the minimum distance between features A and B? In this case, the features A and B can be on the same part or different parts.
- Will the pin fit within the hole or will the tab and slot mate without any problems?
- What are the worst case dimensionalities of the part?
- What is the source of a particular non-conformance? Does it arise within the part or is it due to the assembly?

These are some of the questions that a design engineer would like to answer before the manufacturing begins. He can use the results of tolerance analysis to review the GD&T scheme applied and make necessary corrections before it becomes too late. This method of analysis is often called the tolerance stack up. This is because, when the dimensions and tolerances are added together and they
stack up to reveal the possible variations. These stacked up dimensions and
tolerances form a chain of dimensions and tolerances and this chain can be
traversed. The variation of the analyzed dimension arises due to the accumulation
of dimensional and geometric variations in this tolerance chain. According to
Shen, Ameta et al [6], tolerance analysis includes determining the following:

- Contributors – the dimensions or features that cause variations in
  the analyzed dimension
- Sensitivities with respect to each contributor
- The percentage contribution to variation of each of the contributor
towards the total variation of the analyzed dimension
- Worst case variations, statistical distribution and acceptance rates

There are many approaches to compute the quantities mentioned above.
These approaches can be classified in general as follows [6, 17]:

- According to dimensionality – 1D, 2D, 3D
- Worst case (100% acceptance rate) and statistical (less than 100%
  acceptance rate)
- Dimensional and dimensional-geometric, according to the type of
  variations
- Part level and assembly level

The worst case and statistical methods are the common approaches to tolerance
analysis. The following sections show a brief survey of some of the popular
analysis methods as described in [6].
2.4.1 Tolerance Charts:

This is a type of worst case analysis and one of the most commonly used tools for tolerance analysis. The simplicity of the procedure involved has made it popular among draftsmen, designers and machinists. This procedure has been in use for decades in the industry and they calculate the maximum and minimum distances between two features. The modern charts for tolerance analysis include both dimensional and geometric tolerances in the direction of analysis. The procedure for 1D manual tolerance chart construction is summarized below:

- Setting up the Coordinate System: A start point is identified which serves as the reference coordinate system from which the measurements are to be made. Usually, the origin is set to be the left or the lower end point of the dimension of the analysis depending on the orientation of the analyzed dimension. This can be applied to both longitudinal and radial stacks. A stack indicator is then added at the start point of the stack. A stack indicator is a pair of opposing arrows with positive ‘+’ and negative ‘−’ signs attached to the arrows. This specifies when to add or subtract in the stack calculation. If a part dimension is in the positive arrow direction, then it is added and vice versa. By adopting such a convention, the final accumulation result will indicate if there is clearance (positive direction) or interference (negative direction) for assembly level stacks and if the feature exists or not for part level stacks.
• Identifying the stack path: A stack path is a chain of distances or part dimensions from the start point of the stack to the end point of the stack. The stack path should be constructed only with known dimensions, be the shortest possible chain of distances from the start to the end point and must be continuous.

• Creating the tolerance chart: The tolerance chart can be understood by making an analogy with an accountant’s spreadsheet, with each row representing the contributions of each of the dimensions in the stack path. The different columns of each row contain the maximum, minimum and range of each contributor. Each numerical column is added and the value is entered in the corresponding column of the last row to get the worst case value of the dimension under analysis. The last column of the last row should be equal to the sum of all ranges of the individual contributors and also to the difference between the maximum and minimum values of the analyzed dimension, thus acting as a check.

A simple tolerance chart described by [6] is shown in Table 1.1. When a stack contains both dimensions and geometric tolerances, the dimensions are entered as described above, but the geometric tolerances follow different rules based on rules for each tolerance class under different design specifications. Tolerance charts can be constructed for both assembly level and part level stacks. The worst case scenario for the part level stack is obtained from a single chart but an assembly level stack requires two charts. Another difference is that in assembly level stacks,
the parts are arranged in two extreme mating conditions (maximum or minimum).

For detailed explanation, one can refer to references [6, 29] for more details on the topic.

<table>
<thead>
<tr>
<th>Stack Matrix</th>
<th>Max Column</th>
<th>Min Column</th>
<th>( \Delta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contributor</td>
<td>Sign</td>
<td>Value</td>
<td>Sign</td>
</tr>
<tr>
<td>1</td>
<td>( V_{11} )</td>
<td>( V_{12} )</td>
<td>( U_{1} )</td>
</tr>
<tr>
<td>( \ldots )</td>
<td>( \ldots )</td>
<td>( \ldots )</td>
<td>( \ldots )</td>
</tr>
<tr>
<td>N</td>
<td>( V_{n1} )</td>
<td>( V_{n2} )</td>
<td>( U_{n} )</td>
</tr>
<tr>
<td>Sum</td>
<td>( \sum_{i=1}^{n} V_{i1} )</td>
<td>( \sum_{i=1}^{n} V_{i2} )</td>
<td>( \sum_{i=1}^{n} U_{i} )</td>
</tr>
</tbody>
</table>

Table 2.1 Sample tolerance chart (see Ref [6])

Fig. 2.1. Different configurations for maximum and minimum stackup conditions in an assembly (see Ref [6])

Numerous attempts have been made to automate the manual tolerance chart described above. This ranges from simple spreadsheet applications where the user has to key in the numbers to automation tools based on CAD models of
the part/assembly. Shen [12] describes a method where the tolerance charts have been automated. Automation involves loop detection, automatic assembly configuration and rule application for each GD&T class. Constraints such as dimension, size, geometric tolerances and mating conditions are identified and represented in a tree structure. This tree data structure can then be used to traverse and the corresponding GD&T data are retrieved and saved. The formulas are derived from the applicable rules and constraints and along with the rule checking algorithms, these are used for worst case and statistical analysis. [12, 17]

2.4.2 Parametric Tolerance Analysis:

This is a class of approach in which the functional dimension is expressed algebraically that relates the analyzed dimension to its contributors. The function can then be linearized or retained as such and then directly used for Monte Carlo simulations. This analysis yields results such as the list of contributors, sensitivities, percentage contributions and tolerance accumulations for worst case and statistical analyses.

2.4.2.1 Linearized Tolerance Analysis:

In a linearized analysis, the function, usually called the design function, is linearized about the variables using Taylor’s series expansion. Then the partial derivatives are calculated for each contributor. The derivatives are the sensitivities for each contributor from which worst case limits and variance can be computed. The dimension to be analyzed can be expressed as a function of contributors. The contributors are usually assumed to follow n sigma standard deviations for a
statistical analysis. The sensitivities and the percentage contributions show the effects of current tolerance allocation and they are an important tool for redesign.

2.4.2.2 Non-Linear Tolerance Analysis:

The linearized analysis neglect the higher order terms in the Taylor’s series expansion and assume that the sensitivities about the nominal value is nearly constant over the entire tolerance. Since in engineering applications the tolerances applied are at least couple of orders of magnitude smaller than the parent dimensions, this assumption is reasonably valid. But for applications in which this is not true, a linearized analysis would yield erroneous results. Also, the linearized models assume that the contributors are normally distributed. This assumption may not be true in many of the manufacturing processes. So under these circumstances, a non-linear analysis is performed using the Monte Carlo simulation methods. In Monte Carlo simulation, a random number generator is used to simulate the effects of manufacturing variations on assemblies and parts.

The process consists of the following steps [67]:

- The critical dimensions and their respective tolerances are identified.
- The contributors \((x_i)\) for resultant dimension and tolerances are specified. The distribution function \(p(x_i)\) of each contributor is also specified.
- A relationship function \(A = f(x_1, x_2, x_3)\) is formulated that relates the analyzed dimension with its contributors.
- A set of instance values for \(x_i\) are either picked from the tables corresponding the probability distributions of the contributors or using
random number generators to simulate their actual variations. The resultant value for the analyzed dimension is computed for each value of $x_i$ and compared to its tolerances for conformance.

- The previous step is repeated for a large sample size and then the results are analyzed using histograms or by fitting probability distribution functions and then calculate the percentage of results outside the tolerance limits.

The Monte Carlo simulation is usually performed with these linearized or non-linearized analyses. Since the accuracy of the histogram or the probability distribution function fit to the results depends on the sample size, it is evident that these methods depend on the number of simulations. A sample Monte Carlo simulation graph is shown in Fig. 1.2. The methods described in this section are very popularly used in commercial Computer Aided Tolerancing (CAT) systems.

![Fig 2.2. Non-linear tolerance analysis via Monte-Carlo simulation (see Ref [6])](image)

2.4.3 Parametric Approach Using CAD or Abstracted Geometry:

The parametric approaches described in the previous section can be combined with the direct use of parametric constraint model in CAD and creation
of abstracted feature-parameter model in CATS [67]. Some of the variations of this model are described below:

2.4.3.1 Direct Constraint Model in CAD:

In parametric CAD, constraint equations are based on geometric and dimensional relations. For a CAT system based on parametric CAD, the first step is to obtain the topological connectivity between the geometric elements. The second step is to define the geometric constraints between the model entities and hence the mathematical relationships between the variables of the model entities. The third step is to apply the general solution procedures to the applied constraints and derive a model that satisfies the constraints. Finally, variants of the model are created by changing the values of the constraint variables. If one of the variables is changed at a time, the particular variable’s sensitivity can be studied. Thus indirectly, we get the sensitivities of each of the contributors and hence a linearized or a non-linear analysis can be performed. Thus, this method just becomes an extension of parametric solid modeling.

2.4.3.2 Abstracted Feature Parameter Model:

Current CAD systems use constraints in 2D and 3D operations such as sweep and loft to create geometry. A 3D constraint solving model is not available that would enable tolerance analysis in 3D. So, for performing a 3D parametric tolerance analysis, an abstracted feature model has to be created separately from the actual CAD model. In this method, the features involved in the stack are abstracted to basic geometric entities such as points, planes, lines, circles and
cylinders. These geometric entities are then represented by their corresponding parameters. Finally, a set of distance and angular relations are constructed between the simplified entities and a constraint model is built for use in a parametric CAT system. A variational model is then created based on this 3D constraint model. In this variational model, the metric relations are expressed algebraically in terms of the parameters used to describe the geometric entities. For example, if a point in space is represented by \((X_i, Y_i, Z_i)\) and if a plane is represented by the equation \(A_iX + B_iY + C_iZ + D_i = 0\), then the perpendicular distance between the point and the plane is expressed as follows

\[
d = \frac{|A_iX_i + B_iY_i + C_iZ_i + D_i|}{\sqrt{(A_i)^2 + (B_i)^2 + (C_i)^2}}
\]

A Monte Carlo based simulation can be performed once an abstract parametric model is created. The general procedure for a CAT system based on this model is summarized as follows:

- Import the CAD model into the CAT system and hence erase any GD&T and assembly information in the original CAD model.
- Create an abstracted feature parametric model. Shown in Fig. 1.3.
- Define the tolerance specifications corresponding to the stack
- Specify the assembling sequence and methods
- Define the points and measurements i.e., the analyzed dimensions
- Vary the input model parameters and perform a Monte Carlo based simulation
A detailed explanation of this method is given in the references [6, 12]

![Diagram of assembly and feature variation]

**Fig 2.3.** Abstracted feature parameter method of tolerance analysis in a simple assembly. (a) A simple two part assembly with GD&T. (b) Model creation and feature variation. (c) Assembly constraint satisfaction and measurement computation (see Ref [6,12])

**2.4.4 Vector Loop or Kinematic Based Method:**

Chase et al. developed a method based on transformation matrices [70] which is usually called the kinematic approach to tolerance analysis. Three types of variations – dimensional, kinematic and geometric – are modeled in the vector loop model. In this model, the dimensions of the stack are modeled as vectors. Kinematic variations are the mating relations, also called the joints, in assembly time that occurs as a result of dimensional and geometric variations. The
geometric tolerances are modeled by adding micro degrees of freedom (DOFs) to the joints.

In this approach to tolerance analysis, first an assembly graph is created, which represents the mating conditions as joint types. This identifies the number of vector loops for an assembly analysis. The next step is to locate the features on the parts using the corresponding Datum Reference Frames. A vector loop is then created using these assembly graphs and the datum reference frames. The assembly constraints in the vector loop models can be represented as a concatenation of transformation matrices. This combination of matrices results in a set of non-linear equations. These non-linear equations can then be linearized using Taylor’s series expansion or retained as such for tolerance analysis purposes. The sensitivities and the percentage contributions can be obtained using matrix operations. A statistical analysis using this model assumes that the component distributions are Gaussian and an estimation of rejection rate in the assembly is possible. The estimate of a kinematic or assembly variation is usually treated as three standard deviations along with any deviation from the mean.

2.4.5 Variation Zone based Tolerance Analysis:

These are classes of models where the variations in the toleranced feature are associated with variables. Each degree of freedom of the feature within the tolerance zone is associated with a unique variable. The relationship between these variables representing the different DOFs is modeled by representing the relationship geometrically or algebraically. Numerous attempts have been made to
model each DOF with variables geometrically by different research groups. For a detailed explanation of these one can refer to [6].

However, the methods developed on this model are not suited to a comprehensive tolerance analysis. One main deficiency in these methods is that floating zones such as form tolerances do not have a representation. Also, these approaches do not cater to methods for developing stack up relations in an assembly. The variation zone approach which is otherwise a very capable model for performing tolerance analysis is plagued by such deficiencies. For this reason, Shah and Davidson proposed a bi-level model for modeling tolerances based on the variation zone approach [3]. Central to their model is the ASU Tolerance Map (T-Map) that has proved to be powerful in tolerance modeling in addition to satisfying all the requirements of the ASME standard. The next chapter is devoted to a detailed discussion of the Tolerance Map model.
3.1 Tolerance Modeling with T-Maps:

The previous sections provided an overview of the attempts to develop a mathematical model for representing tolerances. Of particular interest was the variation zone based methods. Davidson and Shah proposed a bi-level model for tolerance modeling using the Tolerance Maps (T-Maps). Of the two levels of models, the local or the metric model models the part variations applied to the feature and to represent the interaction of geometric controls such as size, form, orientation and position. The metric model has the Rule 1 of the ASME standard embedded and also enables the designer to study the trade-offs in tolerance allocation. The global or the topological model is used to relate all the control frames on the part or the assembly. The topological model also provides the basis for geometric validation of the dimensioning and tolerancing scheme. This bi-level model is shown to conform to the rules of the ASME standard, floating zones, bonus tolerances and datum precedence.

A T-Map can be defined as a hypothetical Euclidean volume of points, the shape, size and internal subsets of which represent all possible variations in size, form, orientation and position of a target feature. It is the set of points which form a convex region, which results from a one-to-one mapping from all variational possibilities of a feature within its tolerance zone. The T-Map forms the crux of the bi-level model developed at Design Automation Laboratory at Arizona State
University. A detailed understanding of the concept of T-Maps is necessary, because the concept of T-Maps is central to the ASU model for representing GD&T in the design, manufacturing and inspection stages. The following sections describe the construction of T-Maps for some of the major tolerance and feature types in greater detail.

3.2 Areal Coordinates:

Areal coordinates form the basis on which T-Maps are created. This section sets out the definition and describes the concept of areal coordinates. For a detailed explanation, one can refer to [18]. Areal coordinates are triples of numbers \((\sigma_1, \sigma_2, \sigma_3)\), corresponding to masses placed at the vertices of a reference triangle \(\sigma_1\sigma_2\sigma_3\). These masses then determine the point \(\sigma\), the geometric centroid of the masses and \(\sigma\) can be identified by \((\sigma_1, \sigma_2, \sigma_3)\). The masses we place at the points \(\sigma_1, \sigma_2, \sigma_3\) are correspondingly \(\lambda_1, \lambda_2, \lambda_3\). These masses may be positive or negative. The position \(\sigma\) of the centroid of the masses is uniquely determined by the linear combination

\[
(\lambda_1 + \lambda_2 + \lambda_3) \sigma = \lambda_1 \sigma_1 + \lambda_2 \sigma_2 + \lambda_3 \sigma_3
\]

This is valid under the condition \(\lambda_1 + \lambda_2 + \lambda_3 \neq 0\). We can make the point \(\sigma\) assume any position in the plane of \(\sigma_1, \sigma_2, \sigma_3\) by varying \(\lambda_1, \lambda_2, \lambda_3\). These three masses \(\lambda_1, \lambda_2, \lambda_3\) then become the barycentric coordinates of the point \(\sigma\). In other words, the three points \(\sigma_1, \sigma_2, \sigma_3\) become the basis points for the coordinate system and any point in the plane can be expressed as a linear combination of these. Since the position of \(\sigma\) depends only on two independent ratios of these
magnitudes, the coordinates \( \lambda_1, \lambda_2, \lambda_3 \), can be normalized by setting \( \lambda_1 + \lambda_2 + \lambda_3 = 1 \), thus giving the areal coordinate system. In this case, the points are used as the basis to locate any other points in the system. These are the types of areal coordinates that are being used to create T-Maps for planar surfaces on parts. Similarly, planes and lines can be used as the basis to describe different coordinate systems to locate planes and lines respectively, which we will see in the later sections. Fig. 3.1 describes the areal coordinates used for planar faces.

\[
\sigma = \lambda_1 \sigma_1 + \lambda_2 \sigma_2 + \lambda_3 \sigma_3 \\
\lambda_1 + \lambda_2 + \lambda_3 = 1
\]

\[
\sigma_1 = \{ \lambda_1 , \lambda_2 , \lambda_3 \} = \{1,0,0\} \\
\sigma_2 = \{ \lambda_1 , \lambda_2 , \lambda_3 \} = \{0,1,0\} \\
\sigma_3 = \{ \lambda_1 , \lambda_2 , \lambda_3 \} = \{0,0,1\}
\]

Fig. 3.1. Areal coordinates in a triangle

3.3 Tolerance Maps:

The T-Map for any combination of tolerances on a feature, irrespective of shape is constructed from a basis simplex and described using the areal coordinates. The number of degrees of freedom or the number of unique variations modeled determines the dimension of the T-Map. For example, if ‘n’ types of variations of a feature within the tolerance zone are considered, then the T-Map will be an n-D geometric entity. This means that the basis simplex will also be of n dimensions. Hence, to construct an n-dimensional T-Map, (n+1) basis points are needed.
3.3.1 Tolerance Map for a Planar Round Face:

3.3.1.1 Size Tolerance on Planar Round Face:

Fig 3.2(a) shows the cross section of a round bar of diameter $d$ with a size tolerance $t$. The tolerance zone has been exaggerated in the figure for better visualization. According to the ASME standard, all points on the end face of the round bar should lie in the zone limited by the planes $\sigma_1$ and $\sigma_2$ and within the cylindrical limit determined by the diameter of the bar. This region is defined as the tolerance zone. There are a couple of assumptions that we make before proceeding further on the development of the T-Map. First, we assume that the set of points that make up the top face is assumed to be a perfect feature, in this case an infinitely thin circular disc that can wobble within the limits of the tolerance zone. Second, the face at the other end of the bar is assumed to be of perfect form.

![Fig. 3.2. (a) A diametral cross-section of a round bar with size tolerance ‘$t$’. (b) A point map that represents half the planes in one diametral cross section of the tolerance zone of Fig. 3.2 (a) (see Ref [6])](image-url)
Fig. 3.2(a) shows the cross-section of the bar, where all the planes are parallel to the x axis and hence appear as lines. Initially, we consider only those planes that are either perpendicular to the z axis or tilted clockwise. From the figure, we can see that the plane $\sigma_3$ has the maximum tilt within the given tolerance zone. Planes $\sigma_1$ and $\sigma_2$ are the ones that are perpendicular to the z axis and at the extremes of the tolerance zone. Recollecting the concept of areal coordinates, we can now consider the three planes, $\sigma_1$, $\sigma_2$, $\sigma_3$ to be the basis planes that can define all other planes inside the tolerance zone. We map these to three corresponding basis points as depicted in Fig 3.2(b). The three vertices of the triangle each represent the three basis planes. Thus, any plane inside the tolerance zone can be represented as a point inside this triangle. All the planes that are perpendicular to the z axis are represented by the points that lie along the line $\sigma_1\sigma_2$. All points along the $\sigma_1\sigma_3$ represent the planes that are tilted clockwise and pass through the point B in the tolerance zone. All points along the line $\sigma_3\sigma_2$ represent the planes that are tilted clockwise and pass through the point D in the tolerance zone. Similarly, all points inside the triangle $\sigma_1\sigma_2\sigma_3$ represent planes that are tilted clockwise, but do not pass through the points B and D in the tolerance zone.
The two-dimensional set of planes in Fig. 3.2(a) i.e., those that are parallel to the x-axis and also either are perpendicular to the z-axis or are tilted CW, occurs in any diametral section throughout the tolerance-zone. In other words, the planes that are tilted counter clockwise can also be modeled as planes tilted clockwise by looking from the negative x direction. Therefore, all planes in the tolerance-zone at the end of the round bar is encountered by sweeping this set of planes defined by $\sigma_1$, $\sigma_2$, $\sigma_3$, a full turn about z axis. Similarly, the three-dimensional volume of points that maps to the entire set of planes in the tolerance-zone is the solid of revolution in Fig. 3.3 that is obtained by revolving the triangle in Fig. 3.2(b) a full turn about the line $\sigma_1\sigma_2$. The mapping of the planes in the tolerance zone to the points inside this solid of revolution is one-to-one. The solid has the form of two right-circular cones set base-to-base, both the vertex-to-vertex dimension and the radius of the base being the size-tolerance t.
Borrowing from the terminology for di-pyramidal polyhedra, the solid of revolution is a right-circular dicone. The triangle of Fig. 3.2(a), a diametral half-section of the T-Map, is labeled in Fig. 3.3 with points $\sigma_1$, $\sigma_2$, $\sigma_3$; also labeled are points $\sigma_4$, $\sigma_7$ and $\sigma_8$. Point $\sigma_7$ corresponds to that plane in Fig. that is parallel to the x-axis and has the maximum CCW tilt and goes through points A and D in Fig. 3.2(a), and rhombus $\sigma_1\sigma_7\sigma_2\sigma_3$ provides the entire diametral cross-section of the T-Map.

In the previous paragraph, we fixed a two dimensional areal coordinate system with the points $\sigma_1$, $\sigma_2$, $\sigma_3$. The point $\sigma_4$ is 90 degrees away from the point $\sigma_3$ and $\sigma_7$. Thus, we can use the point $\sigma_4$ as the fourth point in fixing the reference tetrahedron for a three dimensional areal coordinate system. Thus we have an areal coordinate system with the basis points $\sigma_1$, $\sigma_2$, $\sigma_3$, $\sigma_4$. Thus, any point $\sigma$ inside the tetrahedron can be represented as a weighted linear combination of these four basis points.

$$\sigma = \lambda_1 \sigma_1 + \lambda_2 \sigma_2 + \lambda_3 \sigma_3 + \lambda_4 \sigma_4$$

If all the $\lambda$’s are positive, then the point $\sigma$ lies inside the tetrahedron $\sigma_1\sigma_2\sigma_3\sigma_4$. For points outside this tetrahedron, the $\lambda$’s can assume any other values. For example, the areal coordinates of point $\sigma_7$ are (1, 1, -1, 0). One has to note here that the sum of all $\lambda$’s should be equal to 1. In Fig. 3.2(b) a symmetrical shape is chosen for the basis tetrahedron for three reasons.

First, it permits an orthogonal left-handed Cartesian system $(p', q', s)$ that describes the T-Map. The $p'$, $q'$ and $s$ axes can be regarded as the $x$, $y$ and $z$ axes
respectively. This coordinate system is related to a subset of coordinates (p, q, r, s) where p, q and r represent the direction cosines of a plane in the tolerance zone and \(-s\) is the distance of the plane from the origin. For example, the coordinates (p, q, r) of the plane \(\sigma_3\) are (0, t/d, 0). The coordinates (p’, q’, s) are (0, t, 0). Thus, the (p’, q’, s) coordinates can be regarded as a normalized coordinate system.

Second, our choice or basis points and hence the coordinate system decouples the variations in orientation from parallel size variations in the tolerance-zone, thereby simplifying the interpretation of a T-Map. The variations in p’ and the q’ directions represent rotations about lines in the y- and x-directions, respectively, in the tolerance-zone. But parallel displacement within the tolerance-zone is embedded with the s-coordinate in the T-Map. Consequently, every vertical line-segment in the T-Map represents a set of parallel planes in the tolerance-zone, i.e., a partial axial pencil of planes with its axis on the line at infinity in the xy-plane. And every horizontal line corresponds to a partial pencil of planes, the axis to which intersects the centerline of the part. The axis of each such partial pencil in the tolerance-zone is then oriented at right angles to its corresponding line-segment in the T-Map.

The third reason for the choice of such a coordinate system is that it allows the use of traditional formulas for the computation of metric quantities such as length, area, volume related to the T-Map. Ameer [8] has shown the volume of the basis tetrahedron to be \(t^3/6\) and the volume of dicone in areal coordinates to be \(2\pi\). The volumes of all the T-Map thus be computed using simple formulas. Also,
the T-Map for a single part has proved to be a convex. This further enables the use of principles of convex sets when performing further operations on the T-Maps.

3.3.1.2 Size Tolerance combined with Orientation on a Planar Round Face:

Sometimes, an orientation tolerance can also be specified on the planar round face in addition to the size tolerance. The maximum orientation or the tilt allowed by the size tolerance alone is the angle $\alpha = t/d$. A further limitation on orientation by an orientation tolerance $t''$ limits the angular variation further and cannot exceed the angle $t''/d$. According to the ASME standard, the orientation tolerance is defined by two parallel planes that are constrained in orientation with respect to a datum reference frame and are $t''$ apart. The zone defined by the orientation tolerance can float within the size tolerance zone and all the points of the planar round surface must lie within this floating zone. So, by applying an additional orientation tolerance, only the orientation is restricted and hence, there is a uniform change only along the $p'$ and $q'$ axes. The diametral section of the T-Map for the round bar with orientation tolerance is described in Fig. 3.4. The complete T-Map is then obtained by revolving the cross section about the line $\sigma_1\sigma_2$. This is shown in the Fig. 3.5. The T-Map for size combined with orientation is thus a truncated dicone, the radius of the dicone being the orientation tolerance $t''$ and the height remains the size tolerance $t$. 
3.3.1.3 Size Tolerance combined with Form on a Planar Round Face:

A flatness tolerance, t’ can be specified in addition to the size tolerance and according to the ASME standard, the flatness tolerance zone is defined by two parallel planes separated by a distance t’, within which all the points of the surface must lie. The limiting planes of the form tolerance can change both their position and orientation within the tolerance zone. This can be imagined as a
round coin, wobbling inside a metal can with the same radius. The T-Map for this condition does not change, however, there are now two internal subsets as depicted by the Fig. 3.6 showing the diametral cross section of the T-Map. The dashed triangle of height $t'$ is the actual limit defining the form tolerance and all the variations in the round face should be within this limit. The hashed triangle of height $(t - t')$ correspond to the positions the round face can float within the size tolerance zone.

Fig. 3.6. A diametral half section of the T-Map in Fig. 3.3 and its two diconical subsets of dimension $t'$ and $(t - t')$ (see Ref [14])

The Rule 1 of the ASME standard determines the amount of trade-off between the size tolerance and the form tolerance. According to the rule, a feature with maximum allowed size should have a perfect form. This rule is embedded into the formulation of the T-Map and can be explained from the Fig. 3.7. If the allowed form deviation is very small, i.e., $t' \ll t$, then the round face, which can be imagined as a coin has a lot of space in the size T-Map within which it is free to float. However, as the form deviation increases, the space available for the coin
to move decreases. At a point when the allowable form deviation is equal to the size tolerance, the coin can no longer float and the hashed triangle in the Fig disappears. This is illustrated by the Fig. 3.7, in which the upper and the lower dicones correspond to the form and position of the coin respectively. Fig. 3.7(a) represents the case when the coin is perfectly flat and upper dicone has degenerated to a single point. Thus, a perfectly flat plane can move all over the size T-Map. Fig 3.7(b) represents the case when the form deviations are small and hence the amount of position variation is diminished. Fig 3.7(c) represents the case when the form deviation is large and the position deviation is small. The dicone in the Fig 3.7(d) represents the case when the form deviations are equal to the size tolerance and hence the dicone corresponding to the position variation degenerates to a single point.

![Fig. 3.7. The trade-off between the internal subsets for form and range of positions within the T-Map (see Ref [8])](image)

**3.3.2 Tolerance Map for a Planar Rectangular Face:**

In the previous section, we described the T-Map for a round face. The tolerance zone for size tolerance on a rectangular face with cross sectional
dimensions \( dx \) and \( dy \) is depicted in Fig. Also, in this case we can assume the plane to be a rectangular card of dimensions \( dx \times dy \) similar to a coin for the round face. These are depicted by the planes \( \sigma_1, \sigma_2, \sigma_3, \sigma_4' \). Just like how we did for a round face, we select the basis planes in the tolerance zone corresponding to the planes with maximum, minimum sizes and maximum tilt about the \( x \) and \( y \) axes.

![Diagram of a rectangular bar with size tolerance showing the exaggerated tolerance zone and the different basis planes that define the T-Map for the feature.](image)

Fig. 3.8. The end face of the rectangular bar with size tolerance showing the exaggerated tolerance zone and the different basis planes that define the T-Map for the feature (see Ref [9]).

The T-Map for size tolerance of this rectangular face is depicted in the Fig. 3.8. The T-Map is a di-pyramid described by the points \( \sigma_1\sigma_2\sigma_7\sigma_4' \sigma_8' \). The construction procedure is similar to the one for the round face. However, the dimensions of the card also play a role in determining the size and shape of the T-Map. The height of the T-Map is the size tolerance \( t \). For a detailed explanation of
the construction of this T-Map, one can refer [9]. The form and orientation are also treated in a similar manner to the round face.

Fig. 3.9. The T-Map for the rectangular bar for a tolerance zone as defined in Fig. 3.8 (see Ref [9])

3.3.3 Tolerance Map for an Axis:

The tolerance-zone for the position of an axis is generated for a position tolerance $t$ on a hole or pin and is shown in Fig. 3.10 below. It is a thin cylinder defined by two circles $\bar{C}$ and $C$ of diameter $t$ located at each end within which all variations of the axis must lie. In the case of planar faces, the variations were modeled with planes and each plane mapped onto a point in the T-Map. For an axis, variations are modeled as lines with length $\ell$. A total of 4 degrees of freedom (2 tilts and 2 translations about the $x$ and $y$-axes) are available to lines in the tolerance-zone, and hence 5 basis points are required to define the T-Map. The basis points map to 5 basis lines $s_1$, $s_2$, $s_3$, $s_4$ and $s_5$ in the tolerance-zone.
The basis points are arranged on the vertices of a 4-D simplex upon which the T-Map is constructed.

Each line in the tolerance-zone is defined by Plücker coordinates. Plücker coordinates are described in Davidson and Hunt [24] and they represent a line as the axis of rotation for a rotating rigid body. Two vectors are needed to define the line, one for its direction (ω) and one for the coordinates ν of a point on the line. The vector ω may also be interpreted as the rotational velocity of the body about the line (axis), and ν is the translation velocity of that point on the body which is instantaneously at the origin. The notation of a Plücker coordinate is
$= (\omega; v) = (L, M, N; P, Q, R) \quad (1)$

The coordinates $L, M, N$ represent the direction ratios of the line, and coordinates $P, Q, R$ represent the components of tangential velocity of the point at the origin. Each line in the tolerance-zone for an axis has distinct Plücker coordinates associated with it. The six coordinates in Eq. (1) are superabundant, so, for any line the compatibility condition,

$$LP + MQ + NR = 0 \quad (2)$$

usually called the quadratic identity, must hold. It requires that the vector $(P, Q, R)$ be at right angles to the axis directions $(L, M, N)$. Using areal coordinates, each point in the T-Map for an axis can be described as a combination of the basis points. In other words, each line in the tolerance-zone for an axis can be described as a linear combination of the basis lines. The following equation allows for this representation

$$S = \lambda_1 S_1 + \lambda_2 S_2 + \lambda_3 S_3 + \lambda_4 S_4 + \lambda_5 S_5. \quad (3)$$

The $\lambda$’s are the areal coordinates which must satisfy $\sum \lambda_i = 1$ and the $S$ symbols represent the lines in the tolerance-zone. A graphical representation of the 4-D T-Map for an axis is not possible; however, 2-D and 3-D cross-sections can be shown by setting first two and then one of the $\lambda$’s equal to zero, respectively, in Eq. (3). In Fig. 3.11(a) results when $\lambda_4 = \lambda_5 = 0$, 3.11(b) results when $\lambda_4 = \lambda_5 = 0$ and 3.11(c) results when $\lambda_4 = \lambda_5 = 0$. A detailed explanation of the construction of this T-Map is given by Bhide [15].
3.4 T-Map Based Tolerance Analysis:

Consider the assembly of two cylinders are stacked one on top of the other in Fig. 3.12. A functional size tolerance is specified for the target face of the assembly, i.e., the top face of the top cylinder, relative to the datum of the assembly. The two cylindrical parts are assigned size tolerances $t_1$ and $t_2$ respectively and an orientation tolerance $t''$ on both.
3.4.1 Tolerance Accumulation:

The first step in tolerance accumulation is to develop T-Maps for the individual parts. The parts are both cylinders with an end planar round face. Since we also have an orientation tolerance $t''$ applied to both the faces with respect to a datum, we know from the previous sections that the T-Map for these two individual parts is a truncated dicone. Hence we have two T-Maps each of height and radius $t_1$ and $t''$ for the first part and $t_2$ and $t''$ for the second part. The next step is to represent the variations of each feature as the variation of the target face. This is an assembly with parts stacked on top of each other, the conformation is required for only the difference in the size of the target feature and the top face of the bottom cylinder. Once the conformed maps are obtained, the next step is to obtain the accumulation T-Map that represents all the variations of the parts in the assembly at the target face. A functional T-Map represents both the acceptable
range of one-dimensional dimension of interest and the acceptable limits to the
three-dimensional variational possibilities of the target feature consistent with it.
An accumulation T-Map represents all the accumulated three-dimensional
variational possibilities of the target feature which arise from allowable variations
on the individual parts in the assembly. The geometry of the target feature and a
specific value of the dimension of interest are used to establish a functional
surface that intersects the accumulation T-Map. The common points to the
geometric shapes of the accumulation TMap and the functional surface provide a
measure of all variational possibilities of manufacture of the parts, which will
give the specific value of the dimension of interest. This is obtained by a
Minkowski sum [6] of all the conformed T-Maps of all the parts. The Minkowski
sum is the vector sum of all the points of each T-Map with all the points of the
other T-Map. The accumulation map for the assembly shown in Fig. 3.12 is
shown in Fig. 3.13.

Fig. 3.13. Solid model of the accumulation T-Map for the assembly shown in Fig.
3.12.
3.4.2 Worst Case Analysis:

A functional T-Map that represents the intended variations of the target face can be developed for size ($t_f$) and orientation ($t_{f''}$) tolerances as shown in the previous section. To obtain the stack up equation, we overlay the functional and the accumulation T-Maps. Three stackup equations are obtained for this assembly from the contact points between the accumulation T-Map and functional T-Map. These stackup conditions are

$$t_f = t_1 + t_2$$

$$t_{f''} - t_2 = t_1 \left(\frac{dx}{dy}\right)$$

$$(t_{f''} - t_2)^2 = t_{f''}^2 + (t_1 - t_{f''})^2 \left(\frac{dx}{dy}\right)$$

The contributions of each of the parts can be obtained from the size of the worst case envelope circumscribing the conformed T-Map of each of the parts. The ratio of the contributions from each of the parts to the tolerance on each of the parts gives the sensitivities. A functional T-Map represents all the allowable variations of the target feature of the assembly. For the assembly to remain functional, the accumulated variations of the target feature should lie within the allowable variation of the target feature. This implies that the accumulation T-Map should lie completely within the functional T-Map for all the parts produced within specifications to satisfy the assembly function.

3.4.3 Statistical Analysis

The typical reason for statistical tolerance analysis and statistical tolerance allocation is to obtain larger values of tolerances than the worst case analysis but
still have only a small percentage of parts that do not meet the worst case specifications. Statistical tolerance analysis can be performed by assuming probability for manufacturing each of the points in the T-Map and hence each variation within the tolerance zone. For a comprehensive methodology of statistical analysis assuming geometric bias, refer Gaurav [17]. A schematic of the worst case and statistical tolerance analysis is shown in the Fig. 3.14.

![Diagram of Worst Case Tolerance Analysis and Obtain Frequency Distribution](image)

**Fig. 3.14. Steps to perform worst case and statistical tolerance analysis (see Ref [16, 17])**
The methods of statistical analysis described by Ameta [16, 17] take into account only the geometric bias of the given tolerance allocations. However, manufacturing biases also play a very important part in the tolerance analysis and evaluation of the tolerance allocation scheme. All the points within the T-Map i.e., each unique variation are assumed to have equal likelihood of manufacture. However, manufacturing biases due to factors such as tool wear, alignment of fixtures distort this equal likelihood. These cases are also accommodated in the method proposed by attaching weights to different points or regions in the T-Map space. Ameta [16, 17] summarizes the process of statistical tolerance analysis.
CHAPTER 4

TOLERANCE MAP CATALOG

4.1 Need for T-Map Catalog:

This chapter provides a catalog of the Tolerance Maps developed to date at the Design Automation Laboratory of Arizona State University. The T-Maps that are developed are not organized under any sort of classification and it is a strenuous task to collate the list of T-Maps available for different feature types and tolerance types. The list of available T-Maps has been organized according to the feature type and then for each feature type, they are organized according to the applied tolerance classes. The geometric properties of the T-Maps and the geometric construction algorithm for each of these T-Maps have been listed. A searchable web based database application has also been created with this catalog.

4.2 T-Maps for Planar Circular features:

This section gives the list of T-Maps for a planar round feature type. The tolerance classes for which T-Maps have been developed are for combinations of one or more of size, orientation and form. Table 4.1 gives the list of all the T-Maps.
<table>
<thead>
<tr>
<th>Tolerance Types</th>
<th>Geometric Properties</th>
<th>Geometric Construction Procedure</th>
<th>T-Map</th>
<th>Reference</th>
</tr>
</thead>
</table>
| Size Tolerance \((\pm) 't'\) | Dicone with Height \(t\) and radius \(t\). | A - Create Cone \((H = t/2, R = t)\)  
B - Create Cone \((H = -t/2, R = t)\)  
C – Boolean add A and B | ![T-Map](image) | Davidson, J.K., Mujezino vić, A., and Shah, J. J. [8] |
| Planar Circular, Orientation \((t')\) | Truncated dicone with Height \(t\) and radius \(t'\). | A – Create Cone \((H = t/2, R = t)\)  
B – Create Cone \((H = -t/2, R = t)\)  
C – Boolean add A and B  
D – Create Cylinder \((H = t, R = t')\)  
E – Boolean subtract D from C | ![T-Map](image) | Davidson, J.K., Mujezino vić, A., and Shah, J. J. [8] |
| Planar Circular, Size Tolerance \((\pm) 't'\) and Form \((t')\) | Dicone with Height \(t\) and radius \(t\). Internal subset for form. | A - Create Cone \((H = t/2, R = t)\)  
B - Create Cone \((H = -t/2, R = t)\)  
C – Boolean add A and B  
D - Create Cylinder \((H = t, R = t')\)  
E – Boolean subtract D from C  
Repeat A-C with \(R = t'\) for subset | ![T-Map](image) | Davidson, J.K., Mujezino vić, A., and Shah, J. J. [8] |
| Planar Circular, Form \((t')\) | Truncated dicone with Height \(t'\) and radius \(t'\). | A - Create Cone \((H = t/2, R = t')\)  
B - Create Cone \((H = -t/2, R = t')\)  
C – Boolean add A and B | ![T-Map](image) | Davidson, J.K., Mujezino vić, A., and Shah, J. J. [8] |
<table>
<thead>
<tr>
<th>Planar Circularity, Size (±) Tolerance (t), Parallelism (/)(t&quot;)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncated dicone with Height t and radius t&quot;.</td>
</tr>
<tr>
<td>A – Create Cone (H = t/2, R = t)</td>
</tr>
<tr>
<td>B – Create Cone (H = -t/2, R = t)</td>
</tr>
<tr>
<td>C – Boolean add A and B</td>
</tr>
<tr>
<td>D – Create Cylinder (H = t, R = t&quot;)</td>
</tr>
<tr>
<td>E – Boolean subtract D from C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Planar Circularity, Size (±) Tolerance (t), Perpendicularity (t&quot;) w.r.t a face</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncated dicone with Height t and radius t&quot;.</td>
</tr>
<tr>
<td>A – Create Cone (H = t/2, R = t)</td>
</tr>
<tr>
<td>B – Create Cone (H = -t/2, R = t)</td>
</tr>
<tr>
<td>C – Boolean add A and B</td>
</tr>
<tr>
<td>D – Create Cylinder (H = t, R = t&quot;)</td>
</tr>
<tr>
<td>E – Boolean subtract D from C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Planar Circularity, Size (±) Tolerance (t), Angularity (/)(t&quot;) w.r.t a face</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncated dicone with Height t and radius t&quot;.</td>
</tr>
<tr>
<td>A – Create Cone (H = t/2, R = t)</td>
</tr>
<tr>
<td>B – Create Cone (H = -t/2, R = t)</td>
</tr>
<tr>
<td>C – Boolean add A and B</td>
</tr>
<tr>
<td>D – Create Cylinder (H = t, R = t&quot;)</td>
</tr>
<tr>
<td>E – Boolean subtract D from C</td>
</tr>
</tbody>
</table>
| Planar Circular, Size \((\pm)\) Tolerance \((t)\), parallelism \((//)(tA")\) and perpendicularity w.r.t a face \((tB")\) | Truncated dicone with Height \(t\) and radius smaller of \(tA"\) and \(tB"\). | A – Create Cone \((H = t/2, R = t)\)  
B – Create Cone \((H = -t/2, R = t)\)  
C – Boolean add A and B  
D – Create Cylinder \((H = t, R = \text{smaller of } tA" \text{ and } tB")\)  
E – Boolean subtract D from C | Davidson, J.K., Mujezino vić, A., and Shah, J. J. [8] |
|---|---|---|---|
| Planar Circular, Size \((\pm)\) Tolerance \((t)\), perpendicularity \((tA")\) and perpendicularity w.r.t a face \((tB")\) | Truncated dicone with Height \(t\) and radius smaller of \(tA"\) and \(tB"\). | A – Create Cone \((H = t/2, R = t)\)  
B – Create Cone \((H = -t/2, R = t)\)  
C – Boolean add A and B  
D – Create Cylinder \((H = t, R = \text{smaller of } tA" \text{ and } tB")\)  
E – Boolean subtract D from C | Davidson, J.K., Mujezino vić, A., and Shah, J. J. [8] |
4.3 T-Maps for Planar Rectangular features:

This section gives the list of T-Maps for a planar rectangular feature type with tolerance classes such as size, orientation and form applied.

<table>
<thead>
<tr>
<th>Tolerance Types</th>
<th>Geometric Properties</th>
<th>Geometric Construction Procedure</th>
<th>T-Map</th>
<th>Reference</th>
</tr>
</thead>
</table>
| Planar Rectangular \((d_x, d_y)\), Size (\(\pm\)) Tolerance (t) | Dipyramid (Base is Rhombus), Height \(t\), width along \(q' = 2t\), width along \(p' = 2t \frac{d_x}{d_y}\) | A - Create Pyramid \((H = \frac{t}{2}, q' = 2t, p' = 2t \frac{d_x}{d_y})\)  
B - Create Pyramid \((H = -\frac{t}{2}, q' = 2t, p' = 2t \frac{d_x}{d_y})\)  
| Planar Rectangular \((d_x, d_y)\), Orientation Tolerance \((t'')\) | Dipyramid (Base is Rhombus), Height \(t''\), width along \(q' = 2t''\), width along \(p' = 2t\frac{d_x}{d_y}\) | A - Create Pyramid \((H = t''/2, \ q' = 2t'', \ p' = 2t\frac{d_x}{d_y})\)  
| B - Create Pyramid \((H = -t''/2, \ q' = 2t'', \ p' = 2t\frac{d_x}{d_y})\)  

| Planar Rectangular \((d_x, d_y)\), form Tolerance \((t')\) | Dipyramid (Base is Rhombus), Height \(t'\), width along \(q' = 2t'\), width along \(p' = 2t\frac{d_x}{d_y}\) | A - Create Pyramid \((H = t'/2, \ q' = 2t', \ p' = 2t\frac{d_x}{d_y})\)  
| B - Create Pyramid \((H = -t'/2, \ q' = 2t', \ p' = 2t\frac{d_x}{d_y})\)  
| Planar Rectangular ($d_x, d_y$), Size ($\pm$) Tolerance (t), Parallelism ($/\parallel/$(t") | Truncated Dipyramid, Height t, width along $q' = 2t'$, width along $p' = 2t \frac{d_x}{d_y}$ | A - Create Pyramid ($H = t/2$, $q' = 2t$, $p' = 2t \frac{d_x}{d_y}$)  
B - Create Pyramid ($H = -t/2$, $q' = 2t$, $p' = 2t \frac{d_x}{d_y}$)  
C – Boolean add A and B  
D – Do a loft' to create a cylinder of rhombic base with width $t''$.  
E – Boolean subtract D from C |
| Davids on, J.K., Mujezinić, A., and Shah, J. J. [9]| A - Create Pyramid ($H = t/2$, $q' = 2t$, $p' = 2t \frac{d_x}{d_y}$)  
B - Create Pyramid ($H = -t/2$, $q' = 2t$, $p' = 2t \frac{d_x}{d_y}$)  
C – Boolean add A and B  
D – Repeat the above steps for internal subset with $t'$. | Davids on, J.K., Mujezinić, A., and Shah, J. J. [9]|
| Planar Rectangular($d_x, d_y$), Size ($\pm$) Tolerance ($t$), Parallelism ($t''$) and form ($t'$) | A - Create Pyramid ($H = t/2$, $q' = 2t$, $p' = 2t \frac{d_x}{d_y}$)  
B - Create Pyramid ($H = -t/2$, $q' = 2t$, $p' = 2t \frac{d_x}{d_y}$)  
C - Boolean add A and B  
D - Do a loft' to create a cylinder of rhombic base with width $t''$.  
E - Boolean subtract D from C  
Repeat the same for internal subset | Davids on, J.K., Mujezinović, A., and Shah, J. J. [9] |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Dipyramid (Base is Rhombus), Height $t''$, width along $q' = 2t''$, width along $p' = 2t \frac{d_x}{d_y}$ Form is a subset</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Planar Rectangular($d_x$, $d_y$), Size ($\pm$) (t), Parallelism (‖)(t~A), Perpendicularity (⊥)(t~C).

<table>
<thead>
<tr>
<th>Description</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncated Dipyramid, Height t, width along q' = 2t<del>A, width along p' = 2t</del>C $d_x$ / $d_y$.</td>
<td></td>
</tr>
<tr>
<td>Planar Rectangular($d_x$, $d_y$), Size ($\pm$) (t), Parallelism (‖)(t<del>A), Perpendicularity (⊥)(t</del>C).</td>
<td></td>
</tr>
<tr>
<td><strong>A - Create Pyramid</strong> ($H = t/2$, $q' = 2t$, $p' = 2t d_x$ / $d_y$)</td>
<td><strong>B - Create Pyramid</strong> ($H = -t/2$, $q' = 2t$, $p' = 2t d_x$ / $d_y$)</td>
</tr>
<tr>
<td><strong>C – Boolean add A and B</strong></td>
<td><strong>D – Do a loft' to create a cylinder of rhombic base with width t'</strong>.</td>
</tr>
<tr>
<td><strong>E – Boolean subtract D from C</strong></td>
<td></td>
</tr>
<tr>
<td><strong>F – Create a block of width (t~C)</strong></td>
<td><strong>G – Boolean subtract F from E</strong></td>
</tr>
</tbody>
</table>

Planar Rectangular \((d_x, d_y)\),
Size \(\pm (t)\),
Parallelism \((//) (t)\),
Perpendicularity \((\perp) (t')\)

- **A** - Create Pyramid \((H = t/2, q' = 2t, p' = 2t \frac{d_x}{d_y})\)
- **B** - Create Pyramid \((H = -t/2, q' = 2t, p' = 2t \frac{d_x}{d_y})\)

**C** – Boolean add A and B

**D** – Do a loft' to create a cylinder of rhombic base with width \(t'\).

**E** – Boolean subtract D from C

**F** – Create a block of width \((t_c)\)

**G** – Boolean subtract F from E. Repeat the above steps for internal subset

| Planar Rectangular feature (dx, dy), Size (±t), Perpendicularity (∥)(t"B), Perpendicularity (∥)(t"C) | A - Create Pyramid (H = t/2, q' = 2t, p' = \(2t \frac{d_x}{d_y}\))  
| B - Create Pyramid (H = -t/2, q' = 2t, p' = \(2t \frac{d_x}{d_y}\))  
| C – Boolean add A and B  
| D – Create a block of width \((t'B)\)  
| E – Boolean subtract D from C  
| F – Create a block of width \((t'C)\)  
Table 4.2. T-Maps for planar rectangular features.

4.4 T-Maps for Planar Triangular Features:

This section gives the list of T-Maps for a planar triangular feature type.
Table 4.3. T-Maps for planar triangular features.

4.5 T-Maps for FOS Features:

This section gives the list of T-Maps for tolerance classes applied on Feature of Size (FOS) feature types.
Planar circular face (diameter $d$), Size ($\pm$) tolerance ($t_Q$) and orientation controlled by perpendicularity ($\perp$) tolerance ($t''_Q$) and position ($\mp$) tolerance ($t_s$) of an axis. Orientation is independent of position of face.

Diameter of Cylinder = $2dQ\beta$

where $\beta = \frac{1}{2}\alpha + \frac{t_0''}{dQ}$

and $\alpha = \frac{t_s}{h_{H1-E2}}$

$h_{H1-E2}$=height of the tolerance zone for the axis.

Create a cylinder of height $t_Q$ and dia $2dQ\beta$

<table>
<thead>
<tr>
<th>Wen Dong Jian [69]</th>
</tr>
</thead>
</table>

Table 4.4. T-Maps for FOS features.

4.6 T-Maps for Axis Elements

This section describes the T-Maps for circular holes or cylinders, called here as the axis elements since these elements are controlled by the position of their axes.
<table>
<thead>
<tr>
<th>Tolerance Types</th>
<th>Geometric Properties</th>
<th>Geometric Construction Procedure</th>
<th>T-Map</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position (+-peer) tolerance (t) of an axis</td>
<td>(4-D T-Map) Width along L’=t M’=t P=t Q=t</td>
<td>Create a parallelogram (rhombus) with parameter t for ( \lambda_2, \lambda_3, \lambda_4, \lambda_5 = 0 ) Create a circular plane for all other ( \lambda )'s = 0</td>
<td><img src="image1.png" alt="T-Map Image" /></td>
<td>Bhide [15]</td>
</tr>
<tr>
<td>Position (+-peer) tolerance (t) of an axis, with perpendicularity (⊥) tolerance (t&quot;)</td>
<td>(4-D T-Map) Width along L’=t&quot; M’=t&quot; P= t Q=t</td>
<td>Create a parallelogram (rhombus) for ( \lambda_2, \lambda_3, \lambda_4, \lambda_5 = 0 ) Create a circular plane for all other ( \lambda )'s = 0 Boolean subtract with parallelogram of parameter t&quot;</td>
<td><img src="image2.png" alt="T-Map Image" /></td>
<td>Bhide [15]</td>
</tr>
<tr>
<td>Position (†) tolerance (t) of an axis of a cylinder, tolerance on diameter, with parallelism (//) tolerance (t&quot;)</td>
<td>Create a parallelogram (rhombus) for ( \lambda_2, \lambda_3, \lambda_4, \lambda_5 = 0 ) Create a circular plane for all other ( \lambda )'s = 0 Boolean subtract with parallelogram of parameter t&quot;</td>
<td>Bhide [15]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(4-D T-Map) Width along ( L'=t) ( M'=t&quot; ) ( P=t ) ( Q=t )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Position (†) tolerance (t) of an axis of a cylinder, tolerance on diameter, with parallelism (//) tolerance (t&quot;C) and perpendicularity (⊥) tolerance (t&quot;A)</th>
<th>Create a parallelogram (rhombus) for ( \lambda_2, \lambda_3, \lambda_4, \lambda_5 = 0 ) Create a circular plane for all other ( \lambda )'s = 0 Sweep the planes within the parent planar T-Map without the orientation tolerance</th>
<th>Bhide [15]</th>
</tr>
</thead>
<tbody>
<tr>
<td>(4-D T-Map) Width along ( L'=t&quot;C ) ( M'= t&quot;A ) ( P=t ) ( Q=t )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

64
<table>
<thead>
<tr>
<th>Position (⊕) tolerance (t) of an axis of a cylinder, tolerance (τ) on diameter with MMC</th>
<th>(5-D T-Map) Width at LMC along L′ = M′ = P = Q = t + τ Width at MMC along L′ = M′ = P = Q = t Distance between LMC and MMC = τ</th>
<th>Not Applicable</th>
<th>Bhide [15]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position (⊕) tolerance (t) of an axis of a cylinder, tolerance (τ) on diameter</td>
<td>(5-D T-Map) Width along L′ = M′ = P = Q = t (as for case 10). Width along the 5th axis is τ. Internal Subset for cylindricity with max width along 5th axis 2τ</td>
<td>Not Applicable</td>
<td>Bhide [15]</td>
</tr>
<tr>
<td>Position (⊕) tolerance (t) of an axis of a cylinder, with cylindricity (⊙) (τ′) and also tolerance (τ) on diameter</td>
<td>(5-D T-Map) Width along L′ = M′ = P = Q = t (as for case 10). Width along the 5th axis is τ. Internal Subset for cylindricity with max width along 5th axis 2τ′</td>
<td>Not Applicable</td>
<td>Bhide [15]</td>
</tr>
<tr>
<td>Position (⊕) tolerance (t) of an axis of a cylinder, with cylindricity (⊙) (τ') and also (τ) on diameter with MMC</td>
<td>(5-D T-Map) Width at LMC along L' = M' = P = Q = t + τ Width at MMC along L' = M' = P = Q = t Distance between LMC and MMC = τ Internal Subset for cylindricity with max width along 5th axis 2τ'</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Not Applicable

Bhide [15]
### Tolerance on an axis established by position (+) tolerance (tE and tF) of two holes E and F, where position tolerance on F is with respect to E.

\begin{align*}
L' &= t_E + \frac{2t_F h_E}{4b + h_F} \\
M' &= t_E + \frac{2t_F h_E}{4b + h_F}
\end{align*}

Create a parallelogram (rhombus) for \( \lambda_2, \lambda_3, \lambda_4, \lambda_5 = 0 \)
Create a circular plane for all other \( \lambda \)'s = 0

---

[15] Bhide
| Cylindrical surface generated about an axis; straightness ($\pm$) tolerance ($t'$) controlled by total run-out ($\pm$) tolerance ($t$) | (4-D T-Map) Width along $L' = t'/2$ $M' = t'/2$ $P = t'$ $Q = t'$; shown are the 2-D cross-sections | Create two parallelograms of width $t'/2$ and Boolean add | Bhide [15] |
Table 4.5. T-Maps for axis elements.

### 4.7 T-Maps for Mid-Plane Elements

This section describes the T-Maps for 3D prismatic FOS, called here as the mid-plane elements since these elements are usually controlled by the position of their mid-planes.

<table>
<thead>
<tr>
<th>Tolerance Types</th>
<th>Geometric Properties</th>
<th>Geometric Construction Procedure</th>
<th>T-Map</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (±) tol (τ) and position (Θ) tol (t) for the mid plane of a tab or slot.</td>
<td>(4-D T-Map) Width along p'=q'=2s=2t</td>
<td>Not Applicable</td>
<td>Ameta [16]</td>
<td></td>
</tr>
<tr>
<td>Size ($\pm$) tolerance ($\tau$)</td>
<td>Position ($\Theta$) tolerance ($t$) for mid plane of a tab or slot with MMC for position.</td>
<td>(4-D T-Map) Width at LMC along $p'=q'=2s=2(t+\tau)$ Width at MMC along $p'=q'=2s=2t$ Distance between LMC and MMC = $\tau$</td>
<td>Not Applicable</td>
<td></td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>-----------------</td>
<td></td>
</tr>
</tbody>
</table>
### Size (±) tolerance (t)
**Position (⊕) tolerance (τ)** for mid plane of a tab or slot with MMC for position and secondary datum (with size tolerance T).

#### Width (5-D T-Map)
- Width at LMC for position and LMC for datum along p'=q'=2s= 2(t+τ+T)
- Width for MMC position and datum along p'=q'=2s= 2t

#### Distance
- Distance between LMC for position and MMC for position = τ
- Distance between LMC for datum and MMC for datum = T

<table>
<thead>
<tr>
<th>Size (±) tolerance (t)</th>
<th>Position (⊕) tolerance (τ) for mid plane of a tab or slot with MMC for position and secondary datum (with size tolerance T).</th>
<th>Width (5-D T-Map)</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Width at LMC for position and LMC for datum along p'=q'=2s= 2(t+τ+T)</td>
<td>Width for MMC position and datum along p'=q'=2s= 2t</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2(t+τ+T)</td>
<td>2t</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| | Not Applicable |
| Davidson, J.K., Mujezinović, A., and Shah, J. J., Ameta [80] |

#### Table 4.6. T-Maps for mid-plane elements.

### 4.8 T-Maps for Circular and Cylindrical Elements:

This section describes the list of T-Maps developed for circular and cylindrical features and tolerances that apply to them such as circular and total run-out and circularity in combination with other tolerance classes.
<table>
<thead>
<tr>
<th>Tolerance Types</th>
<th>Geometric Properties</th>
<th>Geometric Construction Procedure</th>
<th>T-Map</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cylindrical, curved profile generated about an axis; size (±) tolerance (τ)</td>
<td>Any line connecting two basis points γ₁ and γ₂ fixed on the Δr-axis. Length between points is (½)τ.</td>
<td>Create a line of width (½)τ.</td>
<td><img src="image1.png" alt="T-Map" /></td>
<td>Clasen [13]</td>
</tr>
<tr>
<td>Cylindrical, curved profile generated about an axis with circular run-out (±) tolerance (t)</td>
<td>Equilateral triangle with basis points γ₁, γ₂, γ₃. Length of hypotenuse is t'. Height of triangle is (½)t.</td>
<td>Create Triangular face with Length of hypotenuse is t'. Height of triangle is (½)t.</td>
<td><img src="image2.png" alt="T-Map" /></td>
<td>Clasen [13]</td>
</tr>
<tr>
<td>Conical; circular run-out (±) tolerance (t)</td>
<td>Equilateral triangle with basis points γ₁, γ₂, γ₃. Length of hypotenuse is tₑₒ. Height of triangle is (½)tₑₒ. tₑₒ = t/cos(θ).</td>
<td>Create Triangular face Length of hypotenuse is tₑₒ. Height of triangle is (½)tₑₒ. tₑₒ = t/cos(θ).</td>
<td><img src="image3.png" alt="T-Map" /></td>
<td>Clasen [13]</td>
</tr>
</tbody>
</table>
| Cylindrical, curved profile generated about an axis with circularity \((\pm) (t')\) | Equilateral triangle with basis points \(\gamma_1, \gamma_2, \gamma_3\). Length of hypotenuse is \(t'\). Height of triangle is \((\frac{1}{2})t'\). | Create triangular face
Length of hypotenuse is \(t'\).
Height of triangle is \((\frac{1}{2})t'\). | Clasen [13] |
|---|---|---|---|
| Cylindrical, curved profile generated about an axis; size \((\pm)\) tolerance \(\tau\), circular run-out \((\pm) (\delta)\) \(\gamma_1, \gamma_2, \gamma_3, \gamma_4\). Height of rectangle is \((\frac{1}{2})\tau\). Length of rectangle is \((\frac{1}{2})\tau\). | Create rectangular face.
Height of rectangle is \((\frac{1}{2})\tau\).
Length of rectangle is \((\frac{1}{2})\tau\). | Clasen [13] |
| Cylindrical, curved profile generated about an axis; size \((\pm)\) tolerance \(\tau\), circular run-out \((\pm) (\delta)\) \(\gamma_1, \gamma_2, \gamma_3, \gamma_4\). Height of rectangle is \((\frac{1}{2})\tau\). Length of rectangle is \((\frac{1}{2})\tau\). Triangular subset for circularity has same dimensions as the T-Map for circularity. | Create rectangular face
Height of rectangle is \((\frac{1}{2})\tau\).
Length of rectangle is \((\frac{1}{2})\tau\).
Create triangular face for the form subset | Clasen [13] |
<table>
<thead>
<tr>
<th>Planar circular face; circular run-out (±) tolerance (t)</th>
<th>Equilateral triangle with basis points $\gamma_1$, $\gamma_2$, $\gamma_3$. Length of hypotenuse is $t$. Height of triangle is $(\frac{1}{2})t$.</th>
<th>Create Triangular face Length of hypotenuse is $t$. Height of triangle is $(\frac{1}{2})t$.</th>
<th>Clasen [13]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planar circular face; size (±) tolerance (τ)</td>
<td>Equilateral triangle with basis points $\gamma_1$, $\gamma_2$, $\gamma_3$. Length of hypotenuse is $\tau$. Height of triangle is $(\frac{1}{2})\tau$.</td>
<td>Create Triangular face Length of hypotenuse is $\tau$. Height of triangle is $(\frac{1}{2})\tau$.</td>
<td>Clasen [13]</td>
</tr>
<tr>
<td>Planar circular face; circular run-out (±) tolerance (i); size (±) tolerance (τ)</td>
<td>Equilateral triangle with basis points $\gamma_1$, $\gamma_2$, $\gamma_3$. Length of hypotenuse is $\tau$. Height of triangle is $(\frac{1}{2})\tau$. Triangular subset is T-Map for circular runout on a planar segment.</td>
<td>Create Triangular face Length of hypotenuse is $\tau$. Height of triangle is $(\frac{1}{2})\tau$. Create triangular face for the form subset</td>
<td>Clasen [13]</td>
</tr>
</tbody>
</table>
| Cylindrical, curved profile generated about an axis; circular run-out ($\pm$) tolerance ($\delta$); T-Map represents the angular position of run-out error on part. | Right circular dicone with height and diameter $t$. | A - Create Cone ($H = t/2, R = t$)  
B - Create Cone ($H = -t/2, R = t$)  
C - Boolean add A and B. | Clasen [13] |
|---|---|---|---|
| Planar circular face; circular run-out ($\pm$) tolerance ($\delta$); T-Map represents angular location of run-out. | Right circular dicone with height and diameter $t$. | A - Create Cone ($H = t/2, R = t$)  
B - Create Cone ($H = -t/2, R = t$)  
C - Boolean add A and B. | Clasen [13] |
<table>
<thead>
<tr>
<th>Cylindrical, curved profile generated about an axis; size ((\pm)) tol ((\tau)), circular run-out ((\pm)) tol ((\delta)); T-Map represents the angular position of runout error on part</th>
<th>Right cylinder with height (\tau) and diameter (\delta). Create cylinder with height (\tau) and diameter (\delta).</th>
<th>Clasen [13]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cylindrical surface generated about an axis; size ((\pm)) tolerance ((\tau))</td>
<td>Any line connecting two basis points (\gamma_1) and (\gamma_2) fixed on the (\Delta r)-axis. Length between points is ((\frac{1}{2})\tau). Points represent cylinders. Create line Length between points is ((\frac{1}{2})\tau).</td>
<td>Clasen [13]</td>
</tr>
<tr>
<td>Cylindrical surface generated about an axis; total run-out ((\pm)) tolerance ((\delta))</td>
<td>Right tetrahedron with height (\delta) and base length (\delta) and base altitude ((\frac{1}{2})\delta). Create Pyramid, height (\delta) and base length (\delta) and base altitude ((\frac{1}{2})\delta).</td>
<td>Clasen [13]</td>
</tr>
<tr>
<td>Cylindrical surface generated about an axis; cylindricity $(\pm), \tau$ tolerance $(t')$</td>
<td>Right tetrahedron with height $t'$ and base length $t'$ and base altitude $(\frac{1}{2})t'$</td>
<td>Create Pyramid. height $t'$ and base length $t'$ and base altitude $(\frac{1}{2})t'$</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Cylindrical surface generated about an axis; size $(\pm)$ tolerance $(\tau)$, total run-out $(\pm)$ tolerance $(\bar{t})$</td>
<td>Extruded right triangle with base $(\frac{1}{2})t'$ height $t'$ and length $(\frac{1}{2})\tau$</td>
<td>Make triangular face with base $(\frac{1}{2})t'$ height $t'$ Sweep for length $(\frac{1}{2})\tau$</td>
</tr>
<tr>
<td>Cylindrical surface generated about an axis; size $(\pm)$ tolerance $(\tau)$, total run-out $(\pm)$ tolerance $(\bar{t})$, cylindricity $(\pm)$ tolerance $(t')$</td>
<td>Extruded right triangle with base $(\frac{1}{2})t'$ height $t'$ and length $(\frac{1}{2})\tau$; right tetrahedron subset for cylindricity is contained within</td>
<td>Make triangular face with base $(\frac{1}{2})t'$ height $t'$ Sweep for length $(\frac{1}{2})\tau$ For form subset, Create Pyramid. height $t'$ and base length $t'$ and base altitude $(\frac{1}{2})t'$</td>
</tr>
<tr>
<td>Planar circular face; total runout (±) tolerance (( \bar{t} ))</td>
<td>Equilateral triangle with basis points ( \gamma_1, \gamma_2, \gamma_3 ). Length of hypotenuse is ( \bar{t} ). Height of triangle is ((\frac{1}{2})\bar{t}).</td>
<td>Create triangular face. Length of hypotenuse is ( \bar{t} ). Height of triangle is ((\frac{1}{2})\bar{t}).</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Planar circular face; total runout (±) tolerance (( \bar{t} )); size (±) tolerance (( \tau ))</td>
<td>Equilateral triangle with basis points ( \gamma_1, \gamma_2, \gamma_3 ). Length of hypotenuse is ( \tau ). Height of triangle is ((\frac{1}{2})\tau). Triangular subset is T-Map for total run-out on a planar segment.</td>
<td>Create triangular face. Length of hypotenuse is ( \tau ). Height of triangle is ((\frac{1}{2})\tau).</td>
</tr>
</tbody>
</table>

Table 4.7. T-Maps for circular and cylindrical features.
CHAPTER 5

ANALYSIS OF METROLOGY PRACTICES

5.1 Dimensional Metrology:

The objective of dimensional metrology is to check manufactured parts for conformance to tolerance specifications on the drawings. A phenomenal increase in quality consciousness fuelled by intense competition among companies has resulted in the past couple of decades seeing a tremendous increase in the hardware and software capabilities used in metrology. Srinivasan [40] states two important axioms that form the reason as to why we need to make measurements and hence inspect quality. First, all manufacturing processes produce parts that can never have an ideal form and second, no measurement can be considered to be absolutely accurate. These are the reasons for having tolerances in the design specifications and a dedicated standard, the ASME standard [1, 2], which governs how these tolerances are used and interpreted. There are numerous methods for measuring these tolerances on the shop floor. This paper also provides a brief survey of measurement methods. The measurement methods and the tolerance standard have evolved over time to be consistent with each other.

In the previous sections, we saw how GD&T, when applied to product designs, need to represent the functional intent of the designer. We came across many models for GD&T representation with varying degrees of conformance to the ASME standard. The designers’ functional intent is communicated to the manufacturing and inspection stages through these geometric tolerances. These
tolerances ensure the form, fit and functionality as envisioned by the designer. Since the GD&T symbols and methods at least theoretically provide an unambiguous representation of the product functionality, it is the duty of the inspection engineer or technician to check the parts in such a way that reflects the intent of the designer. So the task of the person making the measurement can be summarized as follows.

- Understand the geometric tolerancing symbols and rules that govern them
- Understand the capability of the various inspection and gaging equipments at his disposal
- Determine the right method or tool to evaluate the part or feature
- Interpret the results without ambiguity

The fundamental take away of this discussion is that one has to understand the set up for measurement and ensure that the datum references are contacted properly and the tolerance zone for the part are simulated properly before making measurements. This is because the datums and the tolerance zone of the applied tolerance determine the appropriate inspection methods and techniques. Following these lessons will go a long way in helping the inspector performing an accurate and functional inspection of geometric tolerances. The next section gives a detailed survey of manual measurement practices. The sections that follow explore GD&T measurement using CMM machines. A survey of algorithms used in CMM software is presented and some problems reported by metrology research groups are discussed. The last section of the chapter presents the results of an
investigative study of the algorithms used in commercial CMM software and their accuracy and performance are reported.

5.2 Definition for Tolerance Classes in the GD&T Standard:

5.2.1 Size Tolerance:

The limits of size is explained in the standard in the form of Rule 1.

What the standard says:

“The limits of size of a feature prescribe the extent within which variations of geometric form, as well as size, are allowed”

The size control applies to only regular features of size and the actual local size of an individual feature at each cross section of the FOS should be within the size tolerance limits. The size tolerance does not have any datums associated with it. The overall measurement of the size is made in the nominal cross sectional direction of the FOS, while the local cross-sections are measured in the direction normal locally to the opposing surfaces of the FOS. It is evident from this definition that all the points of the feature should lie between the limits of maximum material condition defined by the size tolerance. Also, where the actual local size of a feature of size departs from MMC towards LMC, a local variation in form is allowed equal to the amount of departure. These concepts are depicted in Fig. 5.1 below.
Fig. 5.1 Size Tolerance on a feature of Size

The definitions in the standard become ambiguous in some cases. This is a much debated issue and active research is still being carried on by the tolerancing community. For further discussion on the ambiguity of size tolerances and their multiple interpretations, the reader is referred to Voelcker [50].

5.2.2 Orientation Tolerances:

An orientation tolerance controls parallel, perpendicular or any other angular relationship between features. Orientation tolerances are always specified on features by relating them to a datum. The datum can be a plane or FOS features such as holes, pins, tabs and slots. In case of FOS features used as datums, the derived elements such as the axis or the mid-planes are used to establish the datums. These tolerances specify a zone within which the considered feature, its line elements, its axis or its center plane must be contained. An orientation tolerance also controls flatness to the extent of the orientation tolerance applied. Control of orientation does not control the location of the features and consideration must also be given to the control of orientation already
established through other tolerances such as location, run-out and profile tolerances.

**5.2.2.1 Parallelism Tolerance:**

Parallelism is the condition of a surface or center plane, equidistant at all points from a datum plane.

What the standard says:

“A tolerance zone defined by two parallel planes, parallel to the datum plane or axis, within which the surface or center plane of the considered feature must lie.

A tolerance zone defined by two parallel lines, parallel to the datum plane or axis, within which the line element of the surface must lie.

A cylindrical tolerance zone parallel to one or more datum planes or datum axis, within which the axis of the feature must lie.

A tolerance zone defined by two parallel lines parallel to the datum plane or axis, within which the line element of the surface must lie”

Fig. 5.1 shows parallelism tolerance applied on a planar surface and the corresponding tolerance zone.
5.2.2.2 Perpendicularity Tolerance:

Perpendicularity is the condition of a surface, center plane, or axis at a right angle to a datum plane or axis.

What the standard says:

“A tolerance zone defined by two parallel planes, perpendicular to the datum plane or axis, within which the surface or center plane of the considered feature must lie.

A tolerance zone defined by two parallel lines, perpendicular to the datum plane or axis, within which the line element of the surface must lie.

A tolerance zone defined by two parallel planes perpendicular to a datum axis, within which the axis of the considered feature should lie.

A cylindrical tolerance zone perpendicular to a datum plane, within which the axis of the considered feature must lie.”

Fig. 5.2. Parallelism tolerance on a planar surface
5.2.2.3 Angularity Tolerance:

Angularity is the condition of a surface, center plane or axis at a specified angle (other than 90°) from a datum plane or axis.

What the standard says:

“A tolerance zone defined by two parallel planes that are at specified basic angle from one or more datum planes or a datum axis, within which the surface or the center plane of the considered feature must lie.

A tolerance zone defined by two parallel planes at the specified basic angle from one or more datum planes or datum axis, within which the axis of the considered feature must lie.

A cylindrical tolerance zone at the specified basic angle from one or more datum planes or a datum axis, within which the axis of the considered feature must lie.

A tolerance zone defined by two parallel lines at the specified basic angle from a datum plane or axis within which the line element of the surface must lie”
5.2.3 Position Tolerance:

Position Tolerances are usually applied to FOS elements such as holes, pins, tabs and slots. The position of the FOS elements can be controlled using their extremal surfaces or with their medial elements. In case of holes and pins, the position is controlled using the medial axis. For tabs and slots, the position can be controlled either by the side surfaces or by their derived mid-planes. The definition is given by the standard for both the cases.

What the standard says:

“While maintaining the specified size limits of the hole, no element of the hole surface shall be inside a theoretical boundary located at true position.

The axis of the hole must fall within a cylindrical tolerance zone whose axis is located at true position. The diameter of this zone is equal to the positional tolerance. This tolerance zone also determines the limits of variation of the attitude of the axis of the hole in relation to the datum surface.”
While maintaining the specified width limits of the feature, its center plane must be within a tolerance zone defined by two parallel planes equally disposed about true position, having a width equal to the positional tolerance. This tolerance zone also defines the limit within which variations in attitude of the center plane of the feature must be confined.

While maintaining the specified width limits of the feature, no element of its side surfaces shall be inside a theoretical boundary defined by two parallel planes equally disposed about true position and separated by a distance equal to \( W \)

Where, \( W = (\text{MMC size}) \) plus or minus (Position Tolerance value), depending on the feature being internal or external.

A positional tolerance defines a zone in which the center axis or center plane is permitted to vary from true (theoretically exact) position. Basic dimensions establish the true position from datum features and other related features. A positional tolerance is the total permissible variation in location of a feature about its exact location. For cylindrical features such as holes and outside diameters, the positional tolerance is generally the diameter of the tolerance zone in which the axis of the feature must lie. For features that are not round, such as slots and tabs, the positional tolerance is the total width of the tolerance zone in which the center plane of the feature must lie. The position tolerance also controls the orientation in addition to the location. For example, the axis of the hole can be tilted with respect to its true position, but the extent of the tilt is restricted such
that the axis remains within the position tolerance zone. The position tolerance defined above applies when the FOS is at MMC. However, when the actual size of the feature deviates from the MMC towards LMC, this results in additional positional tolerance. This is called the bonus tolerance. A position tolerance zone is illustrated in the Fig. 5.4 and Fig. 5.6.

Fig. 5.5. Position Tolerance on two holes
Fig. 5.6 Explanation for position tolerance on a slot w.r.t center plane and side surfaces (see Ref [1])

Fig. 5.7 Explanation for position tolerance on a hole (see Ref [1])
5.2.4 Form Tolerances:

5.2.4.1 Flatness Tolerance:

Flatness is the condition of a surface having all its elements in one plane.

What the standard says:

“A Flatness tolerance specifies a tolerance zone defined by two parallel planes within which the surface must lie.”

An example of flatness tolerance is depicted in Fig. 5.8. Flatness tolerance does not have any datums associated with it. Hence the two parallel planes shown in Fig. can have any orientation but should be separated by a maximum distance of the flatness tolerance value.

![Fig. 5.8. Flatness tolerance applied on a planar surface](image)

5.2.4.2 Straightness Tolerance:

Straightness is a condition where an element of a surface, or an axis, is a straight line. A straightness tolerance is applied in the view where the elements to be controlled are represented by a straight line.

What the standard says:
“A Straightness tolerance specifies a tolerance zone defined by two parallel lines within which the considered element or derived median line must lie.”

Fig. 5.9. Straightness tolerance applied on a cylindrical bar

When a diameter modifier is used with the straightness tolerance, the derived medial line of the FOS must lie within a cylindrical tolerance zone of diameter specified by the straightness tolerance. Each circular element of the surface must be within the specified limits of size. In other words, adding a diameter controls the straightness through the medial axis while the method without the diameter modifier controls straightness through the opposing surfaces of the FOS.

5.3 Datums and Datum Reference Frame (DRF):

The criteria and the algorithms used to establish the datums are an integral part of inspection. The importance of datums is paramount because they are used as the basis for establishing the geometric relationship of features of a part. Before
we proceed, we can recollect the definitions of a few important concepts that are associated with establishing datums. The standard defines the datums as,

“Datums are theoretically exact points, planes and axes. These elements exist within a framework of three mutually perpendicular intersecting planes known as the datum reference frame.” [1, 2].

Datums are thus imaginary, but they are created from actual features on the part. An actual feature consists of one or more surfaces of a manufactured part that corresponds to one of datums, dimensions or tolerances. Datum reference frames are coordinate systems used to locate and orient part features. The ASME standard specifies that datum reference frames are established from a set of part features. A feature used for creating datum reference frame is referred to as a datum feature. A datum feature is an actual feature of the part used to establish a datum and they are selected in such a way that they define a functional relationship with another feature within the part or the assembly. Datum features intend to remove all the six degrees of freedom of rigid motions. By contacting a datum with the corresponding datum feature, the degrees of freedom can be subsequently reduced. The order of precedence of a datum feature determines the contacting sequence.

The primary datum contacts the corresponding datum feature first, leaving certain degrees of freedom undetermined. The secondary datum contacts the corresponding datum feature next, determining certain degrees of freedom left by the primary datum. The tertiary datum next contacts the corresponding datum
feature and completely removes all the available degrees of freedom left by the primary and the secondary datums. The datum feature of size, such as tabs, slots, pins, holes and spherical features can be referenced at MMC (Maximum Material Condition), or LMC (Least Material Condition), or RFS (Regardless of Feature Size). If one or more datum features are at MMC or LMC, a set of candidate datum reference frames can be established. It should be understood that the method of establishing datum reference frames depends on the type of datum feature, the precedence of datum feature, and material condition of feature of size.

The concept of DRF is illustrated in the Fig. 5.9. The ASME standard defines that for fixing a primary datum plane, the plane has to contact the datum feature at a minimum of three high points. This is depicted in Fig. 5.9(a). For a secondary datum plane, the datum should contact the secondary datum feature at a minimum of two high points. Also, this secondary datum plane should be perpendicular to the primary datum. This is depicted in Fig. 5.9(b). The tertiary datum plane should be perpendicular to both the primary and secondary datum planes and should contact the tertiary datum feature at least at one point. This is depicted in Fig. 5.9(c). For FOS features being used as the primary datum, the simulated datum is the axis or center plane of the true geometric counterpart. For cylindrical features, the true geometric counterpart is the maximum inscribed or minimum circumscribed depending on internal or external features respectively. For tabs and slots, it is the pair of planes separated by the minimum and maximum distances respectively. For FOS used as secondary and tertiary datums,
the procedure is the same except that the simulated datums are now constrained to the primary and secondary datums. If Maximum Material Condition (MMC) or Least Material Condition (LMC) is specified for the datums, then the true geometric counterpart is determined based on the MMC or LMC virtual condition.

Fig. 5.10. Datum Reference Frame (DRF) defined by three datum planes

5.4 Manual Inspection Practices:

This section describes the manual inspection practices that are common in the industry for different tolerance types. The section provides a review of methods described in metrology books, inspection manuals and feedback from inspection technicians in the shop floor. There are no comprehensive standards for inspection of GD&T. The ASME Standards [1, 2] are the ones that describe the definition of tolerance classes and to an extent the measurement of these
tolerances. There are a few standards [83, 84, 85] that define the design of gage blocks and dial indicators that also explain indirectly the interpretation of the Standards [1 and 2]. The actual measurement practices on the shop floor are shown in the photographs in the ensuing sections. The scope of the review is limited to measurement of size, orientation, form and position tolerances that apply to planar and cylindrical features.

5.4.1 Fixing Datums in Inspection:

Fixing datums is one of the most important tasks in inspection. Simulated datums are various inspection equipments that include surface plate accessories such as surface plates, gage blocks, precision parallels, sine plate surfaces, gage pins and rings that are brought into contact with a part datum feature to establish datum points, lines and planes [1, 2]. For example, a part that has a datum surface is brought into contact with a surface plate and hence the surface plate can be considered as the datum and all measurements can be made with respect to the surface plate. Actually, it is the contact points of the datum feature and the surface plate that creates the datum. Since surface plates are highly accurate, they can be used to approximate a datum. This is illustrated in the Fig. 5.10.
Other accessories such as hole and L-blocks blocks could be used to simulate datums. This is shown in Fig. 5.10 and Fig. 5.11.
Fig. 5.13. Using L-blocks for simulating datums

For cylindrical features, a set of gage pins and V-blocks can be used to simulate the datums. For example, Fig. 5.13 shows the use of an appropriate gage pin being used as a datum when a hole is being used as a datum. In Fig. 5.14 the use of V-block to simulate a datum when an external feature such as a pin is used as a datum.

Fig. 5.14. Using pin gauges to simulates axis datums

97
Fig. 5.15. Using pin gauges and V-blocks to simulates axis datums

5.4.2 Size Tolerances:

The size tolerances in general can be inspected using go/no-go gages, micrometers, vernier calipers or using gage blocks and dial indicators [20, 21, 22, 23]. When using gages, we must ensure that the feature is completely covered by the gage surfaces. We can also use parallels or gage blocks for internal features. This is shown in Fig. 5.16. One important consideration in choosing a method for measuring size is the accuracy with which we want our measurements. For low accuracies, a simple vernier or a micrometer would suffice. For accuracies higher than what these instruments can provide, we use high-accuracy dial indicators and gage blocks. Typically, we measure the MMC sizes of features. A set of gage blocks for the MMC size are wringed together. We then set the dial indicator to be zero at the MMC size on the surface of the gage block. The dial indicator is then run over the entire surface. The part is accepted if we do not get any readings above zero. We follow a similar method for LMC sizes.
Fig. 5.16. Measuring size tolerances using screw gauge

Fig. 5.17. Measurement of the width of a slot using parallels
5.4.3 Parallelism Tolerance:

5.4.3.1 Parallelism Tolerance on Surface Elements:

The most common method for inspecting parallelism tolerances on planar surfaces is the use of surface plates and dial indicator. The part is placed on the surface plate such that the datum feature comes in contact with the surface plate [20, 21, 22, 23]. The surface plate should be first inspected for any visible defects and should be clean of oil, dirt, grease etc. The datum plane is then formed by the three high points of the datum feature that comes in contact with the surface plate, and hence simulated by the surface plate. The surface with the parallelism tolerance is then inspected with a dial indicator mounted on a stand. The Full
Indicator movement (FIM) gives the parallelism tolerance used up in the part being inspected. The part is accepted if the FIM is less than the specified tolerance value, else it is rejected. Fig. 5.18 and Fig. 5.19 shows the schematic and the actual measurement of parallelism tolerance respectively.

Fig. 5.19. Measurement of parallelism tolerance using surface plates and dial indicator (see Ref [21])

Fig. 5.20. Dial indicator being used to measure parallelism on a planar surface
5.4.3.2 Parallelism Tolerance on FOS Elements:

For planar FOS, the same method using the dial indicators and the surface plate can be followed. However, we have to take pairs of measurements with two dial indicators simultaneously, one indicator for each of the planar faces. Calculate the difference between the two and divide the difference by two to get the value of ‘tm’ at each point of measurement. The difference between the ‘tm’ max and ‘tm’ min gives the value of the parallelism tolerance used up by the part. Fig. shows the schematic of parallelism tolerance measurement for a slot. A similar setup can be followed for a tab.

\[
\delta_p = \left(\frac{A_{oi} - A_{ui}}{2}\right)_{\text{max}} - \left(\frac{A_{oi} - A_{ui}}{2}\right)_{\text{min}} \leq t_p
\]

Fig. 5.21. Measurement of parallelism tolerance on FOS feature using dial indicator

5.4.4 Perpendicularity Tolerance:

5.4.4.1 Perpendicularity Tolerance on Surface Elements w.r.t Planar Datum:

Perpendicularity tolerances on surface elements can be measured using surface plates, dial indicators and L blocks [20, 21, 22, 23]. Perpendicularity
tolerances can have multiple datums and that necessitates the use of multiple datum simulators. Fig. 5.21 shows one method of inspecting perpendicularity tolerances. The primary and secondary datums can be simulated by two L blocks. Fig. 5.22 shows another method of inspection where, the primary datum is simulated by the L block and the secondary datum is simulated by a gage pin. The primary datum is a plane formed by the three high points of the primary datum feature. The secondary datum is simulated by the pin that is placed in contact with the secondary datum feature and also the surface plate. This produces a two point line contact which is necessary for a secondary datum. The dial indicator on the height stand is set to zero on any point on the inspection surface and the indicator probe is moved all over the surface. The FIM gives the perpendicularity tolerance used up by the surface. The part is accepted if FIM is less than ‘t\text{perp}’ and rejected if FIM is greater than the specified tolerance.

Fig. 5.22. Fixing DRF for perpendicularity tolerance using surface plates and dial indicator (see Ref [22])
Fig. 5.23. Measuring perpendicularity tolerance on a planar surface using surface plates, simulated datums and dial indicator

5.4.4.2 Perpendicularity Tolerance on Surface Elements w.r.t Axis Datum:

Perpendicularity tolerance can also use holes and cylindrical features as datums. In such cases, the axis of the feature is taken as the datum to which the given surface must be perpendicular. Fig. 5.23 shows a method for measuring perpendicularity tolerance w.r.t a hole. A pin gage with the maximum diameter is chosen and inserted into the hole that acts as the datum. The assembly is then mounted on a surface plate with one end of the pin gage completely resting on the surface plate. Thus the pin gage is made perpendicular to the surface plate. The assembly can then be clamped to make it rigid while taking measurements. The surface with the tolerance is then inspected with a dial indicator on a height gage. The FIM gives the perpendicularity tolerance used up by the surface.
5.4.4.3 Perpendicularity Tolerance on FOS:

Perpendicularity on FOS may be measured in a way similar to that of parallelism tolerances. The part is fixture to the surface plate and L blocks such that they simulate the primary and secondary datums [20, 21, 22, 23]. The FOS can then be inspected simultaneously using two dial indicators as we did for parallelism. Fig. 5.24 and Fig. 5.25 shows another method of measuring slots, where gage blocks are fit into them. The gage blocks can then assumed to be the biggest tab that can be fit inside the slot and the sides of the gage blocks can be measured for perpendicularity using dial indicators.
Fig. 5.25. Measuring perpendicularity tolerance on a slot using gages and dial indicator (see Ref [21])
5.4.5 Angularity Tolerance:

5.4.5.1 Angularity Tolerance on Surface Elements:

Inspection of angularity on surfaces is generally done either by using gage blocks and sine bar or by using angle blocks [20, 21, 22, 23]. Fig. 5.26 shows the measurement of angularity tolerance on a surface by using a sine bar and gage blocks. This is one of the most widely used methods for angularity measurement and the combination of sine bars and gage blocks are generally accurate up to ten-thousandths of an inch. The inspection part is placed on the sine bar such that the primary datum comes in contact with the sine bar surface. Then we have to calculate the length of the gage blocks needed to achieve the basic angle dimension specified on the drawing. The inspection surface is thus made parallel to the surface plate and then inspected using a dial indicator. The FIM gives the
angularity tolerance used up by the part. Another method of making the inspection surface parallel to the surface plate is the use of angle blocks. This is shown in Fig. 5.26 In this method, the surface is inspected using a dial indicator after the surface is made parallel to the datum.

5.4.5.2 Angularity Tolerance on FOS Elements:

Angularity tolerances on FOS can be measured in the same way as the parallelism tolerance. Here again, the part is oriented and fixtured using a sine bar or angle blocks such that the FOS is parallel to the surface plate. The FOS can

Fig. 5.27. Measurement of angularity tolerance on a planar surface using sine plate and dial indicator. (see Ref [21])
then be measured using a pair of dial indicators simultaneously and the angularity tolerance can be determined.

**5.4.6 Position Tolerances:**

Fig. 5.27 shows the inspection of the position of a slot. This is done by measuring the size of the two tabs on either side of the slot. Then the total lateral size of the part is also measured. These measurement values give the total width of the slot, and hence the position of the slot with respect to its datum reference frame. This method measures the position with reasonable accuracy for simple parts. For more complex parts which require a very high degree of accuracy, a setup as shown in Fig. 5.28 is used. The setup consists of a height gage with a vernier scale coupled with a dial indicator. The part is held parallel to the surface plate using specialized chucks. The chucks can be rotated and positioned at any angle with accuracy in the range of then-thousandths of an inch. For measurement requirements in the range of chuck accuracy, the measurement readings have to be compensated with the known calibration data. The figure shows the measurement of the position of a slot. A combination of gage blocks can be inserted into the slot for measurement convenience. This is because the side surface of the slot may not be accessible with the dial indicator sometimes. The position of one of the side surfaces from the datum surface is set accurately using the vernier scale that indicates the height. Once the nominal position is set, the dial indicator is then run all over the surface. The FIM of the dial indicator gives the position tolerance used up by the part. This process is repeated for the other
side surface as well. The part is accepted only if the position tolerance requirement is satisfied for both the side surfaces [20, 21, 22, 23].

Fig. 5.28. Measurement of position tolerance of a slot by measuring the size of the slot using screw gauges
Fig. 5.29. Measurement of position tolerance of a slot using special mounting fixtures with holding chucks, height gauge and dial indicators (see Ref [21, 22])

5.4.7 Flatness Tolerance:

Inspection of flatness is relatively simple when compared to orientation or position tolerances. This is due to the fact that we need not be concerned about the datums since form tolerances do not have any. The basic principle of measuring flatness on a surface is to make the surface parallel to the surface plate that we are measuring on. Fig. 5.29 shows the measurement of flatness tolerances on surfaces. The most common method is to use dial indicators to inspect the surface. The dial indicator is first set to zero at some point on the surface on which the flatness needs to be measured. The dial indicator is then run all over the surface and the FIM of the dial indicator is the flatness tolerance used up by the surface.
However, the inspection of flatness tolerance could be iterative. This is due to the fact that there could be imperfections in the surface that is in contact with the surface plate. Any large imperfection in the orientation of the surface can throw off the flatness readings. For this reason, the flatness inspection is done by having the part at different points on the surface plate and checking for repeatability. Another method to circumvent this problem is shown in Fig. 5.33. Here, the surface to be inspected is directly mounted on trammels or measurement pins, thus eliminating any influence of other surfaces. The dial indicator is then run over the surface and the FIM gives the flatness tolerance on the surface. Sometimes, the flatness of a center plane can also be controlled using a straightness tolerance. The measurement of such a tolerance can be done with gages [20, 21, 22, 23].

Fig. 5.30. Measurement of flatness tolerance on a planar surface using inclinometers and dial indicators (see Ref [23])
Fig. 5.31. Measurement of flatness tolerance on a planar FOS using dial indicators

Fig. 5.32. Measurement of flatness tolerance directly using dial indicator

Fig. 5.33. Measurement of the accuracy of trammels using dial indicators
5.4.8 Straightness Tolerance:

Inspection of straightness is very similar to the inspection of flatness. The methods that are used for flatness measurement are good for straightness also. The constraint is that the straightness tolerance defines the tolerance zones in
terms of line elements. Hence, when using dial indicators on surfaces, it must be ensured that the indicator is moved in a straight line parallel to the view plane on which the straightness tolerance is applied. The FIM of the dial indicator gives the straightness tolerance used up by the surface [20, 21, 22, 23]. For measuring straightness of a center plane, the most common method is the use of gages. Also, the measurement can be made simultaneously using multiple dial indicators on special fixtures if gages are not available. These methods are shown in the Fig. 5.35.
Fig. 5.36. Measurement of straightness tolerance on a rectangular FOS using dial indicators (see Ref [21, 23])
5.5 Coordinate Measuring Machines:

New methods using CMMs have automated the process of inspection and they are replacing traditional methods of measurement. Modern CMMs are very powerful metrological tools and are widely used in manufacturing plants across the world. Improvement in the accuracy and precision of the measurements coupled with decrease in cost, time and manpower needed have made the CMMs immensely popular. From making small size measurements to reverse engineering complex synthetic surfaces, CMMs have come a long way in the past couple of decades. However, we do not get the features of interest directly such as planes, lines, circles, cylinders, surfaces etc., but only a cloud of points from the CMM machine. One thing that is common to all these advances in the inspection process is that measurements made on a part are essentially communicated in terms of a cloud of points. The responsibility of the hardware, whether it costs $1000 or $10 million ends with churning out a set of points and a few numbers associated with them. The software that is associated with the CMM machine is usually what that does the bulk of the ‘intelligent’ work that can be used for making inferences about conformance to design specifications. So, the essence of coordinate metrology using CMMs can be summarized in as follows:

- Generate data sets by point to point measurements of work piece using the CMM probe
- Calculate relevant substitute geometry in terms of parameters specifying size, form, orientation and location
• Evaluate the substitute features in light of design specifications and tolerances

A comparison of traditional and CMM metrology as described in [28] is shown below.

<table>
<thead>
<tr>
<th>Conventional Metrology</th>
<th>Coordinate Metrology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alignment is manual and time consuming</td>
<td>No alignment necessary but registration may be necessary</td>
</tr>
<tr>
<td>Adaptation to multiple measuring tasks is difficult</td>
<td>Easy adaptation to multiple tasks made possible by software</td>
</tr>
<tr>
<td>Physical comparison of parts with gage blocks, pins</td>
<td>Comparison made between mathematical models</td>
</tr>
<tr>
<td>Multiple setups and machines required for different tolerance types</td>
<td>Minimum setups and machines for measuring all tolerance types</td>
</tr>
</tbody>
</table>

Table 5.1. Comparison of conventional metrology and coordinate metrology using CMM

Although CMMs give an impression of being all powerful and accurate, there are still many ways that errors could occur. The most common types of errors that are associated with the CMMs are listed below:

• Hardware errors such as problems with the rack and pinion drives, loose fittings of joints in the CMM
• Thermal errors arising in both the parts and probes
• Calibration errors
Measurement uncertainty, which in itself is a big topic investigated by a lot of researchers

CMM software errors arising due to incorrect implementation of algorithms or wrong choice of algorithms for evaluating features

One focus of this thesis is to investigate these errors inherent in the CMM software. The ensuing chapters present a survey of the popular fitting algorithms available for fitting substitute geometry to data points. Then the results of a study, investigating the fitting algorithms used in a couple of commercial CMM software is presented.
CHAPTER 6
FITTING ALGORITHMS IN CMM SOFTWARE

6.1 Computational Metrology:

The objective of metrology is to check the manufactured parts for the conformance of tolerance specifications on the drawings. Fitting substitute geometry is a quintessential step in metrology using CMM’s. It is the process of establishing substitute geometry from real manufactured parts. One does not get the features of interest such as planes, lines, circles, cylinders etc directly from the measurements. Instead, to estimate the quality of the features in question, we need to fit relevant substitute geometry to the point data set and create a model for the feature we are inspecting. One can then check if the feature complies with its tolerances. The results of geometric fitting are not only used to qualify the manufactured parts, but also to make an assessment of the manufacturing processes that produced the parts. Thus, feature fitting algorithms form the core of any CMM software. Feature fitting algorithms are coded in data analysis software which take the inspection data points as input and output the parameters that define the best fit substitute geometry. The challenge in creating these fitting algorithms is how to precisely and accurately approximate the actual geometry of manufactured features. Despite the obvious advantages of such software, computations to convert raw data to useful features are an important source of error in CMMs. The term computational metrology can be defined as the process of fitting and filtering of discrete geometric data [40]. It deals with the study of
effects of data analysis algorithms and computations on the performance of measurement systems. It is believed by many that errors due to feature fitting are negligible and even the ASME standard is of the same opinion. However, this has proved to be wrong time and again and computational metrology has emerged as a separate discipline in the field of design and manufacturing automation.

6.2 Fitting Algorithms:

All fitting problems can be generalized as optimization problems. The aim of any fitting process is to find the parameters of the substitute geometry that optimize a particular fitting objective for a given set of data points. We must know geometric feature what we want to fit from a set of data. This is what is meant by the fitting objective. For a given set of data, we can fit any number of different feature types and the chosen fitting objective must reflect that need and produce a unique substitute feature. This is illustrated in Fig. 6.1 where for the same set of three points, it is shown that we can fit a line, plane or a circle.

Fig. 6.1. Different objective functions result in different substitute features for the same set of data points

Forbes, Anthony et al [43, 46] have described the construction of various feature fitting algorithms and their properties. It is evident from their work that all
fitting problems can essentially be considered as minimization problems. For a given dataset, we can fit different feature types depending on the chosen fitting objective. Almost all the fitting procedures attempt to minimize the error between the data points and the substitute feature. This error is expressed mathematically according to the fitted feature and then popular optimization methods are used to solve the problem. Nassef and ElMaraghy [52] explain a method to find the best objective function that can be used for a particular problem. They explain the fitting process as depicted in Figure 6.2. The most frequently used fitting algorithms are based on the L_p norm, which is discussed in the next section.

![Fig. 6.2. Schematic of fitting substitute features in CMM software (see Ref [53])](image)

6.3 The L_p Norm:

The L_p norm is given below:

\[
\left[ \frac{1}{n} \sum_{i=1}^{n} |d_i|^p \right]^{1/p}
\]
In the above expression, \( p \) varies between zero and infinity, \( n \) is the total number of data points and \( d_i \) is the shortest distance between \( p_i \), the \( i^{th} \) data point and the substitute feature. This is the objective function that needs to be minimized and the parameters that minimize this objective function are the parameters of the substitute feature. For simplicity of computation, the \( L_p \) norm can be written as follows:

\[
\left[ \sum_{i=1}^{n} |d_i|^p \right]
\]

Based on the value of \( p \) in the above expression and hence the objective function optimized, the fitting problems are classified into two broad categories. They are the Least squares fitting and the Chebyshev fitting. These are the most popular methods of substitute feature fitting. The value of ‘\( p \)’ in the \( L_p \) norm results in different objective functions and hence different types of fitting problems, each resulting in a different substitute feature.

**6.3.1 Least Sum of Distances Fitting:**

The most simple type of \( L_p \) norm problem is when \( p = 1 \). The objective of this fitting is to minimize the sum of absolute distances. This type of fitting problem is also called the median polish fit. The result of this type of fitting passes through the median of the distribution of the \( d_i \)’s [41]. This type of fitting is less sensitive to the data outliers than other fitting methods. However, this type of fitting is not of much importance for our context. The more important siblings
of these algorithms, namely the least squares and Chebyshev fitting are described in detail in the next sections.

6.3.2 Least Squares Fitting:

The least squares method which results from the $L_p$ norm when $p = 2$, is by far the most widely used method in CMM fitting software because of its simplicity and robustness. The result of the least squares fitting passes through the sample mean of a normal distribution of the distances $d_i$’s [41, 43]. The least squares fitting is more sensitive to outliers than the least sum of distances fitting and requires complex filtering techniques to remove outliers. NIST has attempted to standardize these least squares algorithms for different feature types, which is explained by Shakarji [42, 45]. An exhaustive description and analysis of least squares fitting techniques for a variety of feature types is given by Forbes [43].

The line and plane fitting algorithms result in linear optimization problems and they reduce to simple eigen vector problems which can be solved efficiently by Singular Value Decomposition technique [42]. All other non-linear fitting problems have to be solved by special optimization algorithms suited for non-linear problems. A review of least squares problem formulation as elaborated in [42, 43] for some of the common feature types are described below:

6.3.2.1 Least Squares Line Fitting:

We specify the line by

i. A point $(x_0, y_0, z_0)$ on the line and
ii. The direction cosines $(a, b, c) = a$ of the line with respect to the coordinate system of measurement.

We can then see that any point $(x, y, z)$ on the line satisfies the equation

$$(x, y, z) = (x_0, y_0, z_0) + t(a, b, c),$$

for some value of ‘$t$’. Now, $d_i$, the perpendicular distance of the line from any data point $(x_i, y_i, z_i)$ can be expressed as

$$d_i = \sqrt{[u_i^2 + v_i^2 + w_i^2]}$$

Where, $u_i = c(y_i - y_0) - b(z_i - z_0)$, $v_i = a(z_i - z_0) - c(x_i - x_0)$, $w_i = b(x_i - x_0) - a(y_i - y_0)$

The objective function is $J(x; a) = \sum |a X (x_i - x)|^2$, where $X$ denoted the vector cross product. Since the least squares line, $L$, passes through the centroid of the data, $(\bar{x}, \bar{y}, \bar{z})$ this specifies a point on $L$. The direction of the line $L$ can be found by solving the minimization problem, subject to the constraint $|a| = 1$. The solution of the problem, i.e., the values of $(a, b, c)$ are the solution of the Eigen value problem $(M^T M) a = \lambda a$, referred to as the normal equations. This can then be solved using the Singular Value Decomposition (SVD) method. The singular vector $(a, b, c)$ corresponding to the largest singular value is the solution for the direction cosines. Thus the best fit least squares line $L = (\bar{x}, \bar{y}, \bar{z})$, $(a, b, c)$.

6.3.2.2 Least Squares Plane Fitting:

We specify the plane by

- A point $(x_0, y_0, z_0)$ on the plane
- The direction cosines $(a, b, c) = a$ of the plane with respect to the coordinate system of measurement.
We can then see that any point \((x, y, z)\) on the plane satisfies the equation 
\[(x, y, z) = (x_0, y_0, z_0) + t(a, b, c),\] for some value of ‘\(t\)’. Now, \(d_i\), the perpendicular distance of the plane from any data point \((x_i, y_i, z_i)\) can be expressed as
\[d_i = a(x_i - x_0) + b(y_i - y_0) + c(z_i - z_0)\]

The objective function is 
\[J(x, a) = \sum |a \cdot (x_i - x)|^2,\] where \(X\) denoted the vector cross product. Since the least squares plane, \(P\), passes through the centroid of the data, \((\bar{x}, \bar{y}, \bar{z})\) this specifies a point on \(P\). The direction of the plane \(P\) can be found by solving the minimization problem, subject to the constraint \(|a| = 1\). The solution of the problem, i.e., the values of \((a, b, c)\) are the solution of the Eigen value problem \((M^T M)a = \lambda a\), referred to as the normal equations. This can again be solved using the Singular Value Decomposition (SVD) method. The singular vector \((a, b, c)\) corresponding to the smallest singular value is the solution for the direction cosines. Thus the best fit least squares plane \(L = (\bar{x}, \bar{y}, \bar{z})\), \((a, b, c)\).

### 6.3.2.3 Least Squares Sphere Fitting:

We specify the sphere by

i. \(x - (x_0, y_0, z_0)\) the center of the sphere

ii. \(r\) – the radius of the sphere

Now, \(d_i\), the perpendicular distance of the sphere from any data point \((x_i, y_i, z_i)\) can be expressed as
\[d_i = \sqrt{[(x_i - x_o)^2 + (y_i - y_o)^2 + (z_i - z_o)^2]} - r = |x_i - x| - r\]

The objective function for the minimization problem is then
\[ I(x, r) = \sum (|x_i - x| - r)^2 \]

The Levenberg – Marquardt algorithm is a popular technique used to solve these types of non-linear optimization problems. This technique sometimes requires the partial derivatives of the objective or the distance function. For solving the least squares spheres problem, we need to furnish the partial derivatives of the distance function with respect to the unknown variables.

Forbes [43] explains the method of using the Gauss – Newton optimization algorithm to solve the least squares spheres. He also explains the linear least squares spheres approximation which can be used as a good initial estimate for the non-linear problem.

6.3.2.4 Least Squares Two Dimensional Circle Fitting:

We specify the two-dimensional circle by:

i. \( x - (x_0, y_0) \) the center of the circle

ii. \( r \) the radius of the circle

Now, \( d_i \), the perpendicular distance of the circle from any data point \((x_i, y_i)\) can be expressed as

\[ d_i = \sqrt{[(x_i - x_0)^2 + (y_i - y_0)^2]} - r = (|x_i - x| - r) \]

The objective function for the minimization problem is then

\[ I(x, r) = \sum (|x_i - x| - r)^2 \]

Similar to solving the least squares spheres problem, we need to furnish the partial derivatives of the distance function with respect to the unknown variables for the two-dimensional circle problem as well when using the

6.3.2.5 Least Squares Three Dimensional Circle Fitting:

We specify the two-dimensional circle in a 3-D coordinate frame by

i. \( x - (x_0, y_0, z_0) \) the center of the circle.

ii. \( A \) – direction numbers of the normal to circle’s plane. The direction cosines of the normal to the circle’s plane are used in the place of the direction numbers \( A \)

iii. \( R \) – radius of the circle.

Now, \( d_i \), the perpendicular distance from data point \((x_i, y_i, z_i)\) can be expressed as

\[
d_i = \sqrt{[(g_i)^2 + (f_i - r)^2]}
\]

The function, \( g(x_0, x, A) \) denotes the distance from the point \( x_i \) to the plane defined by the point \( x \) and the normal to the direction \( a = A/|A| \)

\[
g_i = g(x_0, x, A) = a \cdot (x_i - x) = a(x_i - x) + b(y_i - y) + c(z_i - z)
\]

The function, \( f(x_0, x, A) \) denotes the distance from the point \( x_i \) to the line defined by the point \( x \) and the direction \( a = A/|A| \)

\[
f_i = f(x_0, x, A) = a \cdot (x_i - x) = \sqrt{u_i^2 + v_i^2 + w_i^2}
\]

where, \( u = c(y_i - y) - b(z_i - z), v = a(z_i - z) - c(x_i - x), v = b(x_i - x) - a(y_i - y) \)

The objective function for the minimization problem is then
\[ I(x, A, r) = \sum (y^2 + (f_i - r)^2) \]

Similar to solving the least squares sphere problem, we need to furnish the partial derivatives of the distance function with respect to the unknown variables for the two-dimensional circle problem as well when using the Levenberg–Marquardt optimization algorithm.

Shakarji [42] describes the 3-D circle fitting as a multi-step process as described below:

i. Compute the least squares plane of the data.

ii. Rotate the data such that the least squares plane is the x-y plane.

iii. Compute the 2-D circle fit in the x-y plane.

iv. Rotate back to the original orientation.

v. Perform a full 3-D minimization search over all the parameters.

The details of the Gauss–Newton algorithm are explained by Forbes. Additionally, Chernov and Lesort [57] explain the usage of two other optimization algorithms which are used especially for least squares circles fitting, namely

a. Landau algorithm

b. Spath algorithm

6.3.2.6 Least Squares Cylinder Fitting:

We specify the cylinder by

i. \( x = (x_0, y_0, z_0) \) a point on the cylinder axis.
ii. $A$ – direction numbers of the cylinder’s axis. The direction cosines of the normal to the circle’s plane are used in the place of the direction numbers $A$

iii. $R$ – radius of the cylinder.

Now, $d_i$, the distance from data point $(x_i, y_i, z_i)$ can be expressed as $d(x_i) = f_i - r$

The objective function for the minimization problem is then

$$J(x, A, r) = \sum (f_i - r)^2$$

Similar to solving the least squares spheres problem, we need to furnish the partial derivatives of the distance function with respect to the unknown variables for the two-dimensional circle problem as well when using the Levenberg – Marquardt optimization algorithm.

The least squares technique is a well researched topic and always yields an unique solution. These are the strengths of this method that makes it easy to use and implement. Even the methods for solving non-linear least squares problems are computationally stable and robust. This is probably the main reason for its widespread use in CMM software.

6.3.3 Chebyshev Fitting:

The second class of problems, the Chebyshev fitting methods also called minimax fitting, arise when $p$ approaches infinity. Anthony et al [45, 46] describes the complete family of Chebyshev fitting problems along with the optimization algorithms to solve them. The same paper also describes the maximum inscribed and minimum circumscribed algorithms, which are a subset
of the Chebyshev problems. A brief survey of the Chebyshev fitting problems from these references is described here.

6.3.3.1 Two-Sided Fits:

Minimax fitting minimizes the maximum distance between all the sampled data points. If the substitute geometric element is described using the vector of parameters $u$ and $d_i(u)$ denotes the distance from the $i^{th}$ data point to the element defined by $u$, the optimization problem is formulated as follows:

Objective function – Min ($\text{Max } |d_i(u)|$)

The above objective function can be written as the following constrained minimization problem

Minimize ($s$), subject to the constraint $s - |d_i(u)| \geq 0$.

This Chebyshev Minimum Zone (MZ) fitting procedure fits two perfect features of similar type very close together such that the distance between them is minimized and all the measured data points fall between the two perfect features. For lines and planes, the two perfect features must be parallel and for circles, cylinders and spheres, the two features must remain concentric. For these problems, the distance function in equation 1 is positive or negative depending on which side of the element the $i^{th}$ data point lies. For these problems, the parameters $(u, s)$ define a zone element and hence the problem is called the minimum zone problem. The zone element $(u, s)$ consists of the two geometric elements composed of points at distances $-s$ and $s$ from the geometric element $u$. The distance $2s$ between the two surfaces of the zone element is often called the
width of the zone. The solutions for some of the popular MZ fitting problems for particular feature type are listed below.

**MZ lines in 2D** – 3 points with 1 on one line projecting orthogonally into the interior of the line segment defined by the other 2 points.

**MZ planes** – This can have two solutions. The first one is 4 points with 1 on one plane projecting orthogonally into the interior of the triangular plane segment formed by the other 3 points. The second solution is 2 points on each plane defining line segments that intersect internally when projected orthogonally onto either plane.

**MZ circles** – This can have two solutions. The first one is four points, two on each circle which when radially projected onto a common concentric circle, define chords that intersect internally.

**MZ spheres** – 5 points, 3 on one sphere and 2 on the other sphere which, when radially projected onto a common concentric sphere, define respectively the vertices of a triangle and the ends of a chord that intersect internally.

### 6.3.3.2 One-Sided Fits:

Sometimes, it is required that the material on the part lie completely on one side of the substitute feature. This is achieved by one-sided fits, which are again a variation of the two-sided fits. This fitting is done by assigning signed errors and doing an optimization on either the positive or negative side of the substitute feature depending on the requirement. The mathematical formulation of this one-sided fitting is explained as follows.
Min (Max $|d_i(u)|$), subject to the constraint, $d_i(u) \leq 0$ or Min (Max $|d_i(u)|$), subject to the constraint, $d_i(u) \geq 0$.

Subclasses of these one-sided fitting problems are the Minimum Inscribed (MI) and Maximum Circumscribed (MC) fits. These are commonly used to evaluate circular and cylindrical features. The solutions of some common MI and MC problems are listed below.

**MC Circle** – This can have two solutions. The first one is 2 points defining the diameter of the circle. The second one is 3 points at the vertices of a triangle containing the circle center (an acute angled triangle).

**MC Sphere** – This can have two solutions. The first one is 2 points defining a diameter, 3 points at the vertices of a triangle that contains the center. The second solution is 4 points at the vertices of a tetrahedron that contains the center.

**MC Cylinder** – This is nothing but the MZ lines problem in 3D. Here the distance function can take only one sign and the parameters $(u, s)$ define a cylinder of radius $s$, whose axis is the line with parameters $u$.

**MI Circle** – 3 points at the vertices of triangle for which the circle center lies within its interior.

**MI Sphere** – 4 points at the vertices of a tetrahedron for which the sphere center lies within its interior.

**MI Cylinder** – The MI cylinder problem is similar to the MC cylinder problem, with just the appropriate signs in equation 1 reversed.
We must note that the distance functions of the MC and the MI fitting problems are measured from the core \( u \), rather than from the geometric form of the fitted feature.

Anthony et al. [46] describes in detail the formulation, algorithms and solutions of these Chebyshev type fitting problems. Gass and Witzgall [58] explain an algorithm for minimax circle fitting called the approximate model of minimax fitting. Lai and Wang and Etesami and Qaio [60] have proposed algorithms using voronoi diagrams for solving Chebyshev type minimax fitting problems. Murthy and Abdin [59] describe a survey of minimum zone fitting algorithms and optimization techniques in detail. To get into the internals of these algorithms is beyond the scope of this thesis and the reader is referred to these references for a detailed explanation of these algorithms. Shakarji [45] summarizes the Chebyshev fitting problems in the following table. The functions \( g_i \) and \( f_i \) mentioned in the table hold the same meaning as explained in the least squares problems.
Table 6.1. Table summarizing the Chebyshev one-sided and two-sided fits (see Ref [45])

| 6.4 Fitting Algorithms and Tolerance Types: |

| 6.4.1 Establishing Datums: |

| 6.4.1.1 Planar Datums not Subject to Size Variations: |

The Standard defines that for planar datum features, datums are established by contacting one or more high points depending on the hierarchy of the particular datum feature. The primary datum is established by the perfect plane that contacts the primary datum feature at a minimum of three points. The second datum plane is required to be perpendicular to the primary datum and should contact the secondary datum feature at a minimum of two high points. The tertiary datum plane is required to be perpendicular to the primary and secondary datums and should contact the tertiary datum feature at least at one high point.
6.4.1.2 Planar Datums Subject to Size Variations (FOS Datums):

At RFS: Datums from planar features of size such as mid-planes are established by physical contact between feature surface and surface of the processing equipment. The primary datum plane is the mid-plane of the true geometric counterpart (TGC) of the datum feature. The TGC of an external FOS is a pair of parallel planes at minimum separation. The TGC for an internal FOS is a pair of planes at maximum separation. The secondary datum plane is established by the same method as the primary with an additional constraint that the secondary plane should be perpendicular to the primary. Similarly, the tertiary plane should be perpendicular to the primary and the secondary datums.

The first usual step in establishing planar datums from FOS is to fit a plane that contacts the high points of the first side of the FOS. Then a plane parallel to the above plane is fit to the second side of the FOS to get the first pair of planes. Then a second pair of parallel planes is obtained by fitting a least squares plane to the second side and then fitting a parallel plane to the first side of the FOS. The pair of planes with the minimum separation distance is the TGC for an external FOS and similarly, the pair with the maximum separation is the TGC for an internal FOS. The next step is to determine the mid-plane of the TGC which is the datum associated with the FOS. This is shown in Fig. 6.3. Zhang and Roy [64] explain in detail another method using convex hulls to determine these datums.
At MMC and LMC: The TGC of a planar FOS at MMC is the MMC size limits for the primary datum and the Virtual Condition (VC) limits for the secondary and tertiary datums. This gives a pair of planes separated by the respective distances for both internal and external features. The actual datum is the mid-

Choose the greater of $d_1$ or $d_2$
plane of these two planes. For FOS that is specified at LMC, the primary datums are the center planes at the LMC boundary condition. For secondary and tertiary datums, the TGC are the pair of planes separated by the VC limits.

6.4.1.3 Cylindrical Datums Subject to Size Variation:

At RFS: The primary datum is the axis of the TGC of the datum feature. The TGC of an external FOS is the smallest circumscribed feature and the TGC of an internal FOS is the largest inscribed feature. The secondary datums are established in the same way as the primary method with the constraint that they should be perpendicular to the primary datums. Similarly, the tertiary datums determined by this method should be perpendicular to the primary and the secondary datums.

At MMC and LMC: The establishment of non-planar datums from FOS follows the same pattern as the planar ones when they are specified at LMC or MMC.

6.4.2 Fitting Datum Features:

One of the common methods to obtain a planar datum from a planar datum feature is to fit a least squares plane to the measured points. We know that a least squares plane passes through the centroid of the sample points, hence the requirement of a minimum of three, two and one point contact for a primary, secondary and tertiary datums respectively is met since the number of sampled points is usually large.
6.4.2.1 Deficiencies in Current Datum Fitting Methods:

Some of the disadvantages of using the least squares method are listed as follows. The standard specifies that the datum planes act as the origin from which measurements are taken. However, least squares planes are more of ‘medial’ planes and there is always an error associated with making measurements from the least squares planes. Also, the datums are supposed to be the actual mating envelopes of the datum feature. However, least squares planes often result in planes that are tilted and produce wobbles or result in a ‘medial’ plane as described above. Minor variations of the least squares method are followed to overcome these deficiencies. These include assigning weights to the data points, performing geometric operations such as translation, rotation on the least squares plane. For example, the least squares primary datum plane can be translated such that it contacts three high points or a secondary datum plane can be rotated through a small angle to make it perpendicular to the other datum planes. In spite of its deficiencies, the least squares plane fitting is universally used in most of the CMM fitting software.

6.4.2.2 Wobbling or Unstable Datum Features:

In case of any irregularities on the surface of the primary or secondary datum feature that causes the part to be unstable or that makes the part to wobble, the Standard advocates that the datums can be established by adjusting the part to an optimum position that prevents any wobble. Unfortunately, the above definition of datums in the ASME standard carries a lot of ambiguity. There is no
definition of what is considered to be a wobbling or unstable part and also what an optimum position means, the interpretations of which are left to the user. It is also accepted that rocking datum features will generate more than one possible datum. These are called the ‘candidate datums’. Some methods consider that if the controlled feature complies with its geometric tolerance to at least one of the candidate datum reference frames, the feature can be accepted to pass. In some other methods, the controlled feature has to pass with respect to all candidate datum reference frames. Some other methods are advocated classification of the candidate datum sets to find the most appropriate datum. One such method divides the datum primary datums into thirds. Any candidate plane that has all its high points in the outer thirds of the surface are discarded as rocking datums [49, 51].

The issues of wobble and rocking datums are not formalized by any authority and are still an active area of research. The complexities involved in handling datums has resulted in least squares methods being followed in the CMM fitting software due to the manufacturer’s convenience. But, this is neither a correct solution nor does it conform to the requirements of the standard.

6.4.3 Fitting Algorithms for Individual Tolerance Classes:

This section gives a list of the different algorithms used to substitute features for evaluating specific tolerance classes. The purpose of this section is to give a general awareness that different fitting algorithms can be used to evaluate a particular tolerance type and different tolerance types could be evaluated by the
same fitting algorithm. Of course, the accuracy of the results and hence the integrity of the procedure differs in each case.

The most popular fitting method used is the least squares technique. This can be used and is still used in a large number of CMM software for inspecting almost all tolerance types. However, the results produced by these fits are error prone and may not necessarily simulate the inspected feature or tolerance type accurately. This is because, these algorithms are simple to implement and provide an average substitute feature. They do not exactly reflect the principles behind the tolerancing standards. Anthony, Cox et al [46] state that in general practice, least squares algorithms are used when measurement errors predominate. Minimum zone algorithms are used when measurement errors are negligible and form tolerances are predominant. The MC and MI fitting methods are used when assembly of parts is the most important criterion. The following sections describe the fitting algorithms associated with individual tolerance and feature types.

6.4.3.1 Size Tolerances:

Least squares fitting or Chebyshev one-sided fitting methods depending upon the accuracy required and computational capability. Chebyshev algorithms are accurate and are computationally intensive. This is the case for all feature types that have a general size tolerance.

6.4.3.2 Orientation Tolerances:

Planar surfaces – The most popularly used fitting type for planar surfaces is the one-sided plane fitting methods.
Mid-plane Elements – Least squares method can be used to find set of parallel planes. Alternatively, constrained Chebyshev one-sided plane fitting can be used to fit the planar faces and then the mid plane can be subsequently derived from them.

Cylindrical FOS – MC and MI cylinder fitting are the most popularly used fitting types for external and internal FOS respectively. Alternatively, least squares method can also be used although it is less accurate.

6.4.3.3 Form Tolerances:

Form tolerances are represented in the tolerancing standards are tolerance zones without any datum constraints. This makes them a floating zone inside the bigger size tolerance zones. For this same reason, minimum zone Chebyshev fitting algorithms are the most commonly used algorithms to evaluate all types of form tolerances. This is depicted in Fig. 6.4.

![Chebyshev Two Sided Fit](image)

Fig. 6.4. Chebyshev Two-Sided fit for Form Tolerance

6.4.3.4 Position Tolerances:

Cylindrical FOS – MC and MI cylinder fitting techniques are used for external and internal FOS respectively. The axis of the fitted cylinder can then be
compared with the actual tolerance zone to see if there is conformance to the
tolerance specifications or not. Alternatively, least squares method can also be
used although it is less accurate.

Mid-plane elements – Least squares method can be used to find set of parallel
planes. Alternatively, constrained Chebyshev one-sided plane fitting can be used
to fit to the planar faces and then the mid plane can be subsequently derived from
them. The derived mid-plane can be used to determine tolerance conformance.

6.4.3.5 Run-out Tolerances:

Circular Run out – MC and MI circle fitting are the most commonly used methods
to fit substitute features for the measurement of run out tolerances. This step may
be used to evaluate the eccentricity of the circular elements. Minimum zone
fitting algorithms are used to evaluate the actual circular run out values.

Total Run out – MC and MI cylinder fitting are the most commonly used methods
for measurement of total run out tolerances. This step may be used to evaluate the
eccentricity and tilt of the circular elements. Minimum zone fitting algorithms are
used to evaluate the actual total run out values.
Fig. 6.5. Maximum Inscribed (MI) Circle Fitting

Fig. 6.6. Minimum Circumscribed (MC) Circle Fitting
6.5 GIDEP and NIST Alert on Errors in CMM Inspection Algorithms:

Typically, the accuracy of the software is taken to be granted and much of effort is spent in preventing and rectifying other sources of errors in metrology. However, in the early 90’s, Government-Industry Data Exchange Program and National Institute of Standards and Technology (NIST) made pioneering efforts to identify and correct the errors caused due to uncertainty in software. Shakarji [44] classifies the errors due to software uncertainty into two major categories:

- **Implementation issues** – This occurs due to the incorrect choice of solution or implementation details and trying to solve the problem without having a complete understanding of the mathematics behind the algorithms

- **Software Development Process** – This occurs due to inherent problems that come with developing, writing and testing software

Such a situation arose because the companies that produced CMM software had different implementations of varying levels of accuracy for these fitting algorithms. NIST has made a series of efforts to standardize the algorithms that are used in CMM metrology. The motivation for these efforts was that different software that claimed to use the same fitting algorithm produced different results. For example, even though a least squares technique should produce unique results, different implementations can give different results due to a wrong interpretation of the mathematics behind the algorithm. This promoted efforts from NIST to standardize the implementations of the different software.
NIST came out with a series of data sets, reference algorithms and reference results for all types of fitting algorithms such as least squares and Chebyshev. For a detailed explanation of these topics, the reader is referred to [41 – 45].

However, as one could see, these efforts were more concerned with the problems that come with the implementation of these algorithms. The importance of the choice of a particular fitting type for evaluating a particular class of tolerance has been pushed to back seat. This is one of the reasons for erroneous results that are persistent with CMM software even after these standardization attempts by NIST. Unfortunately, the developers of CMM software do not generally reveal the algorithms they use and their rationale behind the choice of a particular fitting algorithm. Usually the end users of the CMM machine and software are inspection technicians and machinists who do not understand anything about the algorithms that work behind their software. They end up believing what the software tells them irrespective of whether it is right or wrong.

6.6 Case Study on Commercial CMM Inspection Software:

Given the uncertainty in results, it became a necessity to evaluate the performance of industrial software with respect to their choice of a particular fitting algorithm type. The core intent of this thesis is thus to investigate the fitting algorithms used in CMM software, check if they accurately simulate the particular tolerance type, compare the results with the principles laid down by the standard and traditional methods of inspection and finally to standardize the
choice of fitting type for every tolerance class. The cases have been restricted to planar feature types and the tolerance classes associated with them because these are most common and basic ones to investigate. If a study could identify errors in CMM software analyzing these simple features, it is hard to imagine the software doing a better job for complex non-planar features.

A case study was done on two commercial CMM software, which we can call Software A and B, to identify and compare the fitting algorithms they use to represent different tolerance classes. A part with planar features was manufactured with intentional variations added that were known before the inspection. The part that was used in the case study is shown in Fig. 6.3. The curved features of the part, namely the feature with the single and double curvatures are ideally planes. Let us call these features as single curved plane and double curved plane respectively. The part was manufactured with a CNC machine and then first inspected using a CMM arm that was connected to the software A. The points from the CMM were analyzed separately using the standardized algorithms discussed in the previous sections. These standardized

Fig. 6.7. Isometric views of the part used in the case study
algorithms were then used to fit substitute features and the parameters of the substitute feature according to each fitting type were computed. The part was also inspected using traditional measuring methods. The results of the Software A were then compared to the verification done using normative guidelines proposed in the following chapters. A similar procedure was also performed using data collected from a Gantry CMM connected to Software B. The results of the test for each tolerance type are summarized as follows.

6.6.1 Size Tolerance:

6.6.1.1 Software A:

The width or the actual mating envelope of the part between the single curved plane and the side datum plane was measured to be 3.370 and software A specified the width as 3.262. Least squares planes were fitted to the two sets of data. The average distance between these two planes was computed to be 3.262. This corresponds exactly to the value specified by software A. Thus software A computes the size dimensions also using a least squares based method. A two-sided fit was performed on the combined sets of points of both the planes. The width of this two-sided fit was computed to be 3.3694. This corresponds exactly to the size measured by traditional methods. Thus the value reported by the software A has an error of nearly 4%.

6.6.1.2 Software B:

The width of the part between the single curved plane and the side datum plane was again measured with the second CMM and software B specified the
width as 3.285. Independently, least squares planes were fitted to the two sets of data. The average distance between these two least squares planes was computed to be 3.285. This corresponds exactly to the value specified by software B. Thus software B also computes the size dimensions also using a different least squares based method. A two-sided fit was performed on the combined set of points of both the planes. The width of this two-sided fit was computed to be 3.372. This corresponds exactly to the size measured by traditional methods. Thus the value reported by the software B has an error of nearly 4%.

6.6.2 Perpendicularity Tolerance:

6.6.2.1 Software A

The perpendicularity of the double curved plane with respect to the bottom datum plane as the primary datum and the side datum plane as the secondary datum was measured. The traditional inspection methods gave a perpendicularity variation of 0.1740 and software A specified the perpendicularity tolerance as 0.0679. A least squares plane was fitted to the feature, the double curved plane. A least squares plane was computed for the primary datum and the three highest points on one side were determined. A plane was fitted to these three high points and this plane is the primary datum plane. Similarly, a least squares plane was fit to the secondary datum and the two high points were determined. A plane perpendicular to the primary datum plane and passing through these two high points was determined. This is the secondary datum plane. The least squares plane of the feature and a sample set of points on the feature plane were determined.
Now, a two sided fit was done on this sampled set of points. The two sided fit was constrained to be perpendicular to both the primary and secondary datum planes. The width of this two-sided fit was computed to be 0.0713. This compares to the value specified by software A. Then, a two sided fit perpendicular to the primary and secondary datums was done on the actual set of measured points. The width of this two-sided fit was computed to be 0.1730. This compares accurately to the perpendicularity tolerance measured by traditional methods. Thus, the value of perpendicularity tolerance reported by software A differs by nearly 65%.

6.6.2.2 Software B:

The perpendicularity of the double curved plane with respect to the bottom datum plane as the primary datum and the side datum plane as the secondary datum was measured using the second CMM with software B. The traditional inspection methods gave a perpendicularity tolerance of 0.1740 and software B specified the perpendicularity tolerance as 0.1938. A least squares plane was computed for the primary datum. A two sided-fit, perpendicular to the primary datum was fit to the data points. The width of this fit was computed to be 0.195. This compares favorably to the perpendicularity predicted by software B. However, software B did not have the capability for adding a secondary datum to the DRF for measuring perpendicularity with two datums. Now, a two sided fit was done on this sampled set of points for the double curved plane. The two sided fit was constrained to be perpendicular to the primary and secondary datum planes. The datum planes were now constructed using the high points on the
datum features and not from least squares. The width of this two-sided fit was computed to be 0.1730. This compares accurately to the perpendicularly tolerance measured by traditional methods. Thus, the value of perpendicularly tolerance reported by software B differs by nearly 15%.

6.6.3 Parallelism Tolerance:

6.6.3.1 Software A:

The parallelism of the single curved plane to the opposite plane was analyzed. The manual methods and the normative guidelines gave a value of 0.215 and software A specified the parallelism as 0.0251. Similar to the perpendicularly tolerance, a least squares plane was fit to the side datum plane and the three high points were determined. A plane was fit to these high points and this is the datum plane. A least squares plane was then fit to the single curved plane. The least squares plane was then sampled to get a set of data points and a two-sided fit was done on these sampled points. The two-sided fit was constrained such that it is parallel to the datum plane. The width of this two-sided fit was computed to be 0.0258. This compares to the parallelism tolerance specified by software A. Then, a two sided fit parallel to the datum was done on the actual set of measured points. The width of this two-sided fit was computed to be 0.2146. This compares accurately to the perpendicularly tolerance measured by traditional methods. Thus, the value of parallelism tolerance reported by software A differs by nearly 88%.
6.6.3.1 Software B:

The parallelism of the single curved plane to the side curve plane was analyzed with the second CMM with software B. The manual methods and the normative guidelines gave a value of 0.215 and software B specified the parallelism as 0.1894. Similar to the perpendicularity tolerance, a least squares plane was fit to the side datum plane. The perpendicular distance of each of these points was computed from the least squares datum plane and the maximum and minimum distances are determined. The difference of these distances was computed to be 0.191. This compares to the parallelism tolerance specified by software B. Then, a new datum was fit with three high points of the datum feature. This was followed by a two sided fit parallel to the new datum on the actual set of measured points. The width of this two-sided fit was computed to be 0.2152. This compares accurately to the parallelism tolerance measured by traditional methods. Thus, the value of parallelism tolerance reported by software A differs by nearly 12%.

6.6.4 Flatness Tolerance:

6.6.4.1 Double Curved Plane:

The flatness tolerance measured by traditional inspection methods was 0.143. The flatness of the surface specified by software A was 0.1653. A least squares plane was fit to the data points. The parameters of this LS plane are [0.0402 -0.9992 -0.0001 -27.1342]. The perpendicular distance of each measured point from this LS plane was computed and the farthest point on either side of the
plane was determined. The farthest points were at a distance of 0.0768 and 0.0886 on either side of the LS plane. The sum of these two distances is 0.1654, which is nothing but the flatness specified by software A. Thus, software A computes the flatness by using the least squares fitting method. A two sided fit was performed manually using the reference algorithms to the measured data points. The width of the two-sided fit was 0.1426. This corresponds exactly to the flatness measured by traditional inspection methods. The result stated by software A is different by nearly 16%.

A similar analysis with software B predicted a flatness tolerance of 0.1418. A two sided fit was performed manually using the reference algorithms to the measured data points. The width of the two-sided fit was 0.1421. This compares favorably with the actual flatness tolerance on the feature. So software B uses a two-sided fit and predicts the flatness correctly.

6.6.4.2 Single Curved Plane:

A similar analysis of the single curved plane was done. The manual inspection gave a flatness of 0.211 and software A specified a flatness of 0.2240. The least squares fit method as described before gave the parameters of the plane as [0.9997 0.0210 -0.0151 11.4328], the farthest distances as 0.1129 and 0.1110 and hence the flatness as 0.2239. This is the same as specified by software A. A two sided fit done on the set of points gave the width of the two-sided zone as 0.2114. This also corresponds exactly to the flatness measured by traditional methods. The result stated by software A differs by nearly 6%. A similar test for
the second CMM with software B was done. A two sided fit was made for the
points collected from the second CMM. The value of the distance between the
planes of the two sided fit was computed to be 0.2128. This compares favorably
with the flatness value predicted by the software B, which is 0.2164. So software
B uses a two-sided fit and predicts the flatness correctly.

The results of the test are summarized in the following tables.

**Flatness Tolerance:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Actual</th>
<th>Software A</th>
<th>Error</th>
<th>Software B</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Curved</td>
<td>0.2114</td>
<td>0.2339</td>
<td>10%</td>
<td>0.2164</td>
<td>2%</td>
</tr>
<tr>
<td>Double Curved</td>
<td>0.1426</td>
<td>0.1694</td>
<td>18%</td>
<td>0.1418</td>
<td>4%</td>
</tr>
</tbody>
</table>

Table 6.2. Flatness tolerance results of the case study

**Perpendicularity Tolerance:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Actual</th>
<th>Software A</th>
<th>Error</th>
<th>Software B</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Double Curved</td>
<td>0.1730</td>
<td>0.0679</td>
<td>60%</td>
<td>0.1937</td>
<td>12%</td>
</tr>
</tbody>
</table>

Table 6.3. Perpendicularity tolerance results of the case study

**Parallelism Tolerance:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Actual</th>
<th>Software A</th>
<th>Error</th>
<th>Software B</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Curved</td>
<td>0.2146</td>
<td>0.0251</td>
<td>86%</td>
<td>0.1894</td>
<td>12%</td>
</tr>
</tbody>
</table>

Table 6.4. Parallelism tolerance results of the case study
6.7 Reconciliation of Fitting Algorithms w.r.t Standards and Manual Inspection:

In the previous section we saw that the values reported by the CMM software differ to a great extent from the actual values. It is evident that the difference is due to a wrong choice of fitting algorithm. The values reported are so bad, for example, the perpendicularity tolerance reported by the software is lesser than the flatness tolerance for the same surface. This is because the software tries to work with least squares planes that are easy to compute rather than doing a two-sided fit which corroborates with the definition of the standard. The errors are so huge that it requires serious thoughts and efforts to standardize the choice of fitting types that agrees best with the definition of the standard.

Least squares analysis is a very useful tool in coordinate metrology due to the fact that it is a well researched topic and computationally less intensive. The problem arises from the fact that neither the ASME Y14.5 nor the ISO 1101 standards on dimensioning and tolerancing specify the use of least squares fit. The Chebyshev technique, fits substitute features which are more of ‘extremal’ in nature compared to the ‘medial’ planes of the least squares technique. The advantage with these types of fits is that they are more in concordance with the principles and language of the standards. It is obvious that the choice of an optimization algorithm for a fitting problem depends on the type of problem in hand. Optimization algorithms such as Levenberg – Marquardt, Gauss – Newton algorithms, which have matured and have robust implementations in the literature,
work great with least squares problems, but they are not good for Chebyshev type problems. These Chebyshev objective functions often have their absolute minima among local minima which make it harder computationally. Also, Chebyshev objective functions sometimes have objective functions which are non-differentiable, and the optimization algorithm used to solve these should be able to overcome these difficulties. This makes these algorithms computationally intensive and sometimes they do not even converge to the right solution. CMM software developers have traditionally favored least squares based techniques for inspecting the GD&T tolerances because they are easy to implement. In other words, mathematical convenience takes precedence over accuracy and there is a lack of adherence to the principles set forth by the standard.

The measurement methods and the tolerance standard have evolved over time to be consistent with each other. We now have these three important entities – the ASME Y14.5 standard, the traditional methods of measurement and algorithms used in CMMs. It is very important that these three entities should be consistent with each other. We know that the first two agree with each other since they evolved together. But the third entity, the CMM computational algorithms, was an independent branch in mathematics that just found an application in metrology. It was shown in the previous section that numerous attempts have been made by the research community to standardize the use of these computational algorithms. However, the efforts by these organizations and many other researchers were concerned with the mathematical correctness of the
algorithms and developing efficient and intelligent methods to overcome the inherent difficulties associated with the mathematics of these algorithms. But what is missing is a study of the interpretation of these three entities, a check of their consistency and the integrity of these CMM fitting algorithms in measuring tolerances according to the GD&T standard. Thus, the intention of this thesis is to examine the inconsistencies between the standard, traditional measurement practices and fitting algorithms. The next chapter proposes standardized algorithms for each tolerance type for use in CMM software that is consistent with the standards and manual inspection practices.
CHAPTER 7
NORMATIVE GUIDELINES FOR FEATURE FITTING ALGORITHMS IN CMM INSPECTION

7.1 Standardization of Fitting Algorithms:

The last chapter described a case study on the accuracy of CMM results using two different CMM software. The geometric variations on the part were inspected manually and with CMM software. The points from the CMM were analyzed using multiple least squares based implementations and thus the algorithms that were employed in the software were reverse engineered. Also, substitute geometry was fit to the points from the CMM using algorithms that conformed best to the definition in the standard and manual inspection practices. A comparative study of the three results was then done for each tolerance type. The analysis showed significant differences in the tolerances predicted by the analyses. It clearly highlighted the flaws in the least squares based algorithms used in the CMM software. Based on the results of the experiments, a standardization of fitting algorithms is proposed in light of the definition provided in the standard and an interpretation of manual inspection methods. Finally, a step by step explanation of the fitting process applied to a cloud of points for each tolerance type is provided.

7.2 Size Tolerance:

7.2.1 What the standard says:
“Where only a tolerance of size is specified, the limits of size of an individual feature prescribe the extent to which variations in its geometric form, as well as size, are allowed.”

“The actual local size of an individual feature at each cross section shall be within the specified tolerance of size”

7.2.2 Manual Inspection Practices:

Manual inspection practices for size tolerances indicate that size features need to be functionally inspected. This means that a hole or a slot should be inspected in such a way that it simulates the mating assembly part i.e., the pins or tabs that go into the hole and slot respectively. Usually, precision parallels, gage blocks and pins, dial indicators or a combination of these are used to measure size tolerances. From these methods, it is evident that size tolerances are simulated by mating parts that correspond to the maximum allowable extent. For example, the size of the hole corresponds to the biggest gage pin that fits into it and the size of a slot is determined by the biggest gage block that fits into it. Thus it is clear that extremal fits are more suited for the measurement of size tolerances.

7.2.3 Fitting Algorithm for Size Tolerance on Planar Feature Coupled with Orientation:

The assumptions made in the development of Tolerance Maps for size tolerances [8, 9] can be put to use for developing algorithms to fit planes to measure size tolerances. One end of the feature is first measured and a one sided fit to the data yields the first planet. The other side is then measured and then
another plane is computed using a one-sided fit to the second set of data. The distance between the centroids of the two planes give the size tolerance of the feature. The orientation of the second plane is then computed with respect to the first plane to compute the orientation necessary in the T-Map. The detailed algorithm is proposed below.

- Measure two set of data points, one on each side of the feature
- Do a one sided plane fit to each of the data sets – This gives a pair of planes that contains all the measured points within them
- Compute the distance between the two centroid of the planes
- This value gives the size of the positive feature
- Compute the orientation of the second plane with respect to the first plane

7.2.4 Fitting Algorithm for Size Tolerance on FOS Feature without Orientation:

- Measure a set of points on both sides of the feature
- Do a Two sided plane fit – This gives a pair of parallel planes that are separated by the minimum distance and also contains all the measured points within them
- Compute the perpendicular distance between the two parallel planes
- This value gives the size of the positive feature

The above algorithm is depicted in Fig. 7.1.
7.3 Parallelism Tolerance:

7.3.1 What the Standard Says:

Parallelism is the condition of a surface or center plane, equidistant at all points from a datum plane. The definition in the standard is as follows:

“A tolerance zone defined by two parallel planes, parallel to the datum plane or axis, within which the surface or center plane of the considered feature must lie.

A tolerance zone defined by two parallel lines, parallel to the datum plane or axis, within which the line element of the surface must lie.”

7.3.2 Manual Inspection Practices

The tolerance zone for the parallelism depends on the feature being controlled and the type of datum. Parallelism is usually inspected by mounting the datum feature on a simulated datum and then determining the variation of the
feature such as Full Indicator Movement (FIM) with respect to the simulated datum. The FIM is an indicator of points that are at the maximum and minimum distances from the datum, thus giving rise to the idea that one needs to do a two-sided based fit. For FOS features, the parallelism is controlled through the derived medial element of the feature. The manual inspection of such features also involve using dial indicators, gage blocks and surface plates and they measure the maximum and the minimum extent of variation of the derived medial element from the DRF. All these methods again indicate that a two-sided based fit needs to be done for evaluating parallelism tolerances.

7.3.3 Fitting Algorithm for Parallelism Tolerance on Planar Feature:

From the definition of the standard and an interpretation of the manual inspection methods, it is clear that all the points of the surface or the center plane of the FOS should lie within the tolerance zone specified by the orientation tolerance specification. If the substitute feature is analyzed for conformance within the given tolerance zone, there is every possibility that the substitute feature might lie entirely within the zone whereas the actual feature might have a few points that lie out of the zone. But according to the standard, all the points on the surface must lie in the zone.

- Measure a set of points on the datum surface
- Do a ‘one-sided plane fit’ for the datum inspection points to get a plane that contacts the three high points of the primary datum feature. This
primary datum plane should have at least a three point contact with the primary datum feature.

- For cylindrical features such as holes and pins, measure a set of points on the feature being used as datum.
- Do a ‘Maximum Inscribed Cylinder (MIC)’ fit, if the datum feature is a hole and do a constrained ‘Minimum Circumscribed Cylinder (MCC)’ fit if the datum feature is a pin. For more explanation on fitting of representation of axes as datums, the reader is referred to [47].
- Measure a set of points on the surface being inspected.
- Do a constrained ‘two-sided plane fit’ that yields a pair of parallel planes that are separated by the minimum distance and contains all the measured points.
- The two-sided fitting is constrained such that the planes are oriented parallel to the datum surface fit in the previous step.
- The perpendicular distance between the two planes gives the parallelism tolerance used up by the surface being inspected.
7.3.4 Fitting Algorithm for Parallelism Tolerance on FOS

- Measure a set of points on the datum feature that are applicable.
- Do a ‘one-sided plane fit’ to the measured points to get the primary datum plane. This datum plane is formed by the three high points of the primary datum feature.
- For cylindrical features such as holes and pins, measure a set of points on the feature being used as datum.
- Do a ‘Maximum Inscribed Cylinder (MIC)’ fit, if the datum feature is a hole and do a constrained ‘Minimum Circumscribed Cylinder (MCC)’ fit if the datum feature is a pin. The fit is constrained depending on the feature being a primary or secondary datum.
- Split the FOS into ‘N’ number of small subsets along the length of the FOS.
- Measure a set of points on both sides of the FOS for every subset.
• For every subset, do a ‘two-sided plane fit’. We get a pair of parallel planes.

• For a tab, a positive FOS, we do a positive two-sided fit. For a slot, which is a negative FOS, we do a negative two-sided fit.

• For each pair of planes representing each subset, the centroid of the space enclosed between the two parallel planes of the chosen pair is computed. Let us call this the ‘secondary point’ of the first subset.

• Likewise, the secondary points for all the subsets of measured points are computed.

• We then do another constrained ‘two-sided plane fit’ to these N secondary points. This Two-sided fit is constrained such that the planes are oriented parallel to the datum plane that we fit in the beginning.

• This two-sided fit yields a pair of parallel planes that contains all the secondary points between them and are separated by the minimum possible distance

• The perpendicular distance between these two lines gives the parallelism tolerance used up by the center plane of the FOS being inspected. This is shown in Fig. 7.3.
7.4 Perpendicularity Tolerance:

7.4.1 What the standard says:

“A tolerance zone defined by two parallel planes, perpendicular to the datum plane or axis, within which the surface or center plane of the considered feature must lie.

A tolerance zone defined by two parallel lines, perpendicular to the datum plane or axis, within which the line element of the surface must lie.”

7.4.2 Manual Inspection Practices:

The tolerance zone for the perpendicularity tolerance depends on the feature being controlled and the type of datums. One should remember that perpendicularity tolerances can have multiple datums and hence, fixing secondary datums also plays a major part in inspection. These are usually achieved by using different types of gage blocks for different datums. Perpendicularity is usually inspected by mounting the datum features on simulated datums such as L and hole.
blocks and then determining the variation of the feature such as a Full Indicator Movement (FIM) with respect to the simulated datums. The measurement methods are quite similar to parallelism tolerances and hence these also require two-sided Chebyshev based fitting algorithms.

7.4.3 Fitting Algorithm for Perpendicularity Tolerance on Planar Feature:

- Measure sets of points on the datum features
- Do a ‘one-sided plane fit’ for the datum inspection points to get a plane that contacts the three high points of the primary datum feature. This primary datum plane should have at least a three point contact with the primary datum feature.
- Similarly, do a one-sided fit for the points corresponding to the secondary datum. This fit should be constrained such that it is perpendicular to the primary datum fit in the above step and should have a minimum of two point contact.
- For cylindrical features such as holes and pins, measure a set of points on the feature being used as datum.
- Do a ‘Maximum Inscribed Cylinder (MIC)’ fit, if the datum feature is a hole and do a constrained ‘Minimum Circumscribed Cylinder (MCC)’ fit if the datum feature is a pin. The fit is constrained depending on the feature being a primary or secondary datum.
- Measure a set of points on the surface being inspected.
- Do a constrained ‘two-sided plane fit’ that yields a pair of parallel planes that are separated by the minimum distance and contains all the measured points.
- The two-sided fitting is constrained such that the planes are oriented parallel to the datum surface fit in the previous step.
- The perpendicular distance between the two planes gives the parallelism tolerance used up by the surface being inspected.

![Diagram](image)

Fig. 7.4. Fitting for perpendicularity tolerance on a planar feature

### 7.4.4 Fitting Algorithm for Perpendicularity Tolerance on FOS

- Measure a set of points on the datum feature that are applicable
- Do a ‘one-sided plane fit’ to the measured points to get the primary datum plane. This datum plane is formed by the three high points of the primary datum feature.
• Similarly, do a one-sided fit for the points corresponding to the secondary
datum. This fit should be constrained such that it is perpendicular to the
primary datum fit in the above step and should have a minimum of two
point contact.
• For cylindrical features such as holes and pins, measure a set of points on
the feature being used as datum.
• Do a ‘Maximum Inscribed Cylinder (MIC)’ fit, if the datum feature is a
hole and do a constrained ‘Minimum Circumscribed Cylinder (MCC)’ fit
if the datum feature is a pin. The fit is constrained depending on the
feature being a primary or secondary datum.
• Split the FOS into ‘N’ number of small subsets along the length of the
FOS
• Measure a set of points on both sides of the FOS for every subset
• For every subset, do a ‘two-sided plane fit’. We get a pair of parallel
planes.
• For a tab, a positive FOS, we do a positive two-sided fit. For a slot, which
is a negative FOS, we do a negative two-sided fit.
• For each pair of planes representing each subset, the centroid of the space
enclosed between the two parallel planes of the chosen pair is computed.
Let us call this the ‘secondary point’ of the first subset.
• Likewise, the secondary points for all the subsets of measured points are
computed.
• We then do another constrained ‘two-sided plane fit’ to these N secondary points. This Two-sided fit is constrained such that the planes are oriented parallel to the datum plane that we fit in the beginning.

• This two-sided fit yields a pair of parallel planes that contains all the secondary points between them and are separated by the minimum possible distance.

• The perpendicular distance between these two lines gives the parallelism tolerance used up by the center plane of the FOS being inspected.
7.5 Angularity Tolerances:

7.5.1 What the Standard Says:

“A tolerance zone defined by two parallel planes that are at specified basic angle from one or more datum planes or a datum axis, within which the surface or the center plane of the considered feature must lie.

A tolerance zone defined by two parallel lines at the specified basic angle from a datum plane or axis within which the line element of the surface must lie”
7.5.2 Manual Inspection Practices:

The tolerance zone for the angularity tolerance depends on the feature being controlled and the type of datum. The principle of measuring angularity is very similar to measuring parallelism. The difference is that the inspection surface is made parallel to the surface plate and inspected using a dial indicator. The FIM gives the angularity tolerance used up by the part. Angularity is usually inspected by mounting the datum feature on a simulated datum and then determining the variation of the feature such as Full Indicator Movement (FIM) with respect to the simulated datum.

7.5.3 Fitting Algorithms for Angularity on Planar Feature:

- Measure a set of points on the datum surface
- Do a ‘one-sided plane fit’ for the datum inspection points to get a plane that contacts the three high points of the primary datum feature. This primary datum plane should have at least a three point contact with the primary datum feature.
- For cylindrical features such as holes and pins, measure a set of points on the feature being used as datum.
- Do a ‘Maximum Inscribed Cylinder (MIC)’ fit, if the datum feature is a hole and do a constrained ‘Minimum Circumscribed Cylinder (MCC)’ fit if the datum feature is a pin.
- Measure a set of points on the surface being inspected.
• Do a constrained ‘two-sided plane fit’ that yields a pair of parallel planes that are separated by the minimum distance and contains all the measured points.

• The two-sided fitting is constrained such that the planes are oriented at the base angle to the datum surface fit in the previous step. The base angle or the nominal angle is mentioned along with the tolerance specifications.

• The perpendicular distance between the two planes gives the angularity tolerance used up by the surface being inspected.

Fig. 7.6. Fitting for angularity tolerance on a planar feature

### 7.5.4 Fitting Algorithm for Angularity Tolerance on FOS

• Measure a set of points on the datum feature that are applicable

• Do a ‘one-sided plane fit’ to the measured points to get the primary datum plane. This datum plane is formed by the three high points of the primary datum feature.
• For cylindrical features such as holes and pins, measure a set of points on
the feature being used as datum.

• Do a ‘Maximum Inscribed Cylinder (MIC)’ fit, if the datum feature is a
hole and do a constrained ‘Minimum Circumscribed Cylinder (MCC)’ fit
if the datum feature is a pin. The fit is constrained depending on the
feature being a primary or secondary datum.

• Split the FOS into ‘N’ number of small subsets along the length of the
FOS

• Measure a set of points on both sides of the FOS for every subset

• For every subset, do a ‘two-sided plane fit’. We get a pair of parallel
planes.

• For a tab, a positive FOS, we do a positive two-sided fit. For a slot, which
is a negative FOS, we do a negative two-sided fit.

• For each pair of planes representing each subset, the centroid of the space
enclosed between the two parallel planes of the chosen pair is computed.
Let us call this the ‘secondary point’ of the first subset.

• Likewise, the secondary points for all the subsets of measured points are
computed.

• We then do another constrained ‘two-sided plane fit’ to these N secondary
points. This Two-sided fit is constrained such that the planes are oriented
at the base angle to the datum plane that we fit in the beginning.
- This two-sided fit yields a pair of parallel planes that contains all the secondary points between them and are separated by the minimum possible distance.
- The perpendicular distance between these two lines gives the angularity tolerance used up by the center plane of the FOS being inspected.

Fig. 7.7. Fitting for angularity tolerance on a FOS feature

7.6 Flatness Tolerance:

7.6.1 What the standard says:

“*A Flatness tolerance specifies a tolerance zone defined by two parallel planes within which the surface must lie.*”

7.6.2 Manual Inspection Practices:

Form tolerances are usually inspected by running a dial indicator all over the surface being controlled. The FIM of the dial indicator is taken as the form...
tolerance of the particular surface. This makes it clear that form tolerances do not control the orientation of the surface and the manual measurement methods indicate the points which are farthest apart. In other words, the highest peak and the lowest valley are identified.

7.6.3 Fitting Algorithms for Flatness Tolerances:

Based on the arguments above, an algorithm for evaluating flatness tolerance is proposed:

- Measure a set of points on the surface to be inspected.
- Do a ‘two-sided plane fit’ to the set of measured points.
- The two-sided fit yields a pair of parallel planes that contains all the measured points between them and is separated by the minimum possible distance.
- The perpendicular distance between the two planes gives the flatness tolerance used up by the surface

![Diagram](image)

Fig. 7.8. Fitting for flatness tolerance on a planar face
7.7 Position Tolerances

7.7.1 What the Standard Says:

“While maintaining the specified width limits of the feature, its center plane must be within a tolerance zone defined by two parallel planes equally disposed about true position, having a width equal to the positional tolerance. This tolerance zone also defines the limit within which variations in attitude of the center plane of the feature must be confined.

While maintaining the specified width limits of the feature, no element of its side surfaces shall be inside a theoretical boundary defined by two parallel planes equally disposed about true position and separated by a distance equal to W”

Where, W = (MMC size) plus or minus (Position Tolerance value), depending on the feature being internal or external.

7.7.2 Manual Inspection Methods:

Position tolerances are applied to control the location of FOS features. These are usually gaged functionally by simulating their mating features. This is usually done by using corresponding mating features such as pins and hole gages. Sometimes, they are also controlled by mounting the parts on specialized fixtures and using a combination of dial indicators to ascertain the location of the features being controlled by measuring the corresponding sizes that control the features.

7.7.3 Fitting Algorithm for Position Tolerance Based on Center Plane of a Slot:
• Measure a set of points on the datum features that are applicable to the particular tolerance specified

• Do a ‘one-sided plane fit’ to the measured points to get the datum plane. This datum plane is formed by the three high points of the datum feature. Similarly, fit the secondary and tertiary datums where applicable using a constrained ‘one-sided plane fit’. Hence determine the Datum Reference Frame (DRF).

• Compute the center plane of the FOS w.r.t the nominal (basic) dimensions and the DRF fixed in the above step. Let us call this the ‘Nominal Center Plane of the FOS’

• Split the FOS into ‘N’ number of small subsets along the length of the FOS

• Measure a set of points on both sides of the FOS for every subset

• For every subset, do a ‘two-sided plane fit’. We get a pair of parallel planes.

• For a tab, a positive FOS, we do a positive two-sided fit. For a slot, which is a negative FOS, we do a negative two-sided fit.

• For each pair of planes representing each subset, the centroid of the space enclosed between the two parallel planes of the chosen pair is computed. Let us call this the ‘secondary point’ of the first subset.

• Likewise, the secondary points for all the subsets of measured points are computed.
• We then do another constrained ‘two-sided plane fit’ on these N secondary center points. This two-sided fit is constrained such that the fit planes are oriented parallel to the ‘Nominal Center Plane of the FOS’ that we determined before.

• This yields a pair of parallel planes that contains all the secondary points between them.

• The perpendicular distance of the two extreme planes of this ‘two-sided fit’ from the ‘Nominal Center Plane of the FOS’ is computed. The greater distance of the two planes is determined. Twice this distance gives the position tolerance used up by the part.

Fig. 7.9. Fitting for position tolerance based on the center plane of a slot
7.7.4 Fitting Algorithm for Position Tolerance Based on Side Surfaces of a Slot:

- Measure a set of points on the datum features that are applicable to the particular tolerance specified
- Do a ‘one-sided plane fit’ to the measured points to get the datum plane. This datum plane is formed by the three high points of the datum feature. Similarly, fit the secondary and tertiary datums where applicable using a constrained ‘one-sided plane fit’. Hence determine the Datum Reference Frame (DRF).
- Compute the plane that represents the center plane of the FOS w.r.t the nominal (basic) dimensions and the DRF fixed in the above step. Let us call this the ‘Nominal Center Plane of the FOS’.
- Measure a set of points on one of the side surfaces of the FOS.
- Do a constrained ‘one-sided plane fit’ to these points. The fit is constrained such that the plane is parallel to the ‘Nominal Center Plane of the FOS’.
- Measure a set of points on the other side surface of the FOS and repeat the constrained plane fit as above.
- We get two planes which are parallel to the ‘Nominal Center Plane of the FOS’. Compute the perpendicular distance of each of these two planes
from the ‘Nominal Center Plane of the FOS’ and select the plane with the greater distance.

- Multiply this greater distance by 2 to get the width, W of the FOS.
- Now, do a constrained ‘two-sided plane fit’ to the set of measured points including both the side surfaces. The fit is constrained such that the planes are perpendicular to the DRF we fit in the beginning.
- Compute the perpendicular distance between these two parallel planes. This gives the MMC size of the FOS.
- Now, compute the positional tolerance ‘tp’ used up by the part from the MMC size and the width of the slot depending on if it’s a slot or a tab. For an internal FOS, $W = \text{MMC Size} + \text{tp}$ and for an external FOS, $W = \text{MMC Size} - \text{tp}$.
Fig. 7.10. Fitting for position tolerance based on side surfaces of a slot

Internal FOS --> $W = \text{MMC Size} + \text{Position Tolerance}$
External FOS --> $W = \text{MMC Size} - \text{Position Tolerance}$
8.1 Tolerances in Inspection Stage:

In the previous section, we saw the standardization of fitting algorithms for various tolerance types and feature types. In the earlier chapters, the idea of T-Maps was introduced. We can recollect the definition of T-Map as

“A hypothetical Euclidean point-space, the size and shape of which reflect all variational possibilities for a target feature. It is the range of points resulting from one-to-one mapping from all variational possibilities of a feature within its tolerance zone.”

T-Map is a mathematical model to represent the geometric tolerances in the design stage. A T-Map for a tolerance applied on a feature represents the entire gamut of variations that can arise later due to manufacturing imprecision. This not only provides a tool for the designer to visualize the range of variations on a single feature, but also helps the designer to understand the interaction with other tolerances and features and the accumulation that results out of this interaction. But the T-Maps are for the design phase of the product life cycle. We do not have a mathematical representation at the inspection stage to work along with these T-Maps to represent the actual manufactured part. It is important to recollect the two axioms of manufacturing stated by Srinivasan [40].

- Axiom of manufacturing imprecision: All manufacturing processes are inherently imprecise and produce parts that vary.
- Axiom of measurement uncertainty: No measurement can be absolutely accurate and with every measurement, there is some finite uncertainty about the measure attribute or measured value.

So, it becomes imperative that a mathematical representation of actual parts is needed in the inspection stage and hence can be used in conjunction with the T-Maps. The intention of this chapter is to develop a model known as the Inspection Map or the i-Map.

The axioms of manufacturing imprecision and measurement uncertainty makes it obvious that we can neither create perfectly ideal parts nor can we make perfect measurement of the dimension associated with the feature. This is the reason for the existence of the tolerances in design and manufacturing and standard that defines the principles of using these tolerances. This means that a small amount of variation is permitted and there is some room for approximation. Recent technological advancements in manufacturing, tooling and inspection capabilities and the computerization of these processes have made the job of engineers easy, but still one has to do some approximation. The success of the process depends upon how accurately we approximate a given part in inspection. This step is sometimes assumed to be trivial and not enough time and effort is invested in investigating the results of the inspection facilities. The quality control department blindly believes the results of the CMM machine or other automated inspection tools. The results may not always be accurate and a few examples have been illustrated in the previous chapters using some simple parts and basic
tolerances. The inspection results are not only used to evaluate features and tolerances, but also to monitor the processes that produce the parts.

Incorrect or unreliable inspection results can result in two types of issues. First, a valid or a conforming part could be rejected as scrap. This might result in an inefficient process and hence lead to huge losses in the long run. Also, a large number of rejected parts usually reflect problems in the manufacturing process and hence there is every possibility that a perfectly running process could be disturbed based on wrong data. This results in production stoppages and hence decreases productivity. Second, a wrong part may be accepted as the right part. The ramifications of this are not only within the organization, but on the customer side as well. A part that does not conform to tolerances may not deliver good performance as envisioned by the designer and hence results in customer dissatisfaction. Also, there is every chance that there are problems with the manufacturing process that produces these bad parts. The problems accumulate and could lead to serious consequences if they are left unattended and overlooked. The inspection results are thus the ultimate decision making tools and they influence a lot of technical and management decisions. This involves significant amount of time, money and effort spent by any organization that wants to keep its operations efficient.

8.2 Inspection Maps (i-Maps):

The need for accurate representation of substitute features was described in the earlier section. The following sections describe the development of
Inspection Map or the i-Map for different tolerance classes applied on planar features. The definition of an i-Map is as follows.

“An i-Map is a hypothetical point-space which encompasses all the variations of parts in an inspection sample, where each part in the sample is represented by a point-space computed from the substitute feature fitted to a set of data points measured with a CMM. The size, shape and location of an i-Map represent the degree of conformance of a sample to the design specifications and the i-Map is established in the same coordinate frame as the T-Map which represents the intended limits for geometric variations defined by the design specifications”.

The i-Map is derived from what is called the i-zone. The i-zone can be defined as the bounded region of points that represent the actual feature that are measured with a CMM. The data points are expressed in the xyz Cartesian coordinate system that is defined by the operator of the CMM or assumed within the CMM software. Just as tolerances are always associated with datums, the i-zones are also point cloud data associated with the measured features. It should be noted here that the point clouds of the datum and the i-zone should be measured in the same coordinate system to remove any propagation of error and eliminate unnecessary post-processing of the point cloud data. Also, the datum features used for alignment and measurements in the CMM should be the same as the ones specified by the design specifications. From the definition of the i-Maps and T-Maps, one can understand that the i-Map derived from the i-zone should lie within
the boundary defined by the T-Map, if the particular part meets the design specifications. If the i-Map falls outside of the T-Map, then either the part is rejected or sent for rework.

The first step in the computation of i-Maps is the determination of substitute features for the particular tolerance and feature type. i-Maps are envisioned for a batch of parts, since the usual practice in the industry is to inspect a sample or batch of parts from the entire population. In order to do this, the substitute features to represent the actual samples are first computed. The i-Maps for each batch of parts are then computed as the geometric mean or the centroid of the points that represent the substitute features. Thus we have an i-Map for each batch of inspection that summarizes the degree of conformance of the particular batch to the tolerance specifications. This thesis is restricted to size, orientation and form tolerances on planar, circular and cylindrical features.

8.3 The i-Map for Size Tolerance on a Round Face:

The T-Map for the size tolerance on a round face was discussed in Chapter 3. The tolerance zone for the round face with size tolerance is shown in Fig. 8.1. The planes represented by $\sigma_1$ and $\sigma_2$ are the extreme size variations allowed. All the cylindrical bars manufactured must have all their points within these two planes. A typical CMM inspection of such a bar would look like the one depicted in Fig. 8.2. A cross sections of Fig. 8.2 is illustrated in Fig. 8.3 for better illustration.
Fig. 8.1. Tolerance Zone for Size Tolerance on Planar Circular Face

Fig. 8.2. CMM data points measured on a planar circular face
Fig. 8.3. Cross section of the tolerance zone of a planar circular face with the CMM data points and the basis planes of the T-Map

Once the points are measured using the CMM, the substitute feature is fitted using the algorithm described in section 7.2.3. This involves doing two one-sided fits to the point clouds corresponding to the datum plane A and the controlled feature. The detailed algorithm for determining the i-Map for this case is detailed below.

- Do a one-sided fitting for the points corresponding to plane A that represents the datum. The direction cosines of the plane are computed from the co-efficients of the x, y, and z terms of the equation of the plane A with respect to the global coordinate system.
- Do a one-sided fitting for the points corresponding to the controlled feature. The direction cosines of the plane are computed from the co-efficients of the x, y, and z terms in the equation of the plane with respect to the global coordinate system.
• Compute the orientation of the feature plane with respect to plane A from their respective direction cosines. We now get the angles between the two planes in the x and y directions.

• These two angles give the p and q coordinates in the T-Map.

• The p, q coordinates are multiplied by the diameter of the round bar, ‘d’, to get p’ and q’ respectively.

• Compute the centroid of the plane A. The perpendicular distance ‘D’ of the centroid of the plane A from the feature plane gives the centroid-to-centroid distance between the two planes.

• Compute the value D – Nominal Size of the round bar.

• This gives the ‘s’ coordinate at which the i-Map is located inside the T-Map.

• Plot the point I with coordinates (p’, q’, s) within the T-Map. This point is the representation of the actual part within the T-Map.

Once the substitute features for the entire sample is computed, the centroid of the sample is computed as the i-Map of the sample. Representation of two different parts are illustrated below. Fig. 8.4 shows a substitute feature that is tilted about the x axis and whose centroid lies above the centroid of the tolerance zone. Fig. 8.6 shows a substitute feature that is parallel to the planes $\sigma_1$ and $\sigma_2$ and whose centroid lies below the centroid of the tolerance zone. Transformations for these two parts are shown as the point inside the T-Maps in Fig. 8.5 and Fig. 8.7 respectively.
Fig. 8.4. One-Sided Substitute feature fit to the CMM data points – Case 1

Fig. 8.5. Representation of substitute feature for size tolerance on planar circular face inside parent T-Map – Case 1
Fig. 8.6. Substitute feature fit to the CMM data points – Case 2

Fig. 8.7. Representation of substitute feature for size tolerance on planar circular face inside parent T-Map – Case 2
8.4 I-Map for Size Tolerance on Rectangular Face:

The tolerance zone for an size tolerance on a rectangular face is shown in Fig. 8.8.

![Fig. 8.8. Tolerance Zone for Size Tolerance on Planar Rectangular Face with measured CMM data points.](image)

The method for computing the substitute feature for the rectangular face is very similar to the method described for the round face. Once the points are measured using the CMM, the substitute feature is fit using the algorithm described in section 7.2.3. This involves doing two one-sided fits to the point clouds corresponding to the datum plane A and the controlled feature. The detailed algorithm for determining the i-Map for this case is detailed below.

- Do a one-sided fitting for the points corresponding to plane A. The direction cosines of the plane are computed from the co-efficients of the x, y, and z terms of the equation of the plane A with respect to the global coordinate system.
• Do a one-sided fitting for the points corresponding to the controlled feature. The direction cosines of the plane are computed from the coefficients of the x, y, and z terms in the equation of the plane with respect to the global coordinate system.

• Compute the orientation of the feature plane with respect to plane A from their respective direction cosines. We now get the angles between the two planes in the x, y and z direction.

• The first two angles, i.e., in the x, y give the p, q coordinates in the T-Map.

• The p, q coordinates are multiplied by the length of the rectangular face, to get p’ and q’ respectively.

• Compute the centroid of the plane A. The perpendicular distance ‘D’ of the centroid of the plane A from the feature plane gives the centroid-to-centroid distance between the two planes.

• Compute the value D – Nominal Size of the round bar.

• This gives the ‘s’ coordinate inside the T-Map.

• Plot the point I with coordinates (p’, q’, s) within the T-Map. This point is the representation of the substitute feature of the round face.

Once the substitute features for the entire sample is computed, the centroid of the sample is computed as the i-Map of the sample. Fig. 8.9 shows a substitute feature that is parallel to the planes \( \sigma_1 \) and \( \sigma_2 \) and passes through the centroid of the tolerance zone. Fig. 8.11 shows a substitute feature that is tilted clockwise to
the x axis. The angle of tilt is almost equal to the maximum tilt permitted. Substitute features for these two parts are shown as the points inside the T-Maps in Fig. 8.10 and Fig. 8.12 respectively.

Fig. 8.9. Tolerance Zone for Size Tolerance on Planar Rectangular Face with measured CMM data points and substitute feature – Case 1

Fig. 8.10. Representation of substitute feature for size tolerance on a planar rectangular face inside parent T-Map – Case 2
Fig. 8.11. Tolerance Zone for Size Tolerance on Planar Rectangular Face with measured CMM data points and substitute feature – Case 2

Fig. 8.12. Representation of substitute feature for size tolerance on a planar rectangular face inside parent T-Map – Case 2
8.5 Orientation Tolerance Independent of Size:

The last two sections described the development of i-Maps for size tolerances on planar faces. The T-Map for those two cases combined the orientation of the planar face with the size tolerance. This was possible because, the controlled feature was a part of the cylindrical bar which is a FOS feature. However, there may be cases when the planar face has to be controlled and evaluated for orientation independent of size or the planar face may not be a part of an FOS feature. In most FOS features, the datum with respect to which the feature is controlled may not be a part of the FOS feature. In these cases, the orientation has to be controlled independently. Examples of such cases are illustrated in Fig. 8.13 and Fig. 8.14.

Fig. 8.13. Tolerance zone for parallelism tolerance applied on a planar surface
8.5.1 T-Map for Orientation Applied to Planar Faces:

A T-Map exclusively for orientation tolerance is not available. So, before developing an i-Map for the case, it is necessary to have the T-Map on which the i-Map could be overlaid. From Fig. 8.13 and Fig. 8.14 it is clear that the tolerance zone for orientation tolerance is oriented with respect to the datum to which it is referred. The tolerance zone is defined by two parallel planes separated by the allocated orientation tolerance and the planes are oriented at the base angle to the datum feature. The ASME standard also defines the orientation tolerance not in terms of angular dimensions, but in terms of linear dimensions. But in the T-Maps developed for planar features, the orientation of the substitute feature is represented in terms of angular dimensions. The angle that the substitute feature makes with the x and y axes are translated into the q’ and p’ coordinates respectively. Since the tolerance zone is represented in terms of linear dimensions, it makes orientation tolerance independent of the shape of the feature.
A T-Map for orientation tolerance applied independent of size tolerance can be
developed on similar lines to the T-Map for size tolerance on a planar face.

Fig. 8.15. Tolerance zone for parallelism tolerance applied on a planar surface
exaggerated to show the basis planes and the coordinate system.

Fig. shows the tolerance zone for an orientation tolerance applied on a planar round face. The planes $\sigma_1$ and $\sigma_2$ form the boundary of the tolerance zone. The planes are parallel to datum plane A and are separated by a distance of 0.05. All points of the feature should fall within the tolerance zone defined by the planes $\sigma_1$ and $\sigma_2$. However, the entire tolerance zone defined by ABFEGCDH can float inside the size tolerance zone, if a size tolerance is also applied independently. The difference between the T-Map for size and orientation starts after this step of constructing the tolerance zone. According to the ASME
standard, the orientation tolerance is measured by linear distance that separates two parallel planes within which all the points of the feature lies. Hence, the orientations of the planes that are fit do not change and are always parallel to the datum feature. Hence, the basis points of the areal coordinate system are the two planes $\sigma_1$ and $\sigma_2$. The two points are separated by a distance of $t''$. The orientation of the two planes defining the orientation tolerance is fixed by the datum and they do not have any orientational degrees of freedom. The only degree of freedom available for this zone is the size of the zone or the distance that separates the two planes. Thus, the T-Map for the orientation tolerance becomes just a straight line connecting these two basis points. The two basis points lie on either side of the origin since we have chosen the origin of the coordinate system of the tolerance zone to be midway between the two planes $\sigma_1$ and $\sigma_2$. The T-Map for orientation tolerance is illustrated in the Fig. 8.16.

Fig. 8.16. T-Map for orientation tolerance applied on a planar face
8.5.2 The i-Map for Orientation Applied to Planar Faces:

The last section described the development of T-Map for orientation applied on a planar face independent of size. Inspection of orientation tolerances using CMM involves computing a two sided fit constrained to the orientation of the DRF. The width of the two sided fit gives the orientation tolerance used up by the actual part. Detailed CMM fitting algorithms for orientation tolerances are explained in sections 7.3 – 7.5. The only parameter required for determining the i-Map for orientation tolerance is the width of the two sided fit, ‘d’. This is because, the width is the only degree of freedom available when orientation is applied independent of other tolerances. The two parallel planes of the two sided fit are plotted on either side of the origin of the T-Map depicted in Fig. 8.16. The two points are equally spaced about the origin and are at a distance of d/2 from the origin. Once the substitute features for the entire sample is computed, the mean of the sample is computed as the i-Map of the sample. Thus, the i-Map for orientation tolerance is a line on a line. This is illustrated in Fig. 8.16. The line $\sigma_1O\sigma_2$ is the T-Map represented by the dotted line. The solid line $\sigma_1'O\sigma_2'$ represents the i-Map.
8.6 I-Map for Flatness Tolerances:

8.6.1 Flatness Tolerances Coupled with Size:

The development of T-Maps for form tolerances on a planar face are illustrated in section 3.3. The T-Maps for form tolerances are modeled as internal subsets of the size tolerance. The inspection of form tolerance involves determining two parallel planes that are separated by the least distances and contain all the points of the feature within them. The substitute features are then computed from the two planes that are a result of the two-sided fit done on the data points. The detailed algorithm for computing the substitute feature is described below:

- Do a one-sided fitting for the points corresponding to plane A. The direction cosines of the plane are computed from the co-efficients of the x, y, and z terms of the equation of the plane A with respect to the global coordinate system.
• Do a two-sided fitting for the points corresponding to the controlled feature. This gives a set a parallel planes and the perpendicular distance between the planes ‘d’.

• The direction cosines of the mid-plane of the two-sided fit are computed from the co-efficients of the x, y, and z terms in the equation of the plane with respect to the global coordinate system.

• Compute the orientation of the mid-plane with respect to plane A from their respective direction cosines. We now get the angles between the two planes in the x, y and z direction.

• The first two angles, i.e., x, y give the p, q coordinates in the T-Map.

• The p, q coordinates are multiplied by the diameter of the round bar, ‘d’, to get p’ and q’ respectively.

• Compute the centroid of the plane A. The perpendicular distance ‘D’ of the centroid of the plane A from the mid-plane gives the centroid-to-centroid distance between the two planes.

• Compute the value (D – Nominal Size of the round bar).

• This gives the ‘s’ coordinate of the mid-plane inside the T-Map.

• Plot the point I with coordinates (p’, q’, s) within the T-Map.

• The areal coordinates of the top and bottom planes of the two-sided fit inside the T-Map are computed as (p’, q’, s+d/2) and (p’, q’, s-d/2) respectively.
• Now, construct a dicone with the point corresponding to the mid-plane as the origin. The radius and the apex to apex height of the dicone is the distance ‘d’.

This method of evaluating the substitute features for form coupled with size tolerance is very powerful. This is because, the form tolerance zone is allowed to be oriented in any angle and its position can float anywhere within the tolerance zone as long as it conforms to Rule 1 of the ASME standard. The current i-Map not only gives the form tolerance used up by the part, but also gives an idea of the overall orientation and position of the feature within the tolerance zone. This is important because, this can act an indicator in process control to determine how close a part is to the actual limits given by the design specifications. If the orientation and overall position are not taken into account, then the part which is very close to the limits will be treated no different than a perfect part and there is not much data available for inference during process control. The i-zones and the corresponding i-Maps for three different parts is illustrated in Fig. 8.17.
8.6.2 Flatness Tolerances Independent of Size:

Similar to orientation tolerances, form tolerances are also sometimes controlled and evaluated independently of size. Inspection of form tolerances using CMM involves computing a two sided fit. The width of the two sided fit gives the form tolerance used up by the actual part. Detailed CMM fitting algorithms for form tolerances are explained in sections 7.6. The T-Map for
flatness tolerance is a line. This line for the flatness tolerance is derived in the same lines for the orientation tolerance described in section 8.5. The two points corresponding to the two basis planes form the end points of the line. The two points are placed on either side of the origin and separated by a distance specified by the form tolerance ($t'$).

The only parameter required for determining the I-Map for flatness tolerance independent of size is the width of the two sided fit, ‘d’. This is because, the width is the only degree of freedom available when flatness is applied independent of other tolerances. The two parallel planes of the two sided fit are plotted on either side of the origin of the T-Map. The two points are equally spaced about the origin and are at a distance of $d/2$ from the origin. Thus, the I-Map for flatness tolerance is a line on a line. This is illustrated in Fig. 8.18. The line $\sigma_1O\sigma_2$ is the T-Map represented by the dotted line. The solid line $\sigma_1'O\sigma_2'$ represents the I-Map. Since the width separating the two planes of the two-sided fit is the only degree of freedom, the shape of the controlled feature does not have any effect on the T-Map or the I-Map for flatness tolerance.

Fig. 8.19. Representation of substitute feature for flatness tolerance applied on a planar surface overlaid on the parent T-Map
CHAPTER 9

QUALITY CONTROL USING INSPECTION MAPS

9.1 Quality in Manufacturing:

Quality is one word around which the annual reports, future plans and objectives of every organization are built upon. This is because quality is one of the foremost attributes that customers look at when selecting among competing products and services. Consequently, it becomes imperative that companies understand what quality really means and take continuous efforts to improve the quality of their products. Any organization that integrates quality successfully in their work culture and overall business strategy is bound to reap significant returns on their investment. The last chapter discussed the development of Inspection Maps for different feature and tolerance types. This intent of this chapter is to integrate these inspection maps into different quality control techniques that are available in manufacturing industries.

Quality is traditionally defined as the condition of products meeting the requirements of customers and satisfying its functional needs. However, a modern definition of quality given by [30] states that

“Quality improvement is the reduction of variability in products and processes.”

The T-Maps and i-Maps are indeed the measures of the degree of variation of a feature from its design specifications. There are many more definitions of quality [35] such as the following:
• Fitness for use
• Conformance to specifications
• Excellence in products and services
• Total customer satisfaction and exceeding customer expectations

It is obvious that quality of conformance is influenced by the dimensioning and tolerancing scheme followed, choice of manufacturing processes, the type of quality control systems used and finally the effort invested by the organization to achieve and control quality. The later parts of this chapter deal specifically with the ‘quality of conformance’ of the product. This is a branch of quality control that deals with controlling the degree of variability of products and concentrates on ‘conformance to dimensional specifications’ of parts. Quality is monitored and controlled statistically in most manufacturing companies because 100% inspection is neither possible nor efficient. Advanced inspection techniques and optical scanning machines have tremendously improved inspection capabilities. However, they are still not completely mature and are still wrought with problems. Some of the few issues with CMM software are explained in sections 6.5 – 6.7. In the ensuing sections, the use of Inspection Maps in quality control and their integration with Statistical Process Control (SPC) techniques are discussed.

9.2 Statistical Quality Control:

In the earlier chapters, it was described how it is impossible to produce perfect or ideal parts and hence no two parts are identical to each other. The
differences in might be incomprehensible to the naked eye, but still there are minute differences and variability is omnipresent. The task of the quality engineer is to minimize this variability as much as possible. Sources of variability include differences in materials, operating conditions, defects in the manufacturing equipment and process and differences in the way operators perform. Even if key process variables are maintained at constant, it is very difficult or even impractical to attempt to maintain the variables at an exact state over a long period of time. Hence a range of values for the process variables is permitted and hence a little variation is allowed in the product. Though the root cause of the variability can be identified, it is still a random process and a deterministic approach towards quality control will not work. Since variability can be described only in statistical terms, statistical methods play a very important role in quality control methods.

Variations are classified into two categories. First is the common cause variation, which is embedded within the process. Common cause variations cannot be eliminated because it is a part of the process itself and the goal of SPC tools is to minimize these variations. The second type is the assignable or special cause variation. It affects the process in unpredictable ways and can be eliminated. A process is said to be in statistical control when it is free of any assignable cause variations. The SPC tools are an effective method of identifying these two types of variations and thus a direct indicator of the production quality and process capability. While SPC tools are valuable indicators of trouble, it does
not solve or rectify the problem. There is always an expected pattern or result that is associated with these tools and problems are identified when actual results differ from the existing pattern. The potential problem must then be identified, investigated and eliminated. The problems may vary so much that they could involve just the operator himself, the team of operators in that department or even sometimes necessitate the formation of a special team of experts to solve the problem. A cause and effect diagram as shown in Fig. 9.1 is a very effective tool in analyzing the problems in the manufacturing process. The team of experts usually brainstorm the root causes for the problems based on the results shown by the SPC tools. The ideas are divided into four main categories as Methods, Measurements, Materials and Machines. This is because, there can be multiple factors that affect the performance of the manufacturing equipment and it is necessary to approach each one of them individually.

Fig. 9.1. Fish Bone or the Cause and Effect Diagram
9.3 Control Charts:

In SPC applications, small samples are taken from the process and their average values are tracked using a control chart. The control charts are thus a representation of a large set of data sampled from the population. The mean $\bar{x}$ and the standard deviation $s$ of the sample are excellent estimates of their corresponding population measures. The three major steps involved in SPC are listed as follows [30, 35, 37]:

- **Data Collection:** This is the step where parts from the production line are sampled and inspection is carried out.
- **Data Analysis:** The collected data is processed and numerical measures such as mean, median and standard deviation are computed. The data along with these measures are then represented in the form of graphs, charts, tables etc.
- **Interpretation:** Conclusions are made from the data analysis and a plan of action is formulated based on the results.

The control charts thus can be used as an estimating device from which we can estimate the mean, standard deviation, fraction conforming and other such parameters. These parameters can then be used in process capability studies. Control charts can be classified into two types, variable and attribute control charts. Any quality characteristic that can be expressed in a continuous scale of measurement is called a variable. Examples are diameter of a pin, size of a slot, form of a planar surface. Many quality characteristics cannot be measured on a
continuous scale and are often a measure of conformance. These are binary measures determining if a part possesses certain characteristic or not. These are called attributes. The control charts can indicate out of control situation either when one or more points fall beyond the control limits or when a sequential set of points exhibit a trend or a non random behavior. This is shown in Fig. 9.2.

In Fig. 9.2 the process depicted by the first chart is out of control because, there are two points that are outside the control limits. In Fig. 9.2, the process depicted by the second chart is out of control because, the points tend to fall below the center line most of the time even if they are within the control limits. For the process to be in statistical control, all the plotted points should be essentially random. The concept of control charts and hypothesis testing are closely related. For example, if \( \bar{x} \) plots between the specified control limits, it can be said that the process mean is in control or the null hypothesis is satisfied. In case the mean plots outside of either of the control limits, we reject the null hypothesis and choose the alternate hypothesis. In other words, the mean \( \bar{x} \) is deemed to be out of control. In this perspective, the control chart is a tool for hypothesis testing, the hypothesis being the process in statistical control.
9.3.1 Shewhart Control Charts:

The following sections describe a variety of control charts that controls processes based on parameters such as mean, median, standard deviation and range, usually called the Shewhart control charts. For simplicity, a normal behavior is assumed for all the quality characteristics. Even if some of the individual characteristics differ from a normal behavior, the characteristic analyzed is usually a function of these individual characteristics and hence exhibit an approximately normal behavior due to the effects of the Central Limit Theorem [27, 30].

9.3.1.1 $\bar{x}$ and R Chart:

In usual manufacturing practice, the mean $\mu$ and standard deviation $\sigma$ are not known. Therefore these parameters are estimated from preliminary samples or subgroups when the process is deemed to be in statistical control. The usual practice is to have approximately 20 to 25 samples of 3-8 observations each and estimate the mean and standard deviation from these samples. Typically, these small sample sizes are a result of rational subgroups because of the high cost and effort involved with sampling and inspection as explained by Montgomery [32]. Suppose $\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m$ are the averages of $m$ samples respectively. Then the best estimator of the process mean, $\mu$, is the grand average of these sample averages.

$$\bar{x} = \frac{\bar{x}_1 + \bar{x}_2 + \ldots + \bar{x}_m}{m}$$
Thus, \( \bar{x} \) is used as the centerline of the \( \bar{x} \) chart. The next step is to construct the control limits. We can estimate the process standard deviation either from the sample standard deviations or the range of each of the samples. The range of a sample is the difference between the largest and the smallest values in the sample.

\[
R = x_{\text{max}} - x_{\text{min}}
\]

The average range of the process is then computed as

\[
\bar{R} = \frac{R_1 + R_2 + \ldots + R_m}{m}
\]

The formulas for the control limits are then given by [30]

**\( \bar{x} \) Chart:**

Upper Control Limit = \( \bar{x} + A_2 \bar{R} \)

Center Line = \( \bar{x} \)

Lower Control Limit = \( \bar{x} - A_2 \bar{R} \)

The values of the constants \( A_2 \) are dependent on the number of observations in the sample, ‘n’ and are tabulated in Table. for different values of n.

<table>
<thead>
<tr>
<th>n</th>
<th>( A_2 )</th>
<th>n</th>
<th>( A_2 )</th>
<th>n</th>
<th>( A_2 )</th>
<th>n</th>
<th>( A_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1.880</td>
<td>8</td>
<td>0.373</td>
<td>14</td>
<td>0.235</td>
<td>20</td>
<td>0.180</td>
</tr>
<tr>
<td>3</td>
<td>1.023</td>
<td>9</td>
<td>0.337</td>
<td>15</td>
<td>0.223</td>
<td>21</td>
<td>0.173</td>
</tr>
<tr>
<td>4</td>
<td>0.729</td>
<td>10</td>
<td>0.308</td>
<td>16</td>
<td>0.212</td>
<td>22</td>
<td>0.167</td>
</tr>
<tr>
<td>5</td>
<td>0.577</td>
<td>11</td>
<td>0.285</td>
<td>17</td>
<td>0.203</td>
<td>23</td>
<td>0.162</td>
</tr>
<tr>
<td>6</td>
<td>0.483</td>
<td>12</td>
<td>0.266</td>
<td>18</td>
<td>0.194</td>
<td>24</td>
<td>0.157</td>
</tr>
<tr>
<td>7</td>
<td>0.419</td>
<td>13</td>
<td>0.249</td>
<td>19</td>
<td>0.187</td>
<td>25</td>
<td>0.153</td>
</tr>
</tbody>
</table>

Table 9.1. Values of constants \( A_2 \) [30]
R Chart:

Upper Control Limit = $D_4 \bar{R}$

Center Line = $\bar{R}$

Lower Control Limit = $D_3 \bar{R}$

The values of the constants $D_4$ and $D_3$ are again dependant on the number of observations in the sample, ‘n’ and are tabulated in Table for different values of n.

<table>
<thead>
<tr>
<th>n</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>n</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>n</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>n</th>
<th>$D_3$</th>
<th>$D_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0</td>
<td>3.267</td>
<td>8</td>
<td>0.136</td>
<td>1.864</td>
<td>14</td>
<td>0.328</td>
<td>1.672</td>
<td>20</td>
<td>0.451</td>
<td>1.585</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>2.575</td>
<td>9</td>
<td>0.184</td>
<td>1.816</td>
<td>15</td>
<td>0.347</td>
<td>1.653</td>
<td>21</td>
<td>0.425</td>
<td>1.575</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>2.282</td>
<td>10</td>
<td>0.223</td>
<td>1.777</td>
<td>16</td>
<td>0.363</td>
<td>1.637</td>
<td>22</td>
<td>0.434</td>
<td>1.566</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>2.115</td>
<td>11</td>
<td>0.256</td>
<td>1.744</td>
<td>17</td>
<td>0.378</td>
<td>1.622</td>
<td>23</td>
<td>0.443</td>
<td>1.557</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>2.004</td>
<td>12</td>
<td>0.283</td>
<td>1.717</td>
<td>18</td>
<td>0.391</td>
<td>1.608</td>
<td>24</td>
<td>0.451</td>
<td>1.548</td>
</tr>
<tr>
<td>7</td>
<td>0.07</td>
<td>1.924</td>
<td>13</td>
<td>0.307</td>
<td>1.693</td>
<td>19</td>
<td>0.403</td>
<td>1.597</td>
<td>25</td>
<td>0.459</td>
<td>1.541</td>
</tr>
</tbody>
</table>

Table 9.2. Values of constants $D_3$ and $D_4$ [30]

9.3.1.2 $\bar{R}$ and s Chart:

The $\bar{R}$ and R charts are used widely for all types of process control. However, it is sometimes desirable to control the process through the standard deviation of the process directly instead of the range. For such cases, we have $\bar{R}$ and s charts, where s is the sample standard deviation. These charts are usually used when the sample size ‘n’ is moderately large since the range method is less efficient for large sample sizes and when the sample size is variable. Setting up $\bar{R}$ and s chart follows that exactly same procedure except that the range is replaced.
by the standard deviation. The formulas for computing the limits in $\bar{X}$ and $s$ charts are described below:

**$\bar{X}$ Chart:**

Upper Control Limit = $\bar{X} + A_2s$

Center Line = $\bar{X}$

Lower Control Limit = $\bar{X} - A_2s$

**$S$ Chart:**

Upper Control Limit = $B_4s$

Center Line = $\bar{s}$

Lower Control Limit = $B_3s$

$$s = \frac{1}{m} \sum_{i=1}^{m} S_i$$

Where $m$ = number of samples and $s = \sqrt{\frac{(s_2 - s_1)^2}{m-1}}$

For more reference on the details of variable sample sizes, one can refer [30, 36, 37].

The Shewhart charts are used to evaluate the performance or capability of the process. Sometimes, the tolerance specifications on a dimension are used as the upper and lower control limits. The fraction of non-conformance can be computed by assuming a normal behavior and using the estimates derived from the preliminary samples. For example, consider the case of linear size of a cylindrical bar. The bar has a tolerance specification of $10 \pm 0.05$ inches. The
preliminary samples yield a mean size of 10.005 with a standard deviation of 0.009. The fraction of non conforming parts is then evaluated as follows:

\[ p = \Phi\left(\frac{9.95 - 10.005}{0.009}\right) + 1 - \Phi\left(\frac{10.05 - 10.005}{0.009}\right) \]

\[ p = \Phi(-6.111) + 1 - \Phi(5) \]

\[ p = 0 + 1 - 0.99998 \approx 0.00002, \text{ or } 20 \text{ parts per million of non conforming parts.} \]

Another way of measuring process capability is in terms of the Process Capability Ratio (PCR), also called Cp. The value for Cp is given by the following formula:

\[ Cp = \frac{USL - LSL}{6\sigma} \]

The 6\(\sigma\) is usually taken as the limits of the process spread for defining the process capability. For the above example, the value of Cp is

\[ Cp = \frac{(10.05 - 9.95)}{6 \times 0.009} = 1.852 \]

The greater is the value for Cp, the lesser the number of non conforming parts produce by the process. The PCR can also be interpreted by its reciprocal. The quantity, percentage of specification band is given by,

\[ P = \frac{1}{Cp} \times 100 \% \]

\[ P = \frac{1}{1.852} \times 100 \% \]

So, for the above example, the value \(P = 54\%\). In other words, the process uses 54\% of the specification band.

The initial set of preliminary control limits have to be tested and readjusted for stability and ensured that the limits reflect the state when process is
in statistical control. Once a set of reliable controls are established, we can use these control charts for future production control. It should also be understood here that there is no connection between the control limits on the $\bar{X}$ and R charts and the tolerance specifications on the part being inspected. The control limits are determined by the natural limits of variation of the process and these are measured by the mean and standard deviation of the samples. This is due to the inherent variation of the process and the control limits are specified as $3\sigma$ above and below the mean. Any part produced by the process should naturally fall within this zone if the process is in statistical control. The specification limits on the other hand, is determined externally. They are set by the manufacturing and design engineers based on the functionality of the part while economic and other business factors also play a role. The control limits or the natural limits of the process variation do not have any statistical or mathematical relationship to the specification limits. [30, 36] A part that falls below the control limits of the process but still within the specification limits can still serve the functional needs of the product. However, such a situation reflects instability in the manufacturing process and demands investigation. A couple of rare outliers may not result in any damage, but a high frequency of such outliers is detrimental to the process and the business in the long run.

9.4 Statistical Process Control Using Inspection Maps:

In the last section tradition methods of Statistical Process Control, in particular the Shewhart charts were introduced. These charts are used to control
the process by analyzing the measurements on sample parts. But one problem with these charts is that for a simple part such as a cylindrical bar with a planar end face which has a size tolerance on the end face, the Shewhart charts plot only the variations in size. The charts do not reflect the performance of the process with respect to the orientation or the form tolerances produced by the process. In earlier chapters on Tolerance Maps, it was explained how T-Maps represent all the variations allowed by a single tolerance. For example, the T-Map for size tolerance on the cylindrical bar can represent the variations in size, orientation and form in a single T-Map. But, at least three Shewhart charts would be needed to investigate this. The following sections explain the use of T-Maps and i-Maps for SPC.

9.4.1 SPC Using I-Maps for Size Tolerance on End Face of Cylindrical Bar:

The algorithm for computing the i-Map is described in section. Suppose, if there are ‘m’ samples each of sample size ‘n’. Similar to computing the mean for the SPC charts, we compute the i-Map for each of observations in the m samples. The i-Maps are a set of coordinates (p’, q’, s) within the coordinate system defined by the T-Map for the tolerance type. The next step is to compute the centroid C for each of these ‘m’ samples. Once, the centroids are computed, the standard deviation σ for each of these coordinates (p’, q’, s) is computed. The control limits are then derived from the centroids C_i and the standard deviations σ_i. The i-Maps and the T-Maps can then be used as a tool for Statistical Process Control.
Control by plotting i-Maps for future inspection samples. A detailed description of the procedure is given below.

9.4.1.1 Construction of the T-Map:

The first step in SPC of size tolerance on the cylindrical bar is to construct the T-Map. This T-Map is the envelope within which all the i-Maps corresponding to the inspected parts have to lie. Any i-Map that lies outside the limits of the T-Map represents a part that is out of conformance with the specifications. The construction procedure and the geometric details of T-Maps were described in Section.

9.4.1.2 Construction of Control Limits:

Once the T-Map is constructed, the next step is to define the control limits for the process. The control limits are determined from preliminary measurements when the process is deemed to be in statistical control. Consider a case where we have say 20 samples, each of sample size 15. i-Maps are constructed for all the observations in each of the sample using the procedure described in chapter 8. Now for each sample the algorithm for defining the control limits is described by the following procedure.

- The i-map is composed of three coordinates \((p'_c, q'_c, s_c)\). The three coordinates each represent the three types of variation possible, i.e., two orientational variations and one size variation. So we get the mean variation in each of these categories from the i-map.
• Compute the standard deviation of the i-maps for all the samples. Thus, the standard deviation $\sigma$ is actually a triad, ($\sigma_p', \sigma_q', \sigma_s$). This represents the variation of the observations in each of the categories.

• Now, construct a rectangular parallelopiped centered on the centroid $C$.

• The length, width and height of the rectangular parallelopiped are $3 \times (\sigma_p', \sigma_q', \sigma_s)$ respectively.

• The value $3 \times (\sigma_p', \sigma_q', \sigma_s)$ denotes the $3\sigma$ spread of the values on either side of the mean, i.e., the centroid.

• This rectangular parallelopiped centered on the centroid is the control limits defined by the first sample. Thus we have a set of control limits that are centered on the mean and spread $3\sigma$ on either side of the mean. This represents the natural variation of the process.

• Fig. 9.3 and Fig 9.4 illustrates two control limits derived from two different samples within the T-Map.
Fig. 9.3. i-Maps within the sample control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for size tolerance on a planar circular face – Sample 1.

Fig. 9.4. i-Maps within the sample control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for a size tolerance on planar circular face – Sample 2.
9.4.1.3 Fitting of i-Maps within the T-Map and Control Limits:

For more accuracy and reliability, a large number of preliminary samples are inspected and all the i-Maps are transferred into the T-Map when the process is in statistical control. The overall mean and standard deviation are computed from these data. The overall mean \( \mathbf{\bar{C}} \) is the centroid of all the i-Maps corresponding to all observations. The overall standard deviation \( \mathbf{\bar{SD}} \) is the standard deviation triad computed from all the i-Maps. The control limits are computed using the procedure described in section 9.4.1.2. Thus, an SPC chart using i-Maps and T-Map is constructed with the mean and \( 3\sigma \) spread from the observation samples. A sample SPC chart using this method is illustrated in Fig. 9.5.

![Diagram of i-Maps within the T-Map and control limits](image)

Fig. 9.5. i-Maps within the overall control limits defined by the rectangular parallelepiped and the tolerance limits defined by the T-Map for size tolerance on a planar circular face.

This method of Statistical Process Control is more powerful because, the different variations pertaining to the part, say orientation and size are represented
in a single chart. Also, this model dissects the variations and splits the assignable causes that would affect each of the variations separately. Compared to this, the traditional Shewhart charts give a blind representation. In other words, they just say that an assignable variation is present, but do not reflect the effect of the assignable cause to the different categories of variations. For example, in a tradition chart, the differences in orientation are coupled to the size variation and a same degree of variation in all these categories are represented as the same point. Whereas, using i-Maps, the variations in orientation and size are analyzed and controlled independently. Since, the i-Maps are also shown to be the most accurate representation of the actual parts, this is a powerful tool that not only evaluates the features accurately, but also categorizes the assignable causes according to different variations.

9.4.1.4 Process Capability Index:

Section 9.3.1.2 described the PCR metrics associated with the traditional Shewhart charts. Similarly, the percentage of specification band used can be computed using the i-Maps. The control limits defined in section 9.4.1.4 is a simple rectangular parallelepiped defined by the standard deviation of the process. The volume of the parent dicone could be computed using the formula [8]

\[ V = \frac{\pi}{3} t^3 \]

Where ‘t’ is the size tolerance on the feature. The volume of the inner rectangular parallelepiped representing the control limits is also be computed. The ratio of the
two volumes is the Percentage of the Specification Band (PSB) that the process uses.

\[ \text{PSB} = \frac{V_{\text{cube} \text{left}}}{V_{\text{T-Map}}} \times 100\% \]

The Process Capability Ratio is then computed as

\[ \text{PCR} = \frac{1}{(\text{PSB}/100)} \]

Based on this model, any i-Map that falls within the control limits is considered to be normal and anything outside the control limits should warrant an investigation. An i-Map which is outside the control limit, but still within the T-Map can still pass the inspection, but indicates some problem with the process. This is because the rectangular parallelepiped reflects the natural variation of the process and the process is expected to produce a part that is within these limits. Ideally, the rectangular parallelepiped should be centered on the origin of the T-Map and should be as small as possible. This indicates a case when the process is precisely producing parts around the required dimension with minimum variation. The position of the rectangular parallelepiped within the T-Map can also indicate problem such as alignment errors within the process. The investigation of the assignable causes can proceed as usual from this stage, just that the quality team is equipped with much more information using this model. For additional tolerances such as orientation that are applied on the face, the procedure for generating the i-Maps and control limits remain the same. The only difference is
the T-Maps, which are now truncated in respective directions. All the control
limits and the i-Maps should lie within this truncated T-Map.

9.4.2 SPC Using I-Maps for Size Tolerance on End Face of Rectangular Bar:

The procedure described in section 9.4.1 for round faces can be followed
for size tolerances on rectangular face as well. The only difference is in the
construction of the T-Map and the i-Maps, which were described in earlier
chapters for this specific case. The SPC chart using T-Maps and i-Maps for size
tolerance on rectangular faces is illustrated in Fig. 9.6.

![Control Limits](image)

**Fig. 9.6.** i-Maps within the overall control limits defined by the rectangular parallelopiped and the tolerance limits defined by the T-Map for size tolerance on a planar rectangular face.

The volume of the T-Map is given by the formula

\[ V_{T-Map} = \frac{2}{3} \delta t^2 \]

Where, \( \delta = \frac{d_y}{d_x} \), which comes from the geometry of the rectangular bar. The PSB
is then computed by the formula
The Process Capability Ratio is then computed as

\[ PCR = \frac{1}{\left(\frac{PSB}{100}\right)} \]

Which are the same as the ones for the circular face.

9.4.3 Control Limits by Offsetting the T-Map:

The i-Maps for the size tolerance on circular and rectangular bars were computed from the individual points representing the inspection samples. The control limits were defined by the rectangular parallelepiped whose dimensions are determined from the standard deviation of the observations. However, this method of determining control limits does not take into account the correlation between the orientation and size variations. Any variation in size also affects the extent to which the feature can have variations in its orientation. To counter this, the control limits can be defined by an offset of the parent T-Map. The revised control limits can then be a fixed percentage of the extent of variation allowed in each axes of the T-Map. For example, for a circular bar, the dimensions of the T-Map along the \( p', q' \) and \( s \) axes can be multiplied by a factor such as 0.75 or 0.9 to get the basis points of the control limits. Thus the control limits can be defined as a shape geometrically similar to the parent T-Map. Since it is geometrically similar to the T-Map, the variations in each of the axes are also correlated. An illustration of control limits geometrically similar to a T-Map is shown in Fig. 9.7. In this case, there is a degree of coupling between the variations in size and
orientation. As the i-Maps move towards the control limits of one axis, the extent of variation allowed in the other axes is reduced correspondingly.

Fig. 9.7  SPC control limits defined by offsetting the parent T-Map

9.4.4 SPC Using i-Maps for Orientation Tolerance on End Face of Circular and Rectangular Bar:

i-Map for the orientation tolerance is a straight line segment and is overlaid on another straight line segment, which is the T-Map. The construction procedure for the i-Map for orientation tolerance was explained in section 8.5.2. Similar to size tolerance, the orientation tolerance can also be controlled statistically using i-Maps. i-Maps are constructed for all the observations in the set of preliminary samples. This gives a value of width of the orientation tolerance used up by the part for the i-Maps, each representing a unique part in the sample observations. The mean and the standard deviation of all the widths are computed
from these i-Maps. Once the standard deviation $\sigma$ is computed, the control limits are plotted as

$$\text{UCL} = \mu/2 + 3\sigma \quad \text{LCL} = \mu/2 - 3\sigma$$

The control limits are placed on either side of the origin of the T-Map. This is shown in Fig. 9.7.

![Diagram showing Tolerance Limits and control limits](image)

Fig. 9.8. Overall control limits defined by $\mu$ and $\sigma$ of the samples and the tolerance limits defined by the T-Map for orientation tolerance on a planar rectangular face.

**9.4.5 Percentage Non-Conformance:**

In the above discussions on fixing the control limits for the i-Maps and control charts, it has been assumed that the distribution is Gaussian. It is then possible to find out the percentage of parts produced by the process that are non-conforming from the formulas underlying Normal distribution.

$$p = P\{x < \text{LTL}\} + P\{x > \text{UTL}\},$$
where LTL and UTL are Upper Tolerance Limit and Lower Tolerance Limits respectively.

\[ p = \Phi\left(\frac{\text{LTL} - \mu}{\sigma}\right) + 1 - \Phi\left(\frac{\text{UTL} - \mu}{\sigma}\right) \]

The value \( p \times 100\% \) is the percentage of non-conforming parts produced by the current process. The advantage of using the method of i-Maps for computing the percentage of non-conforming parts is that, it is possible to compute the percentage of non-conforming parts produced for each type of variation in the applied tolerance zone. For example, for a size tolerance on a circular face, it is possible to find the percentage of parts that are non-conforming due to orientation variations as well as size variations separately. The total number of non-conforming parts is the sum of all these variations. By these methods, it is possible to determine the sensitivity of the process with respect to these individual types of variations. Hence, those assignable causes that pertain to least occurring variations need not be investigated, thus saving time. For example, a manufacturing process may produce parts that have little size variations, but produce parts whose orientational variations are large. In that case, those assignable causes that are known to cause size variations need not be investigated. Time and effort can be focused purely for those that are known to cause orientational variations.

**9.5 Principal Component Analysis Using i-Maps:**

Principal Component Analysis (PCA) is a statistical technique for finding patterns in complex data. It is a technique based on the eigen values and eigen
vectors of the covariance matrix of the sample data. It is used in different fields such as image and signal processing and could be applied to find the relationship embedded within the collected data. It is a way of identifying patterns and expressing data in such a way that the relationship between the different components in the data is highlighted. It is a procedure that removes redundancy in the collected data when the variables are correlated with each other. The objective is to reduce the large number of observed points or data into a smaller number of principal components that will account for most of the variation or spread in the observation. PCA offers a convenient way to control the trade-off between losing information and simplifying the complexity of the problem at hand. A principal component can be defined as a linear combination of optimally-weighted observations. [81]. The first principal component accounts for the maximum variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. PCA can thus be defined mathematically as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on. PCA is theoretically the optimum transform for given data in least squares terms. [80]. Another advantage of using PCA is that complex data could be simplified without much loss of data. The following sections illustrate the process of PCA on a simple data set.
9.5.1 Computing the Principal Components:

Let us assume a simple two dimensional data for illustration purposes as follows.

<table>
<thead>
<tr>
<th>X</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>4</th>
<th>4.5</th>
<th>5.25</th>
<th>6</th>
<th>7.15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>3</td>
<td>3.25</td>
<td>5</td>
<td>6</td>
<td>6.35</td>
<td>7.15</td>
<td>8.25</td>
<td>9.3</td>
</tr>
</tbody>
</table>

Table 9.3. Data for Principal Component Analysis

The above data is first organized such that the mean of the data is subtracted from each of the data points. This produces a data set whose mean value is zero. For the data shown above, the mean of x and y components are 3.925 and 6.0375. Subtracting these values from the original set of data, the new set of data is shown below in table 9.4.

<table>
<thead>
<tr>
<th>X</th>
<th>-2.925</th>
<th>-2.425</th>
<th>-1.925</th>
<th>0.075</th>
<th>0.575</th>
<th>1.325</th>
<th>2.075</th>
<th>3.225</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>-3.0375</td>
<td>-2.7875</td>
<td>-1.0375</td>
<td>-0.0375</td>
<td>0.3125</td>
<td>1.1125</td>
<td>2.2125</td>
<td>3.2625</td>
</tr>
</tbody>
</table>

Table 9.4. Mean subtracted data for Principal Component Analysis

The next step is to calculate the covariance matrix for the above data. The covariance matrix is computed as

```
4.9914  4.9150
4.9150  4.9927
```

Table 9.5. Covariance Matrix for the mean subtracted data

The next step is to compute the eigenvectors and eigen values for the covariance matrix. For the above covariance matrix, the eigen values are (0.0771, 9.9071).
The corresponding eigenvectors for these eigen values are (-0.7072, 0.7071) and (0.7071, 0.7072). The eigenvectors of the covariance matrix along with the mean subtracted data are plotted in Fig. 9.8.

![Sample Data](image)

**Fig. 9.9.** Eigen vectors of the covariance matrix and mean subtracted data plotted together

The scatter of data in Fig. 9.8 shows a definite pattern. The variables X and Y are positively correlated and the clear pattern is shown in the figure. The first principal component is the largest eigen vector and is plotted along with the data. It is clearly seen that the first principal component goes through the points very similar to a line of best fit. In fact this is the least squares line for the given data. In other words, the first principal component summarizes the relationship between X and Y. So, instead of having a large sample of data, the relationship can simply be represented by the principal component. The scatter of data about
the principal component is almost negligible and can be neglected. If that is also
to be taken into account, then the second principal component describes the
scatter of data about the first principal component. Thus, the relationship defined
by the sample data between the X and Y variables can essentially be summarized
by these two principal component vectors. Since the eigenvectors and hence the
principal components are orthogonal to each other, the entire dataset can now be
represented in the new coordinate system where the new X and Y axes are defined
by the two principal component vectors. To do this, the entire data set is rotated
about the first principal component.

9.5.2 Principal Component Analysis Using i-Maps:

Similar to the procedure described in the previous section, a PCA analysis
can be performed with the i-Maps. Since i-Maps are nothing but a representation
of actual parts in an Euclidean space defined by the T-Map, we can represent the
data by resolving them into principal component vectors. This gives us the
relative degree of variation in each of the axes of the T-Map. For example, for the
size tolerance on a round bar, the \((p', q', s)\) axes represent the x orientation, y
orientation and the size of the bar respectively. Suppose in a batch, 10 parts are
inspected. We can fit a substitute feature for each of these 10 parts in the sample
using the normative guidelines proposed in the previous sections. From these, the
points representing each of these ten parts are fitted directly into the T-Map. From
this set of \((p_i', q_i', s_i)\) data, we can do a principal component analysis. The
covariance matrix is then a 3 X 3 matrix and we have 3 eigen values and eigen
vectors corresponding to these eigen values. These eigen vectors in the descending order of their corresponding eigen values give the principal component vectors of the observed data. Thus, a principal component analysis of each batch of inspection can be performed. The spread of the data about these principal vectors gives the degree of variation in each of the axes and also identifies any correlation between the (p’, q’, s) variations. Similarly, a PCA can be performed for a series of batches and the principal components of this analysis gives the overall correlation and variation of the process.

9.6 Convex Hulls for i-Maps:

The previous sections elaborated the method of computing i-Maps from the centroid of the observations within an inspection batch. One disadvantage of this method of computing i-Maps from the centroid of the data is that it only gives information about the mean variation of the particular batch. Any information contained in the sample about the spread of data is lost. For example, in the inspection of say, ten batches, the last few batches may have a larger variance in the data but spread about the same centroid. In this case, it is impossible to distinguish between the batches with a large variance from the ones with smaller variance with just the mean. Worse, two batches can have different standard deviation but the same mean. This leads to information loss and the potential for identifying flaws in the process is reduced. Another method of analyzing the variation is to compute the convex hull of all the points that represent the substitute features in a batch of parts. The convex hull represents the location of
all the samples of a batch as well as identifies the spread of data within each batch. In this case, the i-Map is not just a point, but a region of points that represent all the possible variations within a particular batch.

One disadvantage of this method of computing i-Maps as the convex hull is the complexity in the computation of the convex hull. Also, convex hulls depend only on the points present on its boundary. In other words, the points that lie on the interior of the convex hull have no effect on the size and shape of the convex hull. This way, there is no real statistical meaning that is associated with the i-Map computed from the convex hull. The size and shape of the convex hull is also influenced by the outliers in the data. One or two outliers can drastically alter the i-Map computed. This may lead to situations where false alarms about the process are raised. But convex hulls computed from the points in a batch gives the information regarding both the location and the spread of data. It is also a representation of the worst case scenario in the batch, since the convex hull is defined by the points that are farthest from the centroid of the points. A series of convex hulls could be computed for series of inspection batches and plotted inside a T-Map. This gives rich information about the actual spread and capability of the process.
CHAPTER 10

CONCLUSION

In this thesis a new mathematical model was developed for representing tolerances in the inspection stage. Central to the new math model is the Tolerance-Map (T-Map) which represents the variations of a tolerated feature as points in a hypothetical, Euclidian space. Algorithms used in current CMM software are not in complete conformance with the principles laid down by the ASME Y14.5 standard. A survey of manual inspection practices was made to understand the interpretation of the tolerance standard at the inspection stage. A case study was performed on commercial CMM software with parts of known variations. The CMM data points were operated with different algorithms based on least squares and Chebyshev algorithms for feature fitting. The results were compared to the results predicted by the CMM software and the algorithms used in the CMM software were reverse engineered. Based on the definition of the tolerance classes provided by the ASME Y14.5 standard and the interpretation of manual inspection practices, correct feature fitting algorithms were formulated for tolerance types that apply to planar features such as size, orientation, form and position.

Based on the algorithms developed and the concept of T-Maps, the i-Maps were developed. The i-Map is a hypothetical set of one or more points that represents the actual manufactured part in the inspection stage. The i-Maps are established in the same coordinate frame as the T-Map. The i-Maps for the size
tolerance on circular and rectangular features are points and they are place inside the dicones and dipyramids, which are the T-Maps for the respective cases. The i-Map for orientation tolerance on a planar feature is a line. This line is placed inside a T-Map for orientation, which is again a line. The i-Maps for form tolerances on planar features are subsets of similar shape to the parent T-Map.

Finally, a methodology is proposed for using these i-Maps in the Statistical control of the process. The i-Maps are plotted directly within the parent T-Map. In this method, the region within the T-Map itself acts as a control chart. The control limits of the i-Map control chart are computed using the mean and standard deviation computed from the individual samples. Formulas for calculating SPC metrics such as percentage non-conformance and process capability index are also described.

10.1 Future Work

Standardization of feature fitting algorithms was done for tolerances that applied to planar features. This can be extended to other features such as axis features, run-out and profile tolerances. For these tolerance and feature types, the inspection practices have to be first studied and an interpretation of the substitute feature consistent with the standard has to be determined. Once this is done, the i-Maps for these tolerance types have to be developed consistent with the principles on which their T-Maps have been developed.

In addition to the development of T-Maps, modeling advanced quality control techniques such as the effect of deviation from normal behavior, variable
sample sizes and other advanced control chart types have to be explored using the i-Maps developed in this thesis and for the future ones. All these implementation have to be then implemented into on integrated CMM inspection and quality control Test bed.
References:


University, Tempe, AZ, USA, April 10-12, 2005 (ed. J. Davidson), Springer Netherlands, pp. 45-54.


243


MATLAB IMPLEMENTATION OF FITTING ALGORITHMS

A GUI application for fitting substitute features for different tolerance types and different feature types was created. The input to the program is in terms of ‘.dat’ files. Data points from the CMM for each feature and datums have to be fed in a separate ‘.dat’ file to the program. The data points are stored in the file in ‘csv’ format. For FOS features, the data is typically split into a number of subsets. The data for FOS is still supplied in a csv format; however, the subsets are separated by the delimiter (1000, 1000, 1000). So, whenever the program encounters the line (1000, 1000, 1000), it understands that the next line is the start of a new subset. The tolerances evaluated from the data are displayed right in the GUI application and the results are written to a ‘.txt’ file. Some snapshots of the GUI application are shown in Figs A1 to A4.

![Fig. A1 Opening dialog box of the MATLAB Fitting Module](image-url)

248
Fig. A2 Fitting for size tolerance

Fig A3 Fitting for perpendicularity tolerance on planar feature
Fig A4 Fitting for flatness tolerance

Fig A5 Data format for planar feature data
Functions for implementing algorithms and computing the geometric entities are coded in the following files. Centroid.m, distance.m, distance_line.m, lsline.m, lsplane.m, plane_points.m, TSfit_para.m, TSfit_perp.m, TSfit_ang.m, TSfit_size.m are the implementation files which are used to compute the one and two sided planes and lines from the feature data. For the algorithms that are implemented in these files, the reader is referred to the chapters 6 and 7.
T-MAP LIBRARY

The T-Map universal library is a C++/ACIS code that can be used to generate solid models of different T-Maps developed at DAL. There is a primary T-Map class that defines all the attributes of a T-Map object. Individual T-Maps for different tolerance types and feature types are created using functions for each available combination. A sample of functions for creating T-Maps for different tolerance and feature types are listed below.

```cpp
class cASU_TMap
{
    protected:
    ENTITY* m_solidmodel;
    char* m_TMap_Name; // Name for the T-Map
    int m_TMap_ID;    // Integer ID for the T-Map
    int m_TMap_Dimension; // Dimensionality of the T-Map
    double m_Length;  // Length of the T-Map, if applicable
    double m_Area;    // Area of the T-Map, if applicable
    double m_Volume;  // Volume of the T-Map, if applicable
    double m_Content; // Content of the T-Map, if applicable

    public:
    // Accessor and Modifier functions for the T-Map name
    void setName(char* name);
    char* getName();

    // Accessor and Modifier functions for the T-Map ID
    void setID(int ID);
    intgetID();

    // Accessor and Modifier functions for the T-Map Dimensions
    void setDimension(int dimension);
    int getDimension();

    // Accessor and Modifier functions for the T-Map Length
    void setLength(double length);
    double getLength();

    // Accessor and Modifier functions for the T-Map Area
    void setArea(double area);
    double getArea();
};
```
// Accessor and Modifier functions for the T-Map Volume
void setVolume(double volume);
double getVolume();

// Accessor and Modifier functions for the T-Map Content
void setContent(double content);
double getContent();

// Function for creating a T-Map for Planar Circular Face
ENTITY* createTMap_PlanarCircularFace(double sizeTolerance,
double parallelism, double perpendicularityDatumA, double
perpendicularityDatumB);

// Function for creating a T-Map for Planar Rectangular Face
ENTITY* createTMap_PlanarRectangularFace(double length,
double width, double sizeTolerance, double parallelism, double
perpendicularityDatumA, double perpendicularityDatumB);

// Function for creating a T-Map for Planar Triangular Face
ENTITY* createTMap_PlanarTriangularFace(double length,
double width, double sizeTolerance);

// Function for creating a T-Map for a FOS Feature
ENTITY* createTMap_CircularFaceDiameter(double diameter,
double sizeTolerance, double positionTolerance, double perpTolerance, double heightTolZone);

// Function for creating T-Map for size tolerance on cylindrical profile for an axis
ENTITY* createTMap_CylindricalProfileSize(double sizeTolerance);

// Function for creating T-Map for circular runout tolerance on cylindrical profile for an axis
ENTITY* createTMap_CylindricalProfileCircRunout(double runoutTolerance);

// Function for creating T-Map for circular runout tolerance on conical profile for an axis
ENTITY* createTMap_ConicalProfileCircRunout(double runoutTolerance);

// Function for creating T-Map for circularity tolerance on cylindrical profile for an axis
ENTITY* createTMap_CylindricalProfileCircularity(double circularity);

// Function for creating T-Map for size and circular runout tolerance on cylindrical profile for an axis
ENTITY* createTMap_CylindricalSizeRunout(double sizeTolerance, double runoutTolerance);
// Function for creating T-Map for size circularity and
circular runout on cylindrical profile for an axis
ENTITY* createTMap_CylindricalSizeRunoutCircularity(double
sizeTolerance, double runoutTolerance, double circularity);

// Function for creating T-Map for size, circular runout
for planar circular face
ENTITY* createTMap_PlanarFaceSizeCircRunout(double
sizeTolerance, double runoutTolerance);

It is clear from the code that every function takes in the parameters
required for defining a T-Map and returns a solid model of the T-Map of ENTITY
pointer type. The ENTITY pointer is an ACIS data type and can be used by
including the necessary header files. A Universal library application is developed
using this T-Map library where the user can interactively enter the T-Map
parameters in a GUI and the T-Map is displayed on the screen. The user also has
the option of saving the T-Map model in .sat format for applications that require a
solid model of the T-Map.
Fig. B1 GUI input for creating T-Map for size tolerance on a planar rectangular face

Fig. B2 GUI input for creating T-Map for size tolerance on a circular FOS feature
Fig B3 Displaying the T-Map for size tolerance on a planar circular face

3. Rendering i-Maps:

The i-Maps are written into a text file in the following format.

```
8
0.0005 0.49208 0.2164
-0.1005 0.49208 0.2164
0.2005 0.47208 0.2264
-0.3005 -0.42208 -0.2164
-0.1205 0.19908 -0.3164
0.2005 -0.17208 0.2264
-0.3005 0.42208 -0.2164
-0.1205 0.19908 -0.1164
```

The first line gives the number of samples or the number of i-Maps. The following lines give the coordinates of the i-Maps each separated by a space. The text file should be named “size.txt” and should be stored in the working folder of the i-Map application. The data in this file or the set of i-Maps are computed in the MATLAB module and written to the file. The user has to invoke the i-Map
menu item to activate the i-Map module. This is shown in Fig. A10. Figs A12 and A13 show i-Maps inside T-Maps for size tolerance on circular and rectangular planar faces respectively.

Fig B4 Selecting the i-Map menu item

Once the i-Map menu item is invoked and the feature type is chose, the GUI prompts the user to enter the parameters of the T-Map to create a model of the T-Map first as shown in Fig. A11.
Fig B5 Creating T-Map for size tolerance on a planar rectangular face

Fig B6 Displaying i-Maps inside T-Map for size tolerance on a planar circular face
Fig B7 Displaying i-Maps inside T-Map for size tolerance on a planar rectangular face