Characterising geometric variation in shape manufacturing with a view to process parameter feedback

Bernard F. Rolfe

November 2001
Revised June 2002

A thesis submitted for the degree of Doctor of Philosophy of The Australian National University
Declaration

This thesis contains no material which has been previously accepted for the award of any other degree or diploma in any university, institute or college, and contains no material previously published or written by another person, except where due reference is made.

Canberra, June 2002 (revised copy).

Bernard F. Rolfe
Department of Engineering
Faculty of Engineering and Information Technology
The Australian National University
Canberra ACT 0200, Australia.

Journal Papers:


Conference Papers (Refereed):


________________________________________

i


Acknowledgements

The Shape Odyssey 2001

This PhD journey was an odyssey searching for truth and knowledge, filled with many oracles and distractions.

I would like to sincerely thank my supervisors (who were my oracles in this long journey), Mick Cardew-Hall, Samer Abdallah and Geoff West for their support throughout my doctorate. I want to particularly thank Mick for his calm and collected approach to all problems (and for negotiating the final year and three months of funds for my scholarship). Also, I want to thank Samer for his energy, positivity and writing advice. Finally, I want to thank Geoff for his advice on artificial intelligence techniques.

I would like to acknowledge that this research was supported for two years by an Australian Research Council large grant, number A49532454. In addition, I received assistance from the Centre for Stamping Technology for Automotive Manufacturing Processes (STAMP) who provided the metal samples and lubricant for my metal forming experiments.

Moreover, I would like to acknowledge the support of the technical and workshop staff at the Department of Engineering, Martin Stonebridge, Bruce Mascord and Darius Krivanek. Martin proof-read my mechanical drawings while Bruce and Darius manufactured the parts of my metal forming rig. Also, I would like to acknowledge the support of Shankar Kalyanasundaram for his advice in regard to my finite element analysis of channel forming.

I would also like to acknowledge that the cartoon illustrations were created and drawn by Melinda Rolfe (my sister), while the original idea for the story was mine.

I would also like to thank Matthew Doolan and Margaret Rolfe who both assisted in proof reading many (all in my Mum’s case) of my chapters and gave good suggestions outside my supervisory panel. I also have to thank my other proof readers, Siria Natera and Mike Selinger, for their patience in reading my thesis.

Finally, I would also to acknowledge the support of the administrative staff at the Department of Engineering, particularly Sue Cameron for always keeping my weekly booking with Mick when he was the Head of Department, and Josephine Farmer who allowed me to use her colour printer.
Also, I would like to thank my fellow office mates: Tomasz Celenski, for doing the night shifts; Mick Stuckings, one of the original corner office inhabitants; Michelle McCann, who saw the back of my head for 2 years; Margaret Rossiter, the office mother; Jun Mei and Christoph Hermens, for sitting next to me; Peter McLennan, for being so efficient; and Qiu Lei, for teaching me some Mandarin. Finally, Holger Kreetz who shared an office with me and my Tiger posters during the last 6 or so months of his PhD. It was he who gave me some appreciation for Soccer and things European.

I participated in many things throughout my PhD but there are four main distractions that come to mind.

First, I would like to thank those people I worked with in PARSA (Postgraduate And Research Students' Association) during my time as Treasurer and a short time as Acting-President. Thank you Diane Hutchens for being a marvel in the office with the accounting. Thank you to the rest of the PARSA executive, Peter Chen, Sarah Miskin, Kylie Waring, Tim Dyke and Simon Niemeyer. And thank you Danne Rasmussen, the "numbers man", for your support and advice. But lastly I must thank my good friend Quinton Clements for being the other half of the "Calypso Cabinet".

Second, I would like to thank the X-pilot team which saved me from doing work during the first half of my PhD: Dave Austin, Peter Aigner, Mick Stuckings and Jason Chen.

The third distraction was the "Kicker" table which kept me sane at work during the last half of my PhD. I would like to thank the "Kicker" table group: Holger Kreetz, Jeremy Smith, Paul Kohlenbach (German Paul), Paul Stacy (Australian Paul), Shankar Kalyanasundaram, Adrian Lowe, Joe Coventry, Chris Bales and Christoph Hermens. Many of these people were also in the indoor soccer team, which also included Dan Macdonald, Tom Narongsak Kanchanasaratool, Matt Doolan, Klaus Weber, and Paul Compston (English Paul).

The final distraction, while spending long nights writing up, was to occasionally read about the Richmond Tigers march each week toward their second finals appearance in twenty years, eat 'em alive Tigers.

Finally, I want to thank my parents for their financial and culinary support throughout my PhD. Most importantly, I want to thank my partner and best friend Siria Natera for all her patience, emotional and financial support over the entirety of my PhD.

There were many other people who helped along the way and I'm sorry if I haven't included your name but I have really appreciated your support all the same.

Bernard Rolfe
(November 2001)
Abstract

The aim of this thesis is to extend automated inspection to assist in the control of manufacturing processes. This is achieved by characterising the shape variation of a manufacturing process to develop several feedback models.

One of the major problems facing manufacturing today is the decreasing number of people entering the manufacturing sector. There is a real concern that much of the process knowledge of the existing work-force will disappear. One solution is to increase the amount of automation within manufacturing to capture the current process knowledge. Traditionally, the inspection of manufactured parts has been a manual task that has been automated only in the past few decades. The control information garnered from the inspection of a part, however, is still a manual task performed by the remaining experienced operators. This thesis presents an automated solution to bridge the gap between inspection and control. Specifically, this thesis examined developing feedback information from characterising the shape variation of manufactured parts. The solution is the shape manufacturing feedback model which analyses the shape variation in the manufacturing process and provides feedback information to the control stage.

The shape manufacturing feedback model has two sections. The first section describes the shape variation from the process using a deformable model. This model decomposes a manufactured part into the principal modes of geometric variation of the process. The second section of the shape manufacturing feedback model provides feedback to the controller of the process. The principal modes of variation measured from the manufactured part are used to create two feedback models, a shape error function and an inverse model. Unlike many of the previous approaches to these models, this thesis does not require material models. The shape manufacturing feedback model, therefore, implicitly learns the material models from the output variation of the process.

The shape manufacturing feedback model was applied to forging and sheet metal forming processes. There were a number of contributions from this research. The first contribution was a novel method for measuring the shape error of forged parts. In addition, an inverse model was developed that could determine the initial state of the forging set-up when given a final shape of a forged part, to an accuracy of 80%. Another contribution was the creation of a shape error function for sheet metal formed parts. The domain of this shape error function included parts with discontinuities in shape, unlike the previous sheet metal shape error measures. The shape error function was then extended to quantify the amount of springback that was present in a formed channel. Furthermore, an inverse model was developed that mapped shape variations of the sheet metal channels to the initial settings of the forming press. The inverse model was able to identify the initial parameter settings to an accuracy of over 90%
for most of the investigated parameters. Finally, this thesis presented a shape error function that determined the accuracy of finite element sheet metal simulations. This shape error function is one of the first attempts to quantify the global geometric error between a simulation and an actual part.

The shape manufacturing feedback model is, therefore, a significant advance as it presents an automated solution to the gap between the inspection of a manufactured part and the control of a manufacturing process.
Contents

1 Introduction 1
  1.1 Inspection and control of manufacturing processes ................. 1
  1.2 Characterising geometric shape variation with a view to developing a
      shape-based process parameter feedback model ......................... 4
      1.2.1 Definitions ................................................................. 4
      1.2.2 Scope ........................................................................... 5
      1.2.3 Difficulties ................................................................. 6
      1.2.4 Approaches ................................................................. 9
  1.3 Basic model description ........................................................ 10
  1.4 Contributions ......................................................................... 13
  1.5 Outline of thesis ....................................................................... 13

2 Background 15
  2.1 Introduction ........................................................................... 15
  2.2 Tolerancing and variational modeling ..................................... 15
      2.2.1 Tolerancing approaches ............................................... 16
      2.2.2 Variational models ......................................................... 17
      2.2.3 Discussion of tolerancing ............................................... 17
  2.3 Dimensional inspection .......................................................... 17
      2.3.1 Dimensional inspection planning ................................... 18
      2.3.2 Registration of measured parts ....................................... 19
      2.3.3 Comparison of dimensions ............................................. 20
      2.3.4 Uncertainty in dimensions ............................................. 20
      2.3.5 Dimensional inspection discussion .................................. 20
  2.4 Automated inspection ............................................................ 21
      2.4.1 Brief introduction to automated inspection ....................... 21
      2.4.2 Relevant advances in automated inspection ...................... 24
      2.4.3 Deformable models and automated inspection .................. 24
      2.4.4 Discussion of automated inspection ................................ 25
  2.5 Manufacturing control — Process tuning ................................. 25
      2.5.1 In-process control ......................................................... 26
CONTENTS

2.5.2 Process tuning ................................................................. 26
2.5.3 Approaches to process tuning ........................................... 28
2.5.4 Discussion of process tuning ............................................. 30

2.6 Feedback models ............................................................... 30
2.6.1 Shape error feedback ..................................................... 30
2.6.2 Inverse mode feedback .................................................. 32

2.7 Summary ................................................................. 36

3 Methods .............................................................................. 37
3.1 Introduction ................................................................. 37
3.2 Shape description methods ............................................... 38
  3.2.1 Deformable models in computer vision ......................... 38
  3.2.2 Point distribution model (PDM) .................................. 41
3.3 Pattern recognition methods ............................................ 45
  3.3.1 Statistical classifiers .................................................. 46
  3.3.2 Artificial neural networks (ANNs) ............................... 49
  3.3.3 Error estimation of classifiers .................................... 54
3.4 Data fitting techniques ...................................................... 55
  3.4.1 Linear regression ..................................................... 55
  3.4.2 Non-linear data fitting .............................................. 57
3.5 Experimental design ........................................................ 57
  3.5.1 Factorial experiments .............................................. 58
3.6 Summary ................................................................. 59

4 The Shape Manufacturing Feedback Model ......................... 61
4.1 Introduction ................................................................. 61
4.2 Shape manufacturing feedback model ............................. 62
4.3 Manufacturing shape variation model .............................. 64
  4.3.1 Setting up the manufacturing shape variation model ....... 65
  4.3.2 Implementation of the manufacturing shape variation model . 66
4.4 Shape error function ....................................................... 67
  4.4.1 Advantages of the shape error function ....................... 67
  4.4.2 Devising the shape error function .............................. 68
  4.4.3 Shape error equations ............................................. 71
4.5 Inverse shape variation–process parameter model ............. 72
  4.5.1 Advantages of the inverse model ............................... 72
  4.5.2 Manufacturing inverse model – theory ......................... 73
  4.5.3 Learning the relationship between shape variation and process parameters ........................................... 75
  4.5.4 Implementation of the inverse model ......................... 77
### CONTENTS

4.6 Summary .............................................. 78

5 Forging processes ............................... 81
5.1 Introduction ....................................... 81
5.2 Description of the forging process ........... 82
5.3 Billet alignment and point placement ........ 85
5.4 Shape variation of forging ................. 88
   5.4.1 Major modes of shape variation .......... 88
   5.4.2 Response surfaces of $b$ vector components ... 94
   5.4.3 Varying the number of boundary points ... 96
5.5 Set-up of the forging feedback models ...... 99
   5.5.1 Shape error function ...................... 99
   5.5.2 Discrete shape/parameter inverse modelling ... 99
   5.5.3 Continuous shape/parameter inverse modelling ... 103
5.6 Results from the forging shape error function ... 103
5.7 Results of the discrete shape/parameters inverse model ... 108
5.8 Results of the continuous shape/parameter inverse model ... 110
   5.8.1 Fitting models to the forging data ........ 110
   5.8.2 Prediction models of the forging data ...... 113
5.9 Summary .......................................... 114

6 Experimental sheet metal forming ............. 117
6.1 Introduction ........................................ 117
6.2 Springback ........................................ 120
   6.2.1 Springback and channel forming .......... 124
6.3 "U" bend channel forming process .......... 126
   6.3.1 Channel forming press set-up ............ 126
   6.3.2 Experimental design ....................... 128
   6.3.3 Noise in channel forming ............... 129
6.4 Channel alignment and point placement ...... 129
   6.4.1 Boundary point distribution ............... 129
   6.4.2 Alignment of the channels ............... 133
6.5 Setting up the feedback models ............. 136
   6.5.1 Shape error function ...................... 136
   6.5.2 Springback shape error equations ........ 137
   6.5.3 Set-up of the inverse models ............. 138
6.6 Results .......................................... 138
   6.6.1 Shape variation modes ..................... 140
   6.6.2 Response surfaces of the variational modes .... 140
   6.6.3 Shape error surfaces ....................... 143
CONTENTS

6.6.4 Springback error surfaces .................................................. 148
6.6.5 Evaluation of process noise .................................................. 151
6.6.6 Results from the inverse shape variation–process parameter model 160

6.7 Discussion ................................................................................ 161
6.7.1 Shape error function .............................................................. 161
6.7.2 Springback error function ...................................................... 162
6.7.3 Evaluation of process noise .................................................... 163
6.7.4 Inverse shape variation–process parameter model ................. 164

6.8 Summary .................................................................................. 165

7 Simulated sheet metal forming ....................................................... 167

7.1 Introduction ................................................................................ 167
7.2 Simulation of springback ............................................................ 168
7.2.1 Finite element methods .......................................................... 169
7.2.2 Material models ................................................................. 171
7.3 Set-up of channel forming simulation .......................................... 173
7.3.1 Simulation set-up ................................................................. 173
7.3.2 Design of experiment ........................................................... 175
7.4 Channel alignment and point placement ...................................... 175
7.5 Setting up the simulation shape error model .............................. 176
7.6 Comparison of simulation and experimental shape variation .......... 177
7.6.1 Comparison of the variational modes ....................................... 177
7.6.2 Comparing only the simulated channels ................................. 179
7.7 Shape error of the simulated channels ....................................... 180
7.7.1 Simulation variation modes based on experimental data .......... 181
7.7.2 Shape error between the experimental and simulated channels . 183
7.8 Summary .................................................................................. 190

8 Conclusions ................................................................................... 193
8.1 Introduction ................................................................................ 193
8.2 Major contributions and results ................................................. 193
8.2.1 General conclusions ............................................................. 194
8.2.2 Forging, results and contributions ......................................... 196
8.2.3 Sheet metal forming, results and contributions ....................... 196
8.3 Future work ............................................................................... 197
8.3.1 General improvements .......................................................... 197
8.3.2 Specific improvements .......................................................... 198
8.3.3 Shape manufacturing control systems .................................... 199

References .................................................................................... 201
G "U" channel forming simulation code 259
List of Figures

1 Introduction 1
   1-1 Hedengren's inspection quality cycle ........................................ 2
   1-2 Basic steps within the inspection stage .................................. 3
   1-3 Sheet metal part with buckling .............................................. 7
   1-4 Diagram of shape manufacturing feedback model .................... 11
   1-5 Set-up of the manufacturing shape variation model .................. 12
   1-6 Set-up of the inverse shape/parameter model ......................... 12
   1-7 Implementation of the shape manufacturing feedback model ....... 13

2 Background 15
   2-1 The dimensioning and tolerancing process ............................. 16
   2-2 Process tuning control schematic ......................................... 27
   2-3 Manabe's Artificial Intelligence based process control system .... 29

3 Methods 37
   3-1 Distributing points on a resistor .......................................... 42
   3-2 Basic flow chart of a statistical classifier design .................. 47
   3-3 Model of a single artificial neuron ...................................... 49
   3-4 Feedforward artificial neural network .................................. 51
   3-5 Back-propagation algorithm schematic ................................ 52
   3-6 Example of hyperbolic tangent function ................................ 53
   3-7 General model of a process ................................................. 57

4 The Shape Manufacturing Feedback Model 61
   4-1 Diagram of the basic shape manufacturing feedback model ....... 63
   4-2 Functional basis of the shape manufacturing feedback model .... 64
   4-3 Set-up of the manufacturing shape variation model ................. 65
   4-4 Block diagram that includes noise in the MSVM .................... 69
   4-5 Set-up of the inverse shape variation - process parameter model 76
LIST OF FIGURES

4-6 Separating $\vec{b}$ vector or "shape variation" space into regions .......... 77
4-7 Detailed implementation of the Shape Manufacturing Feedback Model .... 78
4-8 Simple shape manufacturing feedback system. ................................. 78

5 Forging processes 81
5-1 The simple open forging die pair. ...................................................... 83
5-2 The "M" shape impression forging die pair. .......................................... 83
5-3 Alignment of the boundary points for forged shapes. ............................ 86
5-4 The MSVM variational mode effects of the forged parts. ........................ 89
5-5 Velocity process parameter box plots for simple forging case. ............... 90
5-6 Friction process parameter box plots for simple forging case. ............... 90
5-7 Travel process parameter box plots for simple forging case. .................. 91
5-8 Velocity process parameter box plots for "M" shaped forging case. ........... 92
5-9 Friction process parameter box plots for "M" shaped forging case. .......... 92
5-10 Travel process parameter box plots for the "M" shaped forging case. ....... 93
5-11 Offset process parameter box plots for the "M" shaped forging case. ...... 93
5-12 Simple die pair shape variation modal response surfaces. ..................... 95
5-13 "M" shaped die pair shape variation modal response surfaces. ............ 95
5-14 Simple case, friction box plots showing the effects of varying the number of boundary points. ......................................................... 97
5-15 "M" shaped case, friction box plots showing the effects of varying the number of boundary points. ......................................................... 98
5-16 Representative shape error surfaces for simple and "M" shaped die pair cases. ................................................................. 105
5-17 Shape error surfaces showing problems with the simple simulation case. 106
5-18 Shape error surfaces showing problems with the "M" shaped die simulation case. ................................................................. 107

6 Experimental sheet metal forming 117
6-1 Major defects in sheet metal forming. .................................................. 119
6-2 Stress-strain curves for bending and a combination of bending and unbending. ................................................................. 121
6-3 Stress and strain distributions for a blank formed using a single radius. 123
6-4 Relationship between bending moment and tension. .............................. 124
6-5 Simple explanation for angle springback and side wall curl. ................. 125
6-6 Typical variables used to measure springback of a channel. ................. 125
6-7 The "U" channel set-up. ...................................................................... 127
6-8 The inconsistency of the flanges and corners of the channels. .............. 130
6-9 Determination of the mid-point of a "U" channel .................................. 131
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-10</td>
<td>Fixed number of points between features point placement.</td>
</tr>
<tr>
<td>6-11</td>
<td>Distribution of boundary points on the average channel.</td>
</tr>
<tr>
<td>6-12</td>
<td>Types of channel alignment errors.</td>
</tr>
<tr>
<td>6-13</td>
<td>Initial alignment of boundary points on the “U” channel.</td>
</tr>
<tr>
<td>6-14</td>
<td>Example of a “springback free” channel.</td>
</tr>
<tr>
<td>6-15</td>
<td>The B1D1T1 experimental channel results.</td>
</tr>
<tr>
<td>6-16</td>
<td>Diagram of the MSVM variational mode effects of the “U” channels.</td>
</tr>
<tr>
<td>6-17</td>
<td>Sheet metal forming surface responses of the shape variation modes.</td>
</tr>
<tr>
<td>6-18</td>
<td>Springback error surfaces.</td>
</tr>
<tr>
<td>6-19</td>
<td>Springback error surfaces for the main set of sheet metal forming data.</td>
</tr>
<tr>
<td>6-20</td>
<td>Comparison of B1D1T3 and B1D3T3 channel sets.</td>
</tr>
<tr>
<td>6-21</td>
<td>Comparison of the B3D2T2 and B3D3T3 channel sets.</td>
</tr>
<tr>
<td>6-22</td>
<td>Comparison of the B1D1T1 and B2D2T2 channel sets.</td>
</tr>
<tr>
<td>6-23</td>
<td>Comparison of the B3D2T2 and B3D3T3 channel sets.</td>
</tr>
<tr>
<td>6-24</td>
<td>Comparison of B1D1T3 and B1D3T3 channel sets.</td>
</tr>
<tr>
<td>6-25</td>
<td>Comparison of B2D2T2 and B2D2T3 channel sets.</td>
</tr>
<tr>
<td>6-26</td>
<td>Comparison of B3D3T1 and B3D3T2 channel sets.</td>
</tr>
<tr>
<td>6-27</td>
<td>Comparison of B1D1T1 and B2D2T2 channel sets.</td>
</tr>
<tr>
<td>6-28</td>
<td>Comparison of B2D2T2 and B2D2T3 channel sets.</td>
</tr>
<tr>
<td>6-29</td>
<td>Comparison of B3D3T1 and B3D3T2 channel sets.</td>
</tr>
<tr>
<td>6-30</td>
<td>Comparison of B3D3T1 and B3D3T2 channel sets.</td>
</tr>
<tr>
<td>6-31</td>
<td>Comparison of B3D3T1 and B3D3T2 channel sets.</td>
</tr>
<tr>
<td>7-1</td>
<td>Typical stress-strain curve.</td>
</tr>
<tr>
<td>7-2</td>
<td>Several approximations of the stress-strain curve.</td>
</tr>
<tr>
<td>7-3</td>
<td>The stress-strain curve for the G3N steel material.</td>
</tr>
<tr>
<td>7-4</td>
<td>The punch velocity curve for the ABAQUS/Explicit simulation.</td>
</tr>
<tr>
<td>7-5</td>
<td>MSVM variational mode effects of the simulated “U” channels.</td>
</tr>
<tr>
<td>7-6</td>
<td>Comparison of simulated channels when varying blank holder force.</td>
</tr>
<tr>
<td>7-7</td>
<td>Simulated sheet metal forming surface responses of the shape variation modes.</td>
</tr>
<tr>
<td>7-8</td>
<td>Simulation shape error surfaces for the simulated set of channels.</td>
</tr>
<tr>
<td>7-9</td>
<td>Comparison of B1D1T3 and B1D3T3 simulated channels.</td>
</tr>
<tr>
<td>7-10</td>
<td>Comparison of B2D2T2 and B2D2T3 simulated channels.</td>
</tr>
<tr>
<td>7-11</td>
<td>Comparison of B3D3T1 and B3D3T2 simulated channels.</td>
</tr>
<tr>
<td>7-12</td>
<td>Comparison of B3D3T1 and B3D3T2 simulated channels.</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

8 Conclusions 193
  8-1 Three-stage shape manufacturing control system. ............................................. 199

A Analysis of linear models for forging data 217

B Forging simulation code 223

C Sheet metal forming — experimental procedure 227
  C-1 Die set with punch shoe, die shoe and guide pillars. . . . . . . . . . . . . . . . . . . . . . . . . . . 227
  C-2 The “U” channel forming test. ................................................................. 228
  C-3 Diagram of blank holder force sensor set-up. ........................................... 230
  C-4 The four radii of a die block. ................................................................. 230
  C-5 Diagram of the radii on the punch. ......................................................... 231
  C-6 Diagram showing the tool gap between the punch and the die. ............ 232
  C-7 Lubricating the blank. ................................................................. 233
  C-8 The configuration to tighten the blank holder. ..................................... 234
  C-9 The travel measurement set-up. ............................................................. 235
  C-10 Releasing the blank holder. ................................................................. 236
  C-11 Changing the die radii and the tool gap. ............................................ 238

D Sheet metal forming - experimental design 239
  D-1 Measurement of the corner angles for the pilot study. ........................ 243

E Orthogonal regression 251

F Chain coding 255
  F-1 Chain code directions. ................................................................. 256
  F-2 Chain code search directions. ............................................................. 257

G “U” channel forming simulation code 259
# List of Tables

1 Introduction 1

2 Background 15
   2.1 Types of control strategies associated with the relative time constants. 26
   2.2 Process tuning objectives. 27

3 Methods 37

4 The Shape Manufacturing Feedback Model 61

5 Forging processes 81
   5.1 Simple die pair process parameter levels. 84
   5.2 "M" shaped die pair process parameter levels. 84
   5.3 Simple die pair case, numbers of modes for the $P$ matrix. 91
   5.4 "M" shape die pair case, numbers of modes for the $P$ matrix. 94
   5.5 Desired process parameter levels for forging. 99
   5.6 Boundaries for the regions of simple die pair case. 100
   5.7 Boundaries for the regions of "M" shaped die pair. 100
   5.8 Linear classifier results when applied to forging data. 101
   5.9 Gaussian quadratic classifier results when applied to forging data. 101
   5.10 ANN structural details for the simple die pair data. 102
   5.11 ANN structural details for the "M" shaped die pair data. 102
   5.12 Structure of the simple die pair ANN non-linear continuous inverse models. 104
   5.13 Structure of the "M" shaped die pair ANN non-linear continuous inverse models. 104
   5.14 Calculated $\bar{b}$ vectors for the simple and "M" shaped die pair shapes. 105
   5.15 Simple die pair confusion matrices. 109
   5.16 "M" shape die pair confusion matrices. 109
   5.17 RSS for the linear parameter models fitting the forging data. 111
5.18 $R^2$ values for the linear parameter models fitting the forging data. ... 111
5.19 RSS from ANN non-linear model, simple die pair case. ............... 112
5.20 RSS from ANN non-linear model, "M" shaped die pair case. ........... 112
5.21 RSS for the linear prediction parameter models fitting the forging data. 113
5.22 RSS from ANN non-linear prediction models, simple die pair case. ... 114
5.23 RSS from ANN non-linear prediction models, "M" shape die pair case. 115

6 Experimental sheet metal forming 117
   6.1 Channel forming set-up process parameters levels. .................. 128
   6.2 The best ANN classifiers' structural details. ......................... 139
   6.3 Measured flange and floor angles and flange lengths. .............. 145
   6.4 Effects of the process parameters on manually measured springback. 150
   6.5 Typical classifier results for each of the process parameters ....... 161

7 Simulated sheet metal forming 167
   7.1 Measured flange and floor angles and flange lengths, simulated channels. 185

8 Conclusions 193

A Analysis of linear models for forging data 217
   A.1 Velocity linear regression model, simple die pair .................. 218
   A.2 Friction linear regression model, simple die pair .................. 218
   A.3 Travel linear regression model, simple die pair .................... 219
   A.4 Velocity linear regression model, "M" shape die pair ............... 219
   A.5 Friction linear regression model, "M" shape die pair ............... 220
   A.6 Travel linear regression model, "M" shape die pair ................. 220
   A.7 Offset linear regression model, "M" shape die pair ................. 221

B Forging simulation code 223

C Sheet metal forming — experimental procedure 227

D Sheet metal forming - experimental design 239
   D.1 Controllable and uncontrollable variables in "U" channel forming. ... 240
   D.2 "U" channel forming process parameter levels. ....................... 240
   D.3 Pilot study list of experiment order. ............................... 241
   D.4 Pilot study results for the lower corner of the channel. ............ 244
   D.5 Pilot study results for the upper corner of the channel. ............ 245
LIST OF TABLES

D.6 Unrandomised block of tests. .............................................. 247
D.7 Randomised die radii and tool gap combinations. .............. 248

E Orthogonal regression .......................................................... 251

F Chain coding ..................................................................... 255

G "U" channel forming simulation code .................................. 259
Notation

Abbreviations

1-D: One dimensional
2-D: Two dimensional
3-D: Three dimensional
ANN: Artificial Neural Network
ASME: American Society of Mechanical Engineers
B1: Low blank holder force, 3KN
B2: Medium blank holder force, 7.5KN
B3: High blank holder force, 12KN
BHF: Blank Holder Force
CAD: Computer Aided Design
CMM: Coordinate Measuring Machine
CSG: Constructive Solid Geometry
D1: Small die radii, 3.175mm
D2: Medium die radii, 4.7625mm
D3: Large die radii, 6.35mm
DFT: Discrete Fourier Transform
DoE: Design of Experiments
DoF: Degrees of Freedom
DR: Die Radius (Radii)


Notation

ESS: Explained Sum of Squares
FEA: Finite Element Analysis
FEM: Finite Element Model, computer vision deformable model
ICP: Iterative Closest Point, registration method
LB: Lubrication
PDM: Point Distribution Model
MSVM: Manufacturing Shape Variation Model
NURBS: Non-Uniform Rational B-splines
PR: Punch Radii
$R^2$: Square of the multiple correlation coefficient
RSS: Residual Sum of Squares
SMFM: Shape Manufacturing Feedback Model
T1: Small tool gap, 1.05mm
T2: Medium tool gap, 1.30mm
T3: Large tool gap, 1.62mm
TG: Tool Gap
TSS: Total Sum of Squares

Mathematical notation

$\hat{X}$ ⇒ the hat above a variable indicates that the variable is vector of boundary nodes.
$\vec{b}$ ⇒ the “$b$” vector is the vector that contains the weightings of shape variation of each shape.
$\vec{b}(i)$ ⇒ $i^{th}$ component of $\vec{b}$.
$\vec{U}$ ⇒ the arrow symbol above the variable indicates the variable is a vector but not a vector of boundary nodes.
$\mathbf{P}$ ⇒ a bold variable indicates a matrix.
$\mathcal{F}[]$ ⇒ the calligraphic letters, such as $\mathcal{F}$, define a function, and the square brackets always contain the parameters of the function. The square brackets will only be used for this purpose.
$\dot{M}$ ⇒ derivative of $M$ with respect to time.
$\ddot{M}$ ⇒ second derivative of $M$ with respect to time.
Chapter 1

Introduction

Shape Odyssey 2001 begins

1.1 Inspection and control of manufacturing processes

Manufacturing can be regarded as the transformation of material from one state to another, more useful state, by the application of work or energy. Manufacturing has been conducted for many centuries, particularly since the rise of the industrial revolution in the 18th century and the subsequent industrial age. The major quest within manufacturing during this time has been to make the process more efficient in terms of cost, quality and time. Henry Ford revolutionised manufacturing early in the 20th century by his exploitation of part standardisation and the production line (Pursell, 1994). Standardisation increased the speed of production by separating the manufacturing process into distinct small standardised components which were easily replaceable. Production lines were used to streamline the manufacturing process by each worker or work station only fulfilling one task on the product on its journey through the factory. Fordism brought about massive specialisation to increase speed while workers were de-skilled in all areas except their specialty. Work was often boring, tiring and dangerous.
Chapter 1. Introduction

The modern era is now a post-industrial and post-Fordist society where the computer intrudes into all areas of manufacturing. In this new era two major issues face manufacturers. First, as always, the perennial issue of a more efficient process to produce better quality products. The second issue facing manufacturers is that a decreasing number of people are entering the manufacturing sector (particularly heavy manufacturing) and much of the current process knowledge from the existing work force will be lost unless measures are taken to retain this information (Pattinson & Shirvani, 1999). Both these issues can be addressed by increasing automation within manufacturing. Automation will enhance the efficiency of production by removing tedious and sometimes dangerous tasks from the human work force, enabling long term production without fatigue problems. Workers will increase their skill base and knowledge of manufacturing processes to drive and control the automated systems. Furthermore, it is a necessary step to extract information from the manufacturing process in order to put in place an automated system. Thus, automation can be one way in which vital manufacturing knowledge is retained.

According to Hedengren (1989) the manufacturing process consists of four major areas: design, manufacture, control and inspection (see Figure 1-1).

![Diagram showing Hedengren's separate stages within the manufacturing quality cycle.](image)

**Figure 1-1:** Diagram showing Hedengren's separate stages within the manufacturing quality cycle.

The design stage creates the specifications and tolerances of the part \(^1\) to be produced to meet a functional requirement. The manufacture stage produces the part according to the specifications and within the tolerances.

The control stage has two functions. First, there is the in-process control of the manufacture stage where the manufacturing process is controlled by internal variables related to the manufacturing process, not by the accuracy of the end product. In-

\(^1\)Note that part and product are defined as the output of the manufacture stage, and they are used interchangeably throughout this thesis.
process control leads to a more precise, though not necessarily a more accurate, end product. In this case accuracy is defined as the average manufactured part being as close as possible to the overall design. Precision is the ability to manufacture the same shape without much variation, although not necessarily the shape that is desired. The second function of the control stage is to modify the manufacturing stage and the design stage, depending on the accuracy of the end product calculated in the inspection stage. This control function interprets information (both qualitative and quantitative) from the inspection stage and provides correction information to the manufacture stage and the design stage. This function has traditionally been done manually by the plant operators who have built up control knowledge of the manufacture stage through experience. The control stage is therefore highly dependent on the inspection stage which must filter the complex geometric data of the product to provide the relevant information to optimise the manufacture stage. The control stage also must update the design stage if it is impossible for the part to be manufactured under the current conditions.

The inspection stage analyses the part to determine whether it meets the specifications and tolerances; as such it is a vital stage that produces quality information to the control stage. The first problem is hence to define how the inspected product may be in error. That is, the inspection of the manufactured part must determine whether the part is acceptable. Secondly, if one can measure the error then the next problem is what the error can suggest about improving the manufacturing stage. If the manufacturing stage is unable to produce the desired design, then the inspection error can be fed back into the design stage so the design can be modified.

The inspection stage can be separated into three basic steps, see Figure 1-2: gathering the dimensional data from the manufactured part; filtering the data to obtain
only the relevant information; and finally, using tolerance information to classify the part's errors, as well as to determine some reasons why the part may have some error. Traditionally the data gathering stage has been performed manually, however, with the advent of better geometry/boundary surface measuring devices, such as 3-D point laser sensors, there is much more automatically measured data available about the end product. That is, the gathering of the dimensional data from the part can provide a large amount of accurate data. The question remains how can this geometric data be converted into relevant information for the control stage in an automated way?

This thesis will examine one way of converting geometric data into relevant feedback information for the control stage. To be precise:

This thesis investigates whether it is possible to characterise the geometric variation from a manufactured product to develop relevant feedback information for the control stage.

The above statement involves many areas of active research, namely tolerance analysis, industrial inspection, and automated control of manufacturing processes. These areas will be reviewed in Chapter 2.

The remainder of this chapter is organised as follows. Section 1.2 reviews the thesis question in more detail. An overview of the general model developed in this thesis is described in Section 1.3. The contributions that this thesis makes is outlined in Section 1.4. Finally, the last section describes the structure of the remaining thesis chapters.

1.2 Characterising geometric shape variation with a view to developing a shape-based process parameter feedback model

Characterising geometric variation of a manufactured product to develop relevant feedback information involves several terms and concepts that need to be more fully explained. This section will define these terms and concepts as well as develop the scope of the thesis question. There is also a short discussion on the difficulties of developing feedback models from geometric variation and lastly, the reader is introduced to previous approaches that have been described within the literature.

1.2.1 Definitions

Geometric variation is defined as changes in the geometric dimensions of the product away from the desired shape of the product. The dimensions of a product refer to the boundary or outside of the product. The internal properties of a product cannot be determined by visual data gathering sensors.
Deformable models are models of objects that are allowed to vary in shape to model the shape variation of the object or class of objects.

Manufacturing processes have many parameters that affect the end product. Some of these parameters are controllable by the operator. Controllable parameters that can be tuned at the beginning of each production run are hence defined as process parameters.

Feedback is the supply of relevant information about the output to assist the improvement of the manufacturing process.

Shape error is defined as the filtered differences between the output of a manufacturing process and the desired output.

Inverse models are defined as models that relate the end product of a manufacturing process to its process (controllable) parameters.

1.2.2 Scope

Characterising geometric variation of a manufactured product to develop relevant feedback information can be broken into two complementary components. The first component is an investigation of geometric shape variation in the final manufactured product. The characterisation of geometric shape variation examines only the surface boundaries of the manufactured part and does not consider surface texture or material properties of the part. Also, it is assumed that the boundary data will be available in range data point-based format from the inspected part. Furthermore, to limit the scope to a manageable level, this thesis will only investigate two dimensional curves or cross sections of the manufactured part. This allows the modeling of objects that have a constant third dimension, for example, a sheet metal formed channel. The characterisation of the geometric shape variation is based upon already existing shape recognition techniques, which will be explained later in Chapter 3. These techniques were customised to apply to the chosen manufacturing systems studied in this thesis.

The second component examines the feedback that can be provided from analysing the characterised shape variation. The scope of feedback to the manufacturing process will be limited to two forms. The first form of feedback is the error in the geometric shape between the desired shape and the manufactured shape (shape error). The second form of feedback identifies the parameters that affect shape variation in the manufacturing process. Inverse models are then developed that relate a finished product shape to the process parameter levels used in its manufacture. The extraction of information obtained from the description of the geometric shape variation is achieved by standard statistical and non-linear methods.

Finally, there is a scope issue involved with the validation of any approaches that are explored in this thesis. There are a plethora of manufacturing processes but this thesis will focus on shape manufacturing processes. Shape manufacturing is defined as
a process which deforms a billet or blank into a new shape, such as rolling, extruding, forging and sheet metal forming. Shape manufacturing processes were chosen for two reasons: first, shape manufacturing processes are widely used in industry; and second, the set-up of shape manufacturing experiments is not intractable. Sheet metal forming produces the largest number of parts among all the shape manufacturing processes, particularly for the automotive industry. Forging is another important manufacturing technique (Schey, 1987) and therefore, the validation of approaches examined in this thesis will be tested on these two types of shape manufacturing processes, forging and sheet metal forming. The validation is also conducted on both actual and simulated sheet metal forming data with the aim to provide feedback on sheet metal forming simulations using actual data.

1.2.3 Difficulties

The difficulties faced when characterising geometric variation in shape manufacturing parts to develop process parameter feedback models fall into two categories: first, characterising shape changes of the manufactured product; and second, relating these changes back to the manufacturing process.

1.2.3.1 Characterising shape changes

There are four basic questions that need to be addressed when characterising the shape changes of a manufactured object.

1. What is the best method for describing shape variation?

2. Can the describing method be automated?

3. How are the desired and the manufactured shapes compared?

4. Can further information be extracted from the shape comparison between the desired and the manufactured shape?

The first difficulty faced is how to describe the shape of an object. For example, how does one describe the shape of the buckled sheet metal part shown in Figure 1-3? The difficulty in representing shapes in a structured way is that it is hard to find a concise description because of the large dimensionality of the problem. Generally high order mathematical representations of surfaces and/or boundary curves have to be created to model at least complex two dimensional objects. Once these descriptions are available, are they flexible enough to also describe variations in the object's shape? This is where deformable models can be of use. There is a number of deformable model types that can be used to describe a class of shapes, and the question of which is the best model to be applied to characterising shape variation in shape manufactured parts is discussed in more detail in Chapter 3.
Once an appropriate deformable model has been chosen, the second problem that arises is how the deformable model can be applied to the object in an automated way. Most deformable models use some form of control points or distinctive features related to the modelled object to control the shape of the deformable model. It is not a trivial task to develop a method to adequately describe the boundary using an automated method (Cootes et al., 1995). This thesis will endeavour to use only automatic computer based boundary descriptions.

The next question that needs to be addressed is how the manufactured shapes will be aligned against the desired shape? The manufactured shape has some randomness in its boundary due to noise in either the manufacturing process or the measurement sensors. The noise in the manufactured part means that it may be difficult to find appropriate points of reference because the exact geometry of the measured part is not known.

Once the shape of an object and its variations can be described by a deformable model and it is aligned to the desired shape, then the next task is to compare the desired shape to that of the manufactured shape. This task is made slightly easier by the fact that the desired shape should already be described by a CAD model in the design stage.

Finally, the shape variation description method must also be able to provide salient information such that an inverse manufacturing model can be created. This means that the information being provided from the shape description model must be as minimal as possible and also give as much distinction between differing manufactured parts as possible.
1.2.3.2 Feedback to the manufacturing process

Once the challenges of describing the shape changes have been overcome there is the problem of developing appropriate feedback for the manufacturing process. This thesis will examine two types of feedback, shape error and inverse modelling.

There are two main questions arising from shape error feedback.

1. How valid is the shape error function?

2. Can this error be interpreted in a way that improves the manufacturing process?

The shape error that is calculated for a manufactured part must have a direct relationship with the true errors between the manufactured part and the desired part. This can only be validated by comparing the results from the shape error function against known error values for varying conditions of the manufacturing process.

Secondly, if the error function is valid then the error must also provide valuable feedback to the manufacturing process. The feedback to the manufacturing process must be concise and easy to relate to the controllable aspects of the manufacturing process.

There are three basic questions to be addressed when investigating inverse manufacturing models.

1. What is the stochastic nature of the process?

2. Is the process a one-to-one function or a relation (many-to-one)?

3. Is the process non-linear?

The first difficulty faced when determining an inverse model of a manufacturing process is the inherent variability associated with any shape manufacturing system. The stochastic nature of the manufacturing process will determine the preciseness of any inverse model.

The type of relation or function that best models the manufacturing system will affect the ability to determine an inverse model. For example, if many process parameter combinations can manufacture the same shaped object then it would be impossible to find an inverse model because this single shape has multiple correct solutions.

Finally, manufacturing processes are often complex operations that are non-linear in nature and are hard to model forwards let alone as an inverse. For example, the phenomenon of springback within sheet metal forming is a difficult process to model, see Chapter 6.
1.2.4 Approaches

Many approaches to inspection of manufactured parts, primarily those involved in forging and sheet metal forming, traditionally perform dimensional inspection. Dimensional inspection compares the dimensions of the manufactured part to its desired dimensions. Often these results are then compared to the tolerances assigned to the part in the design stage. These results from the inspection process are not usually connected directly to tuning the manufacturing process. We aim to extend inspection further to provide feedback to the manufacturing process. This feedback takes the form of shape error and inverse modelling. Approaches to these two forms of feedback will be briefly described.

1.2.4.1 Shape error

Inspection in shape manufacturing processes has traditionally concentrated on determining the dimensional error of a part which is then compared against the tolerances of the part. Tolerance information is good for determining whether a part meets the designed criteria (should the part be accepted or rejected) but it is perhaps not the best information to feedback to the manufacturing process in a control sense. This is because many of the tolerance criteria are ambiguous or binary in nature. Binary in this case means that, for a given criteria, the part can only have two values, acceptable or unacceptable, and this gives no indication of how close the part is to the criteria nor what should be tuned in the manufacturing process. For example, Modayur et al. (1992) examined manufactured parts to determine if they fitted within a tolerance zone around the desired part. If the part fitted within the zone, then it was acceptable. Otherwise, the part was unacceptable.

Research into die design also uses shape error. Die designers want to know what are the differences between their desired end shape and the end product caused by the designed die. Often the shape error is a simple function that calculates a weighted sum of the projected distances between the desired shape and the manufactured shape (Fourment & Chenot, 1996; Karafillis & Boyce, 1996). However, in this case it is important that the shape error can be used to optimise the design process. That is, the shape error gives an indication of how to improve the manufacturing process.

There is a major improvement that can be made to the shape error functions described in the literature. Consider the shape error of a part, where not all the localised shape error on the part is significant. An improvement could be made to weight significant errors over insignificant errors. A corollary of this improvement is that the simple shape error models may not give the best feedback about how to improve the manufacturing process. A related improvement to shape error functions could be made to highlight the areas of the manufacturing process that need tuning.
1.2.4.2 Shape variation–process parameter inverse models

Most of the inverse model literature within shape manufacturing is related to die design. In the case of die design the prime engineering problem being solved is the question of how to manufacture a part when its desired shape has already been designed. In other words, what is the appropriate profile of the die to correctly manufacture the desired part. Inverse modelling uses shape optimisation to modify the shape of die to obtain the appropriate output shape. For example, Fourment (1996) and Zhao (1997) developed two examples of inverse models in forging to find the appropriate preform profile to assist creating the final shape. Webb and Hardt (1991), and Karafillis and Boyce (1992; 1996) developed die designs in an attempt to predict the effects of springback to create the appropriate final shape. Gelin and Ghouati (1995; 1996) used inverse modelling to determine material parameter levels in large deformation manufacturing process.

This thesis will take a new approach to inverse modelling and use the output shape variation from the manufacturing process to predict what are the process parameters’ levels that created the manufactured part. This is quite a different perspective on the previous approaches which use the material models of the part to provide the variation in the shape and drive the inverse model. Furthermore, because this author is interested in manufacturing control, this thesis is motivated towards investigating what might have caused errors in the manufactured part and whether the process parameters can be tuned to solve this error.

Thus, the essential difference is that this work is not optimising the parameters to find what is the optimal set-up to give the desired outcome but rather, this thesis is developing a model that returns the most likely set-up that created the manufactured output.

1.3 Basic model description

This thesis characterises shape variation in shape manufacturing to develop a novel feedback model, hence this model is defined as the shape manufacturing feedback model. This feedback model takes an end product after the manufacturing stage and inspects the product to determine its shape variation, and hence interprets this error to provide feedback to the control stage. The shape manufacturing feedback model is therefore comprised of two main component models, see Figure 1-4. The first component model describes the variation in shape and is based on the Point Distribution deformable Model (Cootes et al., 1995). The second component model determines what is the most likely parameter combination to cause the measured shape error.

The two component models are experiential or inductively based. The approach is similar to a human's approach of learning through experimentation and empirical behaviour. The two component models are set-up by varying the parameters of the
manufacturing process in a structured way to create a set of manufactured parts. The first model calculates the major modes of shape variation of the process from the set of manufactured parts to use as its basis, see Figure 1-5. The two component models are, therefore, initialised by creating a set of manufactured parts that indicate the manufacturing process' variation. This set of parts is used to calculate the major modes of shape variation that the manufacturing stage causes. These variations are then instantiated into a deformable model, the manufacturing variation model. This model can then describe manufactured parts in terms of the amounts of variation away from the average manufactured shape.

The second model is initialised by taking the resulting variations from the manufacturing shape variation model and combining these with the original parameter levels used to cause the variations. The relationship between the variations and the parameters are then learned using identification and pattern recognition theory. The resulting relationship is the inverse shape/parameter model, see Figure 1-6.

These two initialised models are combined to create the shape manufacturing feedback model, see Figure 1-7. A manufactured part can then be given to the manufacturing shape variation model which will describe the parts' major variations with respect to the average manufactured shape. These variations can then be compared against the variations of the desired shape. This gives the shape error of the manufactured part which is the first feedback to the control stage. The variations are then given to the inverse shape/parameter model which then estimate the parameters used by the process to create the manufactured part. The estimated parameters can then be used in conjunction with the shape error as a part of a control strategy in the control stage to improve the output of the manufacturing stage.
Chapter 1. Introduction

Manufacturing parameters

Vary Parameters

Manufacture Parts

Inspect Parts

Measured set of manufactured parts

Calculate the Shape Variations of the Manufacturing Process

Manufacturing Shape Variation Model

Design Part

Desired Shape

Figure 1-5: Diagram showing the set-up of the manufacturing shape variation model which is a component of the shape manufacturing feedback model.

Manufacturing parameters

Vary Parameters

Manufacturing Shape Variation Model

Set of parameter levels

Determine Shape/Parameter Inverse Relationship

Inverse Shape/Parameter Model

Figure 1-6: Diagram showing the set-up of the inverse shape/parameter model which is a component of the shape manufacturing feedback model.
Chapter 1. Introduction

1.4 Contributions

"Characterising geometric variation in shape manufacturing with a view to process parameter feedback"

This thesis makes a significant research contribution by introducing a method of creating a novel shape error function for shape manufacturing processes, namely forging and sheet metal forming. Furthermore, the shape error measure can be used to gain information on how to improve the shape manufacturing process. This is also applicable to improving simulations of shape manufacturing processes. This thesis provides three general significant contributions including:

1. shape variation based feedback models for simulated forged parts;
2. shape variation based feedback models for experimental sheet metal formed parts;
3. shape variation based feedback functions to determine the accuracy of finite element sheet metal simulations.

1.5 Outline of thesis

Chapter 2. This chapter outlines the background literature that has motivated this thesis.

Chapter 3. This chapter reviews the methods of both describing the geometric shape of objects and pattern recognition. Deformable object models, which are descriptions of objects that are allowed to vary their shape, are discussed. The weaknesses and the strengths of each method are examined. The Point Distribution...
Model (Cootes et al., 1995) is outlined in detail as this is the shape model used throughout this thesis. This chapter also reviews methods of classification and identification as well as some function fitting methods that are used throughout the thesis.

**Chapter 4.** This chapter discusses the model investigated by this thesis in detail. The model combines shape description and identification methods to create the shape manufacturing feedback model capable of determining the major modes of variation and hence calculating what process parameter levels were used to create the part.

**Chapter 5.** This chapter investigates the application of the shape manufacturing feedback model to hot metal forged parts. The metric and inverse models are developed using two sets of simulated hot forged parts created using two different die pairs (simple and “M” shaped die pairs). A neural network is used to classify the shape data into three arbitrarily chosen levels for each parameter.

**Chapter 6.** This chapter implements the shape manufacturing feedback model on actual sheet metal formed channels, and also indicates the success to which the model can be applied to actual shape manufacturing processes. The main contribution of this chapter is to develop a novel shape error metric that identifies geometric shape differences from a nominal sheet metal part. The shape error metric is then extended to develop a simple empirical springback measure. Furthermore, inverse models are created using classifiers that relate the shape variation to the process parameter levels for the parameters blank holder force (BHF), die radii (DR) and tool gap (TG). Several classifiers (linear, quadratic classifiers and artificial neural networks) are used to create models.

**Chapter 7.** This chapter applies the shape manufacturing feedback model on simulated sheet metal formed channels. The sheet metal forming simulation was created to model the channel forming process of Chapter 6. Comparisons between the actual and the simulated channels are made using the feedback models.

**Chapter 8.** This chapter concludes by summarising the contributions this thesis has made toward research in inspection and manufacturing control. Future avenues of research are also outlined.
Chapter 2

Background

Shape Odyssey 2001 continues

2.1 Introduction

The previous chapter outlined the scope and defined the problem this thesis will investigate. This chapter describes the background research areas that have motivated this thesis, namely: tolerancing, dimensional inspection, automated inspection, and manufacturing control. These topics are addressed in that order. The shape manufacturing feedback model creates two forms of feedback model: a shape error function and a shape variation–process parameter inverse model. Finally, the related literature to these two models is also discussed.

2.2 Tolerancing and variational modeling

Tolerancing is the study of the amount of variation allowed for features and dimensions of a product to not affect the functionality or cost of the product. Daniel (1998) de-
scribes the dimensioning/tolerancing process as follows, see Figure 2-1. When a part is designed, the dimensions are created to suit the design criteria. The tolerance synthesis process then creates a set of tolerance specifications that guide the manufacturing and inspection processes. These specifications are analysed to make sure the dimensions and the tolerances are valid. This is defined as tolerance analysis. If the tolerances are invalid, they are revised and sent back for analysis. If these revisions are unable to be made due to the constraints on the tolerances and dimensions, such as a precision limit on a measurement tool, then the whole process is started again to obtain a valid design specification.

2.2.1 Tolerancing approaches

Hillyard (1978) and Requicha (1983) were the first to mathematically formalise the theory of dimensioning and tolerancing. The mathematical models for geometric tolerance representation were extended by three main approaches: the tolerance zone approach (Requicha, 1984), variational geometry approach (Lin et al., 1981; Light & Gossard, 1982; Martino & Gabriele, 1989), and other variation based models (Gupta & Turner, 1993). Mathematical tolerance specifications were standardised by ASME with the ASME Y14.5.1M–1994 standard (ASME, 1994). These standards are being continually
improved upon for more complex geometric shapes (Roy & B.Li, 1999).

### 2.2.2 Variational models

The motivation for this thesis has arisen from the idea of extending variational models for tolerancing to an automatic inspection context. Gupta and Turner (1993) define a variational model to be a computer-based model of a variational class or a collection of related instances of a part or assembly. Light and Gossard (1982) initially developed variational geometric modeling to aid in the design and dimensioning of CAD based parts. The variational models, in this case, are geometric models constrained according to the geometry of the design and they aid in the tolerance analysis of the part. Etesami (1988) established the concept of the Manufactured Part Model (MPM) based on variational models to verify tolerances, where an MPM is a parametric geometric model which describes the nominal part and its variants. Wang et al. (1998) have evolved the MPM further to develop a method to characterise variations in manufactured automotive space-frame extrusions. Essentially the variational model or the MPM is a deformable model where a parameter can be used to vary the model within a certain variational class or constraint.

### 2.2.3 Discussion of tolerancing

Tolerancing provides the limits for a valid part, and this process is becoming increasingly automated. Tolerancing also provides information to the manufacturing stage that indicates which processes have to be carried out with more precision. In other words, the tolerance specifications are used as a guide in controlling the manufacturing process. The important information in manufacturing control, however, is the error between the desired and the actual manufactured part. The variational modeling approaches to tolerancing provided the inspiration that deformable models may be an appropriate way to characterise the shape error of a part. Cardew-Hall et al. (1997) used this fact to base their shape deformation model on a deformable model to compare measured parts versus their geometric tolerances as defined by tolerance standards. This leads naturally to the field of dimensional inspection.

### 2.3 Dimensional inspection

One of the main motivating research areas for this thesis is that of dimensional inspection, also known as metrology. Dimensional inspection is a type of automated inspection where the measured dimensions of a part are compared against the desired dimensions to determine if the part is within the specified tolerances. The research literature of dimensional inspection links closely to that of tolerances primarily because as soon as new and improved tolerance models are created, they are transferred into dimensional
inspection systems. Dimensional inspection is also under-pinned by *CAD based inspection*, explained in Section 2.4.1.4. CAD models provide a mathematical representation of the part's shape and they are also capable of retaining tolerance information about the part.

Dimensional inspection can be split into four areas:

- dimensional inspection planning;
- registration of measured parts to desired parts;
- comparison of measured dimensions against tolerance values;
- uncertainty in the dimensional measurements.

Inspection planning is where the appropriate points or positions that need to be measured for analysis are determined. It also considers what the appropriate measurement path may be. Finally, it determines whether there are regions of the part that are occluded or otherwise impossible to measure. The registration of measured parts considers how to align the measured parts to the desired parts. The research area of comparing measured dimensions against tolerance values considers new methods of performing this comparison. The uncertainty in dimensional measurements explores the problem of noise within the measurement values.

### 2.3.1 Dimensional inspection planning

The challenge within inspection planning is to develop a system that can take the geometric CAD data of a designed part and produce an inspection plan. Lee *et al.* (1992) broke automatic dimensional inspection planning into four main steps:

- Step 1: determine the representation of inspection features;
- Step 2: formalise the measuring process;
- Step 3: model the inspection activities;
- Step 4: synthesise the plan.

Around these four steps they developed a state-based model for inspection planning. This model was composed of logical descriptions of: extracting inspection features from the manufacturing drawings, selecting measuring instruments, selecting the correct fixtures from which to measure, arrangement of the measurement procedures, and evaluations of the measurement data.

One of the major problems faced in inspection planning is determining whether all the dimensions are accessible by the measuring machine. Spyridi and Requicha (1993) established some techniques to determine the accessibility of measurable points on a
part, primarily by establishing the algorithm of direction cones. This aided process planning by helping to eliminate impossible measurement paths. Sobh et al. (1995) suggested a different view of process planning by using the manufacturing process' toolpaths modelled by Bézier splines to aid planning and accessibility. The toolpaths can be a good indicator of accessibility. Spitz et al. (1999) extended accessibility algorithms by proposing two methods that determine the accessibility of dimensions on polyhedral objects.

Another difficulty faced within inspection planning is determining how to inspect complex parts. Lin and Chen (1997) separated the part into the base Constructive Solid Geometry (CSG) elements of the solid model, and created plans from a bottom up approach of combining the inspection plan of each CSG element. Ainsworth et al. (2000) recently developed a planning system for the complex problem of free-form shapes where the surfaces were modelled by non-uniform rational B-splines (NURBS).

2.3.2 Registration of measured parts

The registration of the measured shapes to the desired shape is the second major problem faced in dimensional inspection. Registration is the process of aligning the features of the measured part to the features of the desired part. A particular type of registration is localisation, where a subset of the measured points is located in the desired part's coordinate system. Registration is a complex problem because the measured part is different to the desired part and, therefore, the measured part will not exactly match the desired part. Furthermore, it presents a major uncertainty to inspection planning because reference features may not necessarily be in the desired place.

The first major difficulty of registration is that the optimisation to find the minimum alignment error between the measured shape and the desired shape is a non-linear problem. There have been many solutions to this problem, many are based on the Iterative Closest Point (ICP) registration method (Besl & McKay, 1992). Menq et al. (1992) applied an optimisation technique to determine the minimum distances between the measured and desired shapes. Balasubramanian and Peihua (1995) also determine the shortest distances between the measured and desired surfaces. A neural network was used to learn the transformation matrix between the measured part's coordinate system and the desired part's coordinate system.

The second problem is determining the effect of measurement noise on the registration. Brujic and Ristic (1997) used the Monte Carlo simulation and analysis technique on their free form surfaces to test the ICP registration method (Besl & McKay, 1992). The surfaces were defined by non-uniform rational B-splines (NURBS). They showed that a larger number of measured points improved the statistical confidence of the alignment. In addition, they concluded that measuring a large number of low accuracy points can provide better results than measuring a very small number of points with
high accuracy.

2.3.3 Comparison of dimensions

The third area of research within dimensional inspection is the comparison of the measured part differences against the desired shape. Modayur et al. (1992) define dimensional inspection as more than fitting curves to the measured data and determining the difference against the desired shape. They suggest that dimensional inspection also includes comparing the shape error to the tolerances specified by the designers.

Therefore, the important issue for the comparison of dimensions is how to interpret the tolerances. Modayur et al. (1992) compared the measured shapes to a tolerance zone around the features of the desired part. Medland et al. (1996) extended the tolerance zone analysis by using spherical tolerance zones to verify dimensions, whereas the previous authors had used a square zone. The spherical type of zone gives a better radius of error than the square zone. Cardew-Hall et al. (1997), as previously stated, established a new method of comparing dimensions by developing the shape deformation model (SDM) to compare dimensional tolerances. The SDM was a deformable model whose variations were constrained in the directions specified by the tolerances. The parameter levels of the deformable model, therefore, gave an indication of the error level of each dimension relative to each of the tolerances. Choi and Kurfess (1999) developed a tolerance comparison method by extending the localisation registration technique. They suggested that a part and its tolerances create a zone for the valid part. If the measured points can be transformed to fit inside the tolerance zone then the part is valid.

2.3.4 Uncertainty in dimensions

The final area of research in dimensional inspection is that of uncertainty. Uncertainty or noise affects the measurements of the dimensions and this, in turn, affects the comparison of dimensions. Modayur et al. (1992) included noise in their theory of measurement points to determine the variance on their features. They then used the variance to calculate a statistical test of whether the part is valid. Menq et al. (1992) calculated the transformation error relative to the dimensional error of the measured points, defined as the sensitivity measure. The probability that the measured point falls within the tolerance zone can be calculated. The number of measured points needed for dimensional inspection can then be calculated using confidence intervals when some assumptions of normality are made.

2.3.5 Dimensional inspection discussion

Dimensional inspection is made up of four components: inspection planning, registration, analysis of measurements, and the uncertainty of the dimensional measurements.
This thesis assumes that the inspection planning has already been performed, and that the boundary surface data is available for the inspected part. The registration of parts, and indeed any objects, remains an open problem. Registration of the measured and desired parts is performed within this thesis, but existing methods are used for this process. It is out of the scope of this thesis to investigate new methods of object registration. The analysis of measurements determines whether a measured part fits within the tolerance zone of the desired part. This use of tolerance zones is able to determine whether a part is acceptable, but it does not provide all the information necessary to perform manufacturing control. The best representation of error, in a control sense, is to characterise the error in a way that can be interpreted by a controller. This implies that the error must be able to give an indication of the magnitude of the error both locally and globally. In addition, the reasons why this error may be occurring needs to be investigated. This thesis extends upon the study of Cardew-Hall et al. (1997) to characterise shape error using deformable models. Finally, it is important to include uncertainty in the analysis of dimensional error. This thesis will examine the uncertainty of the manufacturing process, but it is out of the scope of this thesis to conduct an in-depth analysis of uncertainty with regard to characterising shape error.

Tolerancing and dimensional inspection have now been briefly introduced, but note that dimensional inspection is the combination of tolerancing and automated inspection. Can the broader field of automated inspection offer any motivation to this thesis, and particularly with respect to characterising shape error?

2.4 Automated inspection

Automated inspection is defined as the inspection of a part or product to determine if it deviates from a given set of specifications. Automated inspection has been an active area of research for several decades. According to Chin (1988), automated visual inspection began by applying computer vision techniques to inspecting circuit boards and other electronic equipment in the early 1980s.

2.4.1 Brief introduction to automated inspection

Newman and Jain (1995) comprehensively reviewed the huge field of automated inspection and their paper is summarised in the following sections. This will give the reader a flavour for the breadth of automated inspection and some of its underlying systems.

2.4.1.1 Advantages and disadvantages of automated inspection

There are a number of motivating factors for conducting automatic inspection research:

- automated systems have the possibility of testing 100% of the produce rather than the random batch inspection carried out by human inspectors;
• some machine sensors are better than the human senses and are able to better find defects;

• reduction in labour costs;

• reduction in dangerous and tedious work.

Nevertheless, there are a number of weaknesses related to automatic inspection. First, the speed of past automated inspection systems has not been as fast as anticipated. It is believed that this will change in the future with the advent of faster technologies. There are also high capital costs associated with developing and implementing automatic inspection systems. Finally, the flexibility of human inspectors is much greater than automatic inspection systems. That is, human inspectors are faster to re-train and are better able to handle unexpected inspection errors.

2.4.1.2 Data collection systems

Within automatic inspection systems there are many ways in which the sensor data is collected. Visual inspection systems make use of binary, grey level and colour images. Other systems use range data, X-rays, ultra sound and magnetic resonance data.

Binary imaging systems

Binary imaging systems give only a coarse view of an object and they are useful for simple fast inspection analysis. Due to the simple representation of the image much of the background noise is removed from the analysis. On the other hand, some of the subtle information is lost that could aid the inspection process. Binary image inspection has had many applications including bolt production (Batchelor & Braggins, 1992), integrated circuits (IC) (Ninomiya et al., 1989) to surgical staples and ligating clips (Taylor, 1990).

Grey level imaging systems

Intensity or grey level imaging systems add more information to the inspection process by adding more texture to the images. However, this also adds more noise to the analysis. Often this is deleterious in harsh industrial environments with dirt and unfavourable lighting conditions affecting the analysis. Intensity images have had a wide number of applications including ceramic tiles (Desoli et al., 1993), printed wiring boards (Hunt, 1985) and textiles (Norton-Wayne et al., 1992).

Colour imaging systems

Colour imaging systems are important in the analysis of colour sensitive applications such as food or fashion clothing. Humans have good relative colour vision but are not good at absolute colour vision. This means that given a single example without a comparison it is difficult for a human to decide accurately the exact colour of the object.
Automated sensors are able to determine an exact colour, however, similar to the grey level image analysis, colour image analysis suffers from noise and unfavourable lighting problems. Some examples of colour inspection systems are fruit grading (Poole, 1989) and detecting dye streaks (Daley & Rao, 1988).

**Range data systems**

The use of images (binary, grey level, and colour) is best suited for 2-D applications because of the 2-D nature of images. To gain 3-D information about a part from image analysis, a series of 2-D images have to be combined to create a 3-D reconstruction of the object. The 3-D reconstruction is often difficult, particularly in terms of registering the set of 2-D images.

Within manufacturing many of the parts produced are 3-D and for this reason the use of 2-D images is not entirely useful in inspection. Furthermore, an accurate 3-D reconstruction of the part is necessary for dimensional inspection. Range data provides 3-D depth information from a scanned scene. Because range data systems can provide 3-D information about an object they are often used in manufacturing applications. Range data systems have been applied to casting (Newman, 1993), extruded aluminium profiles (Dahle *et al.*, 1990), and open-die forging inspection (Wright & Bourne, 1988).

### 2.4.1.3 Image analysis methods

After gathering the sensory data, inspection systems employ three main methods of analysis: template matching, rule-based methods or a hybrid of the two methods. The template matching scheme compares the sensory data of the part or object against a defect free model, whereas the rule-based scheme compares the sensory data against a list of design rules.

### 2.4.1.4 CAD based inspection systems

Finally, there is a form of automated visual inspection that uses CAD models as the desired shape or part. This is known as *CAD-based inspection*. The advantage of using CAD models is that the desired shape is already defined by a mathematical description. Secondly, the CAD models often retain addition information about the designed part (such as tolerances). However, the main disadvantage with using CAD models for inspection is that often the CAD models are not updated to reflect changes made throughout the manufacturing process. CAD-based inspection has been applied to printed circuit boards (Silvén *et al.*, 1989) and profiles of extruded aluminium (Dahle *et al.*, 1990).
2.4.2 Relevant advances in automated inspection

Since Newman and Jain’s (1995) review, automated inspection has remained a very diverse field and continues to be very application dependent. For example, better inspection solutions are still being found for many different applications such as textiles (Stojanovic et al., 2000), food produce (Mauris et al., 2000) and printed circuit boards (Rice III. et al., 2000; Tonapi & Srihari, 2000; Ayoub, 2001). The main advance in automated inspection, however, has been the use of non-linear techniques to find solutions to complex problems. Artificial neural networks and fuzzy logic have been used to solve an increasing amount of inspection analysis problems (Xu et al., 1999; Dae-un et al., 2000; Kukkonen et al., 2001). In addition, increased computing power and better sensor technology is enhancing the quality of the sensory data (Baba et al., 2001; Tomassini et al., 2001). Finally, the increase in the standard of computer vision methods has increased the capabilities of automated inspection (Capineri et al., 1998; Han et al., 1998; Horng-Hai & Ming-Sing, 1999).

2.4.3 Deformable models and automated inspection

Deformable models, as was stated in the previous chapter, are models of objects that are allowed to vary in shape. These models have been rarely used in relation to automated inspection, even though in certain applications they may be very useful. Etesami and Uicker (1985) first used a deformable model in automated inspection. They used Fourier descriptors to describe the measured shape and desired shape of arbitrary industrial objects. The difference between the two sets of Fourier descriptor parameters were used as a measure of the dimensional error in the shape. Noble et al. (1992) developed a deformable template technique to inspect drill holes in X-ray images. Di Mauro et al. (1996) created an automatic quality control system of industrial parts based on the Point Distribution Model (Cootes et al., 1995). They applied this system to investigating components on a printed circuit board. Daniel et al. (1997) also used the Point Distribution Model to develop an automated inspection system that inspected 3-D turbine blades. Fisker and Cartensen (1998) presented a method to automatically determine the parameters of a free-form deformable template. They applied their automatic deformable model to the inspection of textiles.

In medical image analysis, a related research field to automatic inspection, deformable models have been implemented to solve difficult imaging problems over the past decade (Duncan & Ayache, 2000). Medical researchers have used image analysis to determine abnormalities and disease in the organs of patients. The objects in medical imaging are organic with smooth, rounded, but complex shapes. Often, these shapes are marginally different between patients and some objects have a fourth dimension because they vary their shape over time (for example, the heart). Deformable models are the ideal type of model to analyse these objects. The deformable models used for
this analysis range from 3-D "snakes" (Cohen & Cohen, 1993), finite element models (Nastar & Ayache, 1996), and the Point Distribution Model (Cootes & Taylor, 1992; Hill et al., 1992; Hill et al., 1993). These models will be described in the next chapter.

Given the similarity in the general aims of the two fields it is surprising that more deformable models have not been implemented in automatic inspection.

2.4.4 Discussion of automated inspection

One long term aim of automated inspection is to provide a single standard system that will need only limited customisation when applied to different situations. That is, the objective is to have an application-independent automated inspection system. This will not be achieved in the near future and it will rely upon the computer vision research field developing standard robust feature analysis programs.

Returning back to this thesis and its relevance to automatic inspection, the application of deformable models techniques within inspection of manufactured parts has been slow and this thesis will attempt to redress this issue. The shape manufacturing feedback model will extend the previous automatic industrial inspection systems that use deformable models (Etesami & Uicker, Jr., 1985; Di Mauro et al., 1996; Daniel et al., 1997) by using deformable models to create two feedback models that quantify the error in a part. These two feedback models, in the first instance, examine forging and sheet metal forming parts. The feedback models to quantify error are particularly relevant to sheet metal forming because currently there is a lack of automatic inspection methods to classify shape error problems (Kase et al., 1999).

There is also an increasing amount of data that will be made available from manufactured parts due to better sensor systems. Combining this with the increasing quality of computer vision techniques, leaves greater scope for the automatic analysis of parts. Certainly, it is possibly that within a decade all manufactured parts may be scanned for dimensions after production. This thesis aims to provide a solution to analysing this potentially massive amount of scanned part data. This thesis also makes use of non-linear classification techniques which have increased the accuracies of some recent automated inspection systems.

This increasing quality of information about the errors of a manufactured part, however, can only be made useful if the manufacturing stage is able to be tuned.

2.5 Manufacturing control — Process tuning

As discussed previously, there are two types of manufacturing control: in-process control and process tuning, see Section 1.1.
Table 2.1: Table of the relative time constants for the process and the feedback delay and the appropriate control strategy.

<table>
<thead>
<tr>
<th>Time Constant Comparison</th>
<th>Control Strategies</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_f &lt; T_p$</td>
<td>Real-time or in-process control</td>
</tr>
<tr>
<td>$T_f \approx T_p$</td>
<td>Iterative control</td>
</tr>
<tr>
<td>$T_f \gg T_p$</td>
<td>Statistical process control</td>
</tr>
</tbody>
</table>

2.5.1 In-process control

In-process control, also known as process parameter control, is where a process parameter is controlled within the manufacturing process. Within sheet metal forming the blank holder force\(^1\) parameter has been used to form closed-looped in-process control systems to obtain better material flow (Hirose et al., 1992; Hardt & Fenn, 1995; Siegert et al., 1997; Cao et al., 2000). This has aided the creation of increasingly complex sheet metal part designs. This type of control system, however, does not address the accuracy of the final manufactured part as it concentrates on maintaining a specified process parameter on a particular path or profile. Better control of the in-process parameters does not necessarily lead to greater quality in output.

2.5.2 Process tuning

Process tuning, on the other hand, is where the process parameters of a manufacturing process are tuned after the completion of the manufacturing process to increase the accuracy of future manufactured parts. This means that, after each manufacturing run, a part or a representative of a set of parts is inspected to determine the error of the part or set of parts. This error is fed back via a controller that tunes the manufacturing process to remove the error. Process tuning is also known as process output control because the feedback is based on the output of the manufacturing process. Figure 2-2 shows the basic structure of the process tuning model. The desired geometric properties minus the results of the previous manufacture are fed into the controller. The part is manufactured according to the specifications of the controller. The geometric error can then be calculated, and the error is fed back to the controller. The relative time difference between the manufacturing the part, $T_p$, and calculating the error in the part, $T_f$, determines the type of control strategy used (Hardt, 1993). These strategies are summarised in Table 2.1.

Hardt (1993) states that process tuning and control system objectives have much correspondence, see Table 2.2. He also lamented the fact that researchers have traditionally concentrated on process parameter control while neglecting process tuning. Finally, Hardt urged control engineers to "get more involved" in applying control tech-

---

\(^1\)Blank holder force is explained in Chapter 6.
Desired Geometric Properties \[ \rightarrow \] Controller \[ \rightarrow \] Machine and Material Processing \[ \rightarrow \] Manufactured Part’s Geometric Properties

Processing Time, $T_p$

Feedback Delay, $T_f$

Figure 2-2: This diagram shows the essential components of process tuning (process output control). Note the time constants of the process and the feedback delay, $T_p$ and $T_f$ respectively.

Table 2.2: Table of the basic process tuning objectives and the corresponding control system objectives, according to Hardt (1993).

<table>
<thead>
<tr>
<th>Manufacturing Objectives</th>
<th>Control Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>• minimise production variations for a given geometry and property target in the face of material and processing environment uncertainties (namely quality)</td>
<td>• match output to input when there is noise within the system (disturbance rejection) even if there are plant uncertainties (parameter insensitivity)</td>
</tr>
<tr>
<td>• respond rapidly to demand changes (flexibility)</td>
<td>• high bandwidth response to new inputs (tracking)</td>
</tr>
<tr>
<td>• maximise productivity (rate)</td>
<td>• maximise stable bandwidth</td>
</tr>
</tbody>
</table>
niques to manufacturing systems. Even today process tuning remains only a very small part of manufacturing research. This may have occurred for two reasons: first, the lack of geometric data available; and second, many manufacturing processes are parallel processes. The first reason has come about because there has not been the sensory data available to provide timely and efficient information about the manufactured parts. This will change in the future with better sensor technology and computer vision algorithms becoming available. Parallel processes are those which simultaneously transform all regions of a work-in-progress part. Parallel processes are hard to control because the feedback does not correspond to a localised region of the work-in-progress part, whereas in-process control need only affect one process parameter. There is also a problem of complexity because of the non-linear nature of many manufacturing processes, and this is compounded by the elaborate interactions between each of the regions of the part.

2.5.3 Approaches to process tuning

Hardt et al. (1982) developed the first closed control loop system for shape manufacturing processes. They created a control system that predicted the amount of springback in a three roller, roll-bending process. They developed a mechanical device to measure the amount of curvature (shape error) in the rolled strip. Larrabee and Stelson (1987) were the first to introduce vision analysis (imaging), shape error and control in a simple system that analysed non-homogeneous deformation of metallic cylinders.

This was superseded by Webb and Hardt's (1991) seminal work which established a closed-loop process tuning system for a sheet metal forming process. The system changed the shape of the punch in response to an analysis of the shape error of the final part. They used Fourier descriptor deformable models to describe the desired shape and the measured shape. In addition, they used an innovative reconfigurable die that enabled the punch shape to be modified quickly. The Fourier parameters for the punch and manufactured shape were combined into a linear equation to form the deformation transfer function. This function transformed the punch shape into the manufactured shape, and it could also be inverted to give the punch shape given a manufactured shape. The deformation transfer function measures the springback of a formed part, and it aids the convergence of the control system to find the correct punch shape. Later in this chapter we will discuss whether the Fourier description of the shape is the best available. In addition, with an increasing amount of non-linear methods available, can the linear deformation transfer function be improved upon?

Webb and Hardt's system assumed that they had no material behaviour knowledge about the part prior to manufacturing. Their closed-loop control system would implicitly learn the material behaviour through the deformation transfer function. Karafillis and Boyce (1992; 1996) developed a sheet metal forming control system similar to Webb and Hardt, however Karafillis and Boyce assumed that they had a complete knowledge
of the material behaviour. They used a finite element model of the sheet metal forming process to determine the residual force on the final part. The force profile on the final part was used as feedback information if the shape of the manufactured part was not within the desired geometric tolerances. The material model enabled the force profile to be converted into the necessary changes of the punch’s profile to obtain the desired shape. In essence, this control system is also developing a springback predictor.

The final major process tuning system reviewed is that of Yang and Manabe et al. (1997; 1998; 1999). They also developed a control system for a sheet metal forming process. The structure of the control system can be seen in Figure 2-3. An artificial neural network was used to identify material properties and coefficient of friction values from in-process values of punch force and depth. This information was passed to a fuzzy model that provided the appropriate variation of the influential process parameters to form the desired shape. These are the essential components of the in-process controller. The final part was then evaluated to determine if the fuzzy model or the off-line database needed to be updated. Unfortunately, the post-processing evaluation and pre-processing database were not clearly defined by Manabe et al. This control system is a hybrid system because mainly it is a process parameter control system, but it does include feedback from evaluating the finished part.

---

**Figure 2-3:** The diagram shows the basic set-up of Yang and Manabe et al. (1997; 1998; 1999) process control system. The control system is divided into three main sections: pre-process, in-process, and post-process. This system combines an in-process control system with process tuning using adaptive feedback after analysing the finished part.
2.5.4 Discussion of process tuning

There is a lack of process tuning systems within the control literature for manufacturing. The main reason for this is the difficulty in conducting parallel process control. It is much easier to attempt serial process control or process parameter control where only a few variables are monitored. The problem with parallel processes is how to simultaneously monitor all the modified regions of a work-in-progress part. This is a problem of dimensionality, where a large number of variables are needed to monitor all the modified regions of the part. This thesis uses a deformable model, the \textit{Point Distribution Model}, to filter the shape variation data to return only a minimal number of quantifiable monitoring variables. This reduces the dimensions of the problem to only the principal components of variation.

The second issue that has restricted process tuning is the ability to get timely information from both the work-in-progress and finished parts. The shape of the work-in-progress part is difficult to obtain and currently this problem is intractable until developments are made with in-die sensors. However, the ability to obtain timely and accurate data from the finished part is becoming increasingly probable.

The final issue of process tuning is the interpretation of the geometric error from a finished part with regard to automatic control. This involves converting the error into tuning information for the manufacturing process. Webb and Hardt (1991) and Karafillis and Boyce (1992) developed inverse models that converted difference in shape or material properties into control action information. We develop two feedback models, a shape error function and an inverse model.

2.6 Feedback models

The two feedback models developed in this thesis are a shape error function and an inverse shape variation–process parameter model. This section discusses the approaches within the literature relating to these two models. The discussion of the two models is also split into the two types of manufacturing systems that are to be investigated.

2.6.1 Shape error feedback

2.6.1.1 Shape error measures in forging

Within forging there has been very little research in regard to automatic inspection and forged parts. Wright and Bourne (1988) developed an inspection system for an open-die forged part. Cross sectional layers of the part were calculated by spinning the part around on a turn table while a computer vision system determined the boundary of the cross section. The analysis of the part was performed using a simple difference scheme. The differences found were not very accurate, but this was the first attempt
at dimensional analysis of forged parts.

Another area where shape error has been considered within forging is that of die design. Inverse models for shape optimisation have been developed such that given an end shape and the appropriate finite element model, the inverse models provide the initial profile of the die or profiles of multi-step preform dies. Fourment et al. (1996; 1996) describe the forged shapes using spline functions and the shape is optimised by moving the characteristic points of the splines. The shape error function calculates the sum of squares of the difference between the projected distance of nodes on the simulated forged shape to that of the desired shape. They updated this constraint to weight each node by the squared length of the associated segment between nodes. Zhao et al. (1997) also created inverse models to find the correct preform die design profile. The shape error function that they use sums the square of the area between discretised sections of the simulated forged shape and the desired shape. These relatively simple forms of shape error functions are useful for an optimisation process if one has a finite element model or a material model to develop an inverse model. We assume that the material model is unavailable, which is more consistent with an automatic inspection perspective. If the material model is unavailable, then a more complex measure of shape error may be necessary to develop an inverse model.

2.6.1.2 Shape error measures in sheet metal forming

There have been several approaches to measuring shape error within sheet metal forming. Karafillis and Boyce (1992; 1996) investigated two dimensional and three dimensional shape differences when devising an iterative die design system based on finding the inverse shape effect of simulated force measurements. They used the root mean square (RMS) of the vertical distances between the desired shape and the actual shape as a shape error measure. The RMS error does give an indication of the shape error. It is, however, a simple error model and may not be able to elicit anything more than the indicative amount of error in the stamped component. The major problem is that this form of shape error measure may not emphasise significant problems over insignificant problems. That is, significant errors (or the lack of) can be overwhelmed by insignificant errors because they are measured with the same importance.

Webb and Hardt (1991), previous to Karafillis and Boyce, developed an iterative die design system based on shape error only. They used the Discrete Fourier Transform (DFT) to describe the die's surface and the die's stamped output. The coefficients of the DFT were used to develop a deformation transfer function that closed the control loop between the die and the output. The DFT description of shape is able to reduce the dimensionality of the die design down to a small set of values that is much easier to manipulate. Fourier models, however, suffer the problem that discontinuities (such as corners or edges) are difficult to describe unless some smoothing is done to the surface.
being modelled.

Finally, Kase et al. (1999) devised an innovative shape error description that measures both local and global shape error. They used differential geometry to determine the principal curvature differences (Extended Gaussian Curvature) between the desired shape and the actual shape. These differences were employed to calculate the local shape error of a stamped component. The changes in the principal curvature were calculated on sample points on the surface of the component, and are divided into three different types: Mount; Valley; and Twist. The global shape error was calculated using the average surface normals on arbitrarily defined patches of the component for both the desired and actual shapes. These normals determined the global shape error by calculating the “bent” angle and the “twist” angle between the two shapes' average normal vectors. One of the advantages of this method is that the two compared surfaces do not need to be exactly aligned. As in the case for Fourier models, the surfaces of the shapes need smoothing if there are discontinuities before the shape error description can be utilised.

2.6.1.3 Discussion of shape error measures

There are three main limitations of the above shape error measures. First, there is no weighting of significant errors over insignificant errors. Secondly, in most cases the geometric data has to be low pass filtered or smoothed before processing. Finally, the basic variation modes used to describe errors by the above models may not be best suited to the particular process. These limitations will be addressed by the shape error function explained later in Chapter 4.

2.6.2 Inverse mode feedback

2.6.2.1 Inverse modelling in forging

Most of the inverse model approaches within forging are related to die design. Die design has two basic problems. First, the desired end shape has been designed and the issue is to determine the profile of the die that will produce the desired shape. Second, given the desired end shape and the profile of the final die, the issue is to determine the optimal shape of the undeformed preform billet that will eventually produce the desired end shape. These are both inverse problems where the end result is known but the initial parameters are unknown.

There have been three main approaches to these two die design problems. The first approach is to use artificial intelligence techniques to solve the inverse model. Bakhshi-Jooybari et al. (1996) devised an expert system that was able to compare new forging designs with previously encountered designs and return an assessment of the feasibility of the design. Artificial neural networks have also been used to learn the optimal parameters for a series of preform designs; they are then able to return
estimated parameters for new designs given to them (Roy et al., 1994; Kim et al., 1997; Hsu & Lee, 1997). In some of these approaches it is unclear how the shape of the initial or final forged shapes was characterised as other forging parameters were optimised.

The second approach is to use smart search techniques. There are two main forms of smart searching. Genetic algorithms have been used to find the best parameters that satisfy the desired end forging design when given an evaluation function (Roy et al., 1997; Chung & Hwang, 1998). The procedure of the genetic algorithms is as follows.

1. Determine a method that converts design parameters into “chromosomes”. Often binary strings of “1”s and “0”s are chosen to represent the different levels of the parameters.
2. Randomly select a number of starting chromosomes.
3. Repeat:
   3.1. Rank the chromosomes using the design evaluation function.
   3.2. Retain the top performing chromosomes.
   3.3. Randomly genetically cross the top remaining chromosomes.
   3.4. Add random changes to the remaining chromosome set.
   3.5. Calculate the new end forged shapes from the new chromosome set.
4. Until the global optimal point is reached.

The main issue with this approach is the number of iterations and chromosomes that must be used to find the optimal point. If a full finite element analysis has to be conducted with each chromosome and each iteration, then there may be large computing overheads associated with this approach.

The second type of search technique is that of analytical or numerical optimisation. This approach determines the parameters that produce the desired end shape by minimising an objective function. Kusiak (1990) was the first to combine die design and shape optimisation. Fourment et al. (1996; 1996), as previously stated, established a shape optimisation method that minimised the distance between the desired and the end forged shape. An total energy function was also included within the objective function to ensure uniqueness in the optimised parameters. The general procedure for this approach is as follows.

1. Define the initial shape of the billet. Hence, define the control points of the spline fitting the boundary of the initial billet.
2. Calculate the normal direction to the spline curve at the control points.
3. Apply the optimisation algorithm
3.1. Simulate forging process

3.2. Adjust the spline according to the gradient directions of the optimisation and constraint functions

3.3. Has the end tolerance or iteration limit been reached? If yes, then finish optimisation loop. Otherwise, continue processing.

4. If the end tolerance level has not been reached, return to step 2. where the current optimised shape is now the new initial guess.

Zhao et al. (1997) implemented a very similar approach except they had a different shape error measure, see Section 2.6.1.1.

The final approach is to use reverse simulation methods. In this case, the simulation begins with the end shape and the die velocity is reversed. Careful attention must paid to boundary conditions and equating internal and external forces. Park et al. (1983) established the backward tracing scheme which started with the desired design shape and proceeded to go backwards to predict the initial profile. Chang and Bramley (2000) recently have developed the reverse simulation by applying the upper bound elemental technique (UBET) to minimise the rate of energy dissipation in the system.

2.6.2.2 Inverse modelling in sheet metal forming

There are three main types of inverse models within sheet metal forming. The first type of inverse model determines initial material parameters that were necessary to form the final formed shape. Gehn and Ghouati (1995; 1996; 1998; 2001) use the output of a finite element model to compare against a measured actual part. By using optimisation techniques, where the shape error is the objective function, they determine the optimal parameters that caused the measured part. They also developed analytical gradients for the objective function for both implicit and explicit finite element simulations.

The second type of inverse model determines the contour of the initial undeformed flat sheet or blank. The contour is the top-view shape of the flat blank. In this case, the desired final formed shape is known, but the shape of the initial contour of the blank is unknown. The shape of the initial flat blank is important because too much material can cause wrinkling and too little material can lead to tearing or thickness problems. There were four early methods that determined the blank contour: slip-line method (Karima, 1989); plane-stress (Vogel & Lee, 1990; Chen & Sowerby, 1992); geometric mapping (Sowerby et al., 1986; Blount & Stevens, 1990); and ideal forming (Chung & Richmond, 1992a; Chung & Richmond, 1992b; Chung & Richmond, 1994). These methods have been superseded by two methods that are based on deformation theory: one-step analysis and the inverse approach.

2Implicit and explicit finite element modelling is defined in Chapter 7.
One-step analysis (Liu & Karima, 1992; Lee & Huh, 1997; Yang & Nezu, 1998) and its extension, multi-step analysis (Lee & Huh, 1998) calculate the initial shape of the blank using a one-step reverse simulation. This single step minimises the virtual work of the system using deformation theory. This includes the external forces such as friction and blank holder force. The multi-stage analysis was developed because the deformation theory should really be restricted to only infinitesimal deformations. Deformation theory is only an approximation for large deformation examples. This meant that large errors were accumulated in the one-step analysis for complex parts. The error is reduced by increasing the number of steps in the inverse analysis.

Guo and Batoz et al. (1992; 2000) developed an inverse model also based on deformation theory called the inverse approach. The inverse approach exploits the geometric of known final desired shape and implements an iterative scheme to determine the original position of each material point in the flat blank. One of the major problems with these deformation theory based models is that they do not consider springback. This means that these models rely on the die shape producing the correct part.

The final method to determine the contour of the initial flat blank is that of Park et al. (1999). They established a method that combined ideal forming theory with shape optimisation. Ideal forming theory (Chung & Richmond, 1994) was used to determine a first estimate of the initial blank contour. The optimal shape was then found by applying shape optimisation to a finite element model of the system. The shape error used in this case was the root mean square distance between the edge contour of the desired final formed shape and the current iteration's edge contour.

The third type of inverse model determines the shape of the die to create the desired shape. As has been previously discussed, Webb and Hardt (1991), and Karafillis and Boyce (1996; 1992) developed methods to determine the shape of a die to give the desired end shape. This was an attempt to predict the effects of springback to create the appropriate final part shape.

2.6.2.3 Discussion of inverse modelling

Inverse models are created to return information about an initial state of a process given a final desired state. Often this is conducted using simulations where the material models are known for the process. In the past, industry has approached this "inverse modelling" by experienced operators implementing solution based on empirical evidence. These operators analysed their previous experiences to enhance the process in question. This was achieved without explicitly knowing the material models. We assume in this thesis that the material models may not be available. This begs the question whether one can characterise shape variation in a process to learn enough information to provide relevant feedback.
2.7 Summary

In summary, tolerances provide the specifications for the limits of allowable variations in a manufactured part. These specifications are only a guide for the manufacturing process. The important information, from a manufacturing control context, is the error between the measured and desired parts. This led to investigating dimensional inspection because it is a combination of tolerancing and automated inspection. Dimensional inspection examines the differences between the dimensions of the measured and desired parts, and these differences are compared to the specified tolerances. Often dimensional inspection is only concerned with whether a part is contained within its tolerance zone. This raised the question of what is the most appropriate form of characterising the error. This thesis characterises the difference between the measured and desired parts (shape error) using a deformable model. This gives the advantage of being able to quantify both global and local errors. Unfortunately, there are not many automated inspection systems that use deformable models which is surprising given their success in medical image processing. In addition, there is an opportunity to apply this form of characterising shape error on shape manufacturing processes. Kase (1999) suggested that there is currently a lack of methods to determine shape error problems in sheet metal forming. We will also concentrate on sheet metal forming and also forging. Within manufacturing control there is a lack of process tuning systems. Process tuning is where the output of the manufacturing process is monitored and controlled. The dearth of process tuning systems has occurred because many manufacturing processes are parallel processes. Parallel processes are those which concurrently modify all regions of the mid-formed part. This thesis aims to provide a solution to this large dimensioned problem of monitoring complex surfaces by reducing the problem down to the minimum principal components of variation. Finally, process tuning raises the problem of how to interpret the geometric error from a part. Previous methods have determined a form of inverse model that converts the error into something meaningful for the controller. We will investigate developing an inverse shape variation–process parameter model for the same reason.

Therefore, this thesis develops a deformable model that characterises geometric errors in shape manufacturing parts. This description of error is interpreted by an inverse model to provide feedback about the process parameters. This is the shape manufacturing feedback model.

Before the shape manufacturing feedback model is discussed in detail, however, it is necessary to describe the methods used throughout the thesis.
Chapter 3

Methods

Shape Odyssey 2001 continues

3.1 Introduction

This chapter describes the key methods used throughout this thesis in regard to shape description and pattern recognition techniques. The shape description section describes the major deformable models available and the reasons why the Point Distribution Model (Cootes et al., 1995) was eventually chosen. This is followed by a brief review of the types of pattern recognition techniques used in this thesis. In addition, some data fitting techniques that are used in the results chapters are outlined. Finally, experimental design is briefly described in the last section.
3.2 Shape description methods

3.2.1 Deformable models in computer vision

Deformable models are used in computer vision to track objects which vary in shape over time. Medical researchers have also used deformable models to simulate organs which can vary in shape over time (Nastar & Ayache, 1996). Inspection systems have used deformable models as flexible templates when matching images (Kass et al., 1987; Nastar et al., 1997).

According to Cootes et al. (1995), there are six basic types of two dimensional (2-D) flexible models or deformable templates:

- hand crafted models;
- articulated models;
- active contour models ("Snakes");
- Fourier series model;
- statistical shape models;
- finite element models.

The active contour models, Fourier series models and finite element models are briefly discussed because they are the main alternative models to the chosen model, the Point Distribution Model (Cootes et al., 1995) which is a statistical shape model. The Point Distribution Model is described in Section 3.2.2.

3.2.1.1 Active contour models ("Snakes")

Kass et al. (1987) first developed Active Contour Models ("Snakes") which are flexible contour models that are attracted to image features. The basic snake model uses splines, acting under internal and external forces, to deform around an object that it is attracted to within an image. This is achieved by minimising the following equation,

$$ E_{\text{snake}}^* = \int_0^1 (E_{\text{internal}}(v(s)) + E_{\text{image}}(v(s)) + E_{\text{constraint}}(v(s))) \, ds, \quad (3.1) $$

where $v(s)$ is the spline, $E_{\text{internal}}$ is the internal energy of the snake, $E_{\text{image}}$ is the image energy, and $E_{\text{constraint}}$ is the external constraint energy.

The internal energy is calculated as follows,

$$ E_{\text{internal}} = \frac{\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2}{2}, \quad (3.2) $$

where $\alpha(s)$ controls the elasticity of the snake, and $\beta(s)$ controls the smoothness of the snake.
The image energy is a weighted sum of image energy functionals.

\[ E_{\text{image}} = \omega_{\text{line}} E_{\text{line}} + \omega_{\text{edge}} E_{\text{edge}} + \omega_{\text{term}} E_{\text{term}}, \]  

(3.3)

where \( E_{\text{line}} \) is the line energy functional that attracts the snake to light or dark lines, \( E_{\text{edge}} \) is the edge energy functional that attracts the snake to large image gradients, and \( E_{\text{term}} \) is the terminal energy functional that attracts the snake towards line terminations and corners.

The external constraint energy is an interactive constraint system that enables the user to assist the snake wrap around an object. The two main constraints the user can apply are springs and volcanoes. A spring creates an attractive force between two points, and a volcano is a repulsive force that pushes the snake away from a region in the image.

The main problems with snakes is that they are not automatic in implementation and the shape variation is not described in a quantitative fashion. The snakes are not automatic because the user has to supply external constraint conditions, and also the \( \alpha(s) \) and \( \beta(s) \) parameters have to be defined. Secondly, the snake is able to wrap around the boundary of the object but it does not describe the variation in shape from the desired shape in a quantitative way. Snakes are useful for image segmentation and for finding objects where the shape is generally not known. However, this thesis has the underlying assumption that the general shape of the part is known and it is the variations from this state that are important.

### 3.2.1.2 Fourier series shape models

Scott (1987) proposed modelling shapes using Fourier descriptors. Staib and Duncan (1992) extended this to using elliptic Fourier descriptors, see equation (3.4).

\[
\begin{bmatrix}
    x(t) \\
    y(t)
\end{bmatrix} = \begin{bmatrix}
    a_0 \\
    b_0
\end{bmatrix} + \sum_{k=1}^{\infty} \begin{bmatrix}
    a_k & b_k \\
    c_k & d_k
\end{bmatrix} \begin{bmatrix}
    \cos(kt) \\
    \sin(kt)
\end{bmatrix}
\]  

(3.4)

where

\[
\begin{align*}
    a_0 &= \frac{1}{2\pi} \int_0^{2\pi} x(t) \, dt \\
    c_0 &= \frac{1}{2\pi} \int_0^{2\pi} y(t) \, dt \\
    a_k &= \frac{1}{\pi} \int_0^{2\pi} x(t) \cos(kt) \, dt \\
    b_k &= \frac{1}{\pi} \int_0^{2\pi} x(t) \sin(kt) \, dt \\
    c_k &= \frac{1}{\pi} \int_0^{2\pi} y(t) \cos(kt) \, dt \\
    d_k &= \frac{1}{\pi} \int_0^{2\pi} y(t) \sin(kt) \, dt
\end{align*}
\]

Therefore, the coefficients of the elliptic Fourier descriptors, parameters \( \{a_0, c_0, a_1, b_1, c_1, d_1, \ldots\} \), determine the final shape of the deformable model. Staib and Duncan (1992) developed statistical distributions for the parameters that described a training set of objects. These distributions were then used to classify whether a new example object was of the same class as the training set.

The advantage of the Fourier description is that it can describe the shape variation away from a desired shape by breaking down the variation in harmonic sinusoidal...
modes. Unfortunately, Fourier descriptors find it difficult to describe discontinuities, such as corners. Typically, many harmonics are needed to approximate a corner of an object because the sinusoid basis functions for the Fourier descriptors are smooth. Another issue with the Fourier descriptors is that the basis sinusoidal functions may not be the best set of basis functions to describe the variation in shape. Often there is not an intuitive link between the Fourier coefficients and the actual shape variation.

### 3.2.1.3 Finite element models (FEMs)

Pentland (1990) and Pentland and Sclaroff (1991) developed a finite element deformable model approach to modelling image objects. They began with the standard physical finite element model equation which balances the internal and external forces, but it is modified to include object nodal displacements,

\[ M\ddot{U} + C\dot{U} + K\dot{U} = \dot{R}, \]  

where \( \ddot{U} \) is a \( 3n \times 1 \) vector of the \((\Delta x, \Delta y, \Delta z)\) displacements of the \(n\) nodal points relative to the object’s centre of mass, \(M\), \(C\), and \(K\) are the \(3n \times 3n\) matrices describing the mass, damping and material stiffness between each point within the object, and \(\dot{R}\) is a \(3n \times 1\) vector describing the \(x\), \(y\), and \(z\) components of forces acting on the nodes.

The assumption is made that the inertia and damping effects are minimal which gives a simplified equilibrium condition,

\[ K\dot{U} = \dot{R}. \]  

The stiffness matrix, \(K\), is calculated directly from integrating a combination of the strain-displacement and material matrix,

\[ K = \int_V B^T E B \, dV, \]

where \(B\) is the strain-displacement matrix, and \(E\) is the material matrix. See Bathe (1982) for more details on the FEM equations and integrals.

Once the stiffness matrix has been calculated, modal analysis is conducted on the model. The modes are optimally the eigenvectors of the stiffness matrix. Therefore, the nodal displacements are,

\[ \ddot{U} = \Phi \ddot{u}, \]  

where \(\ddot{u}\) is the modal transformed displacement vector, and \(\Phi = [\Phi_1, \Phi_2, \ldots, \Phi_N]\) is the modal transformation matrix made up of the eigenvectors, \(\Phi_i\), of the stiffness matrix. The modal analysis enables some of the insignificant modes to be discarded to simplify processing and to remove noise.

The shape variation description is, therefore, reduced to a weighting vector and set of vibration modes. This is advantageous because it allows the variations in shape to
be quantified. The difficulties with this method are the ability to obtain the stiffness matrix and the lack of statistical variation in the calculated FEM variation modes. The stiffness matrix can be difficult to ascertain particularly if the material properties are not known. The finite element models do not take into account any statistical variation within the material or the process as the variation modes are based upon a fixed material matrix.

### 3.2.2 Point distribution model (PDM)

The Point Distribution Model\(^1\) (PDM) is a statistical deformable model which underpins the outcomes of this thesis. The PDM compares the variation of points on the boundary of shapes within a set of training shapes. This is performed by comparing each coordinate of each point versus the coordinates of every other point on each shape and across the training set in the form of a covariance matrix. The PDM then uses principal component analysis to reduce the dimensionality of the covariance matrix. The resulting vectors are the major modes of variation of the coordinates of the shapes in the training set.

#### 3.2.2.1 PDM Background

The PDM was developed by Cootes et al. (1995) as a novel deformable model that could be used for pattern recognition in computer vision. They used the PDM to visually examine the variation of resistors on a circuit board. Cootes et al. (1992) also used the PDM to describe modes of variation in two dimensional (2-D) data when examining heart images. Hill et al. (1992; 1993) extended this model to three dimensional (3-D) data to also analyse heart data. Finally, Daniel et al. (1997) used the PDM to determine whether 3-D turbine blades contained a series of production errors (bow, twist, or skew). This work used a C4.5 decision tree to obtain an average classification error of 22.5%. Daniel et al. (1997) also used the PDM to find the major modes of variation from 2-D forging data, but did not investigate the relationship between the shape variation modes and the set-up parameters.

#### 3.2.2.2 Point distribution

The PDM method begins with the training set of objects that contains examples of the types of shapes to be examined. The training set should include all of the variations that need to be recognised. Each shape in the set is then labelled with a series of points that are recognisable and are consistent across the whole training set. Cootes et al. (1995) examined a resistor board and an example of labelling a resistor is given in Figure 3-1. They also suggest that there are three main types of label points:

\(^1\)The Point Distribution Model is also known as the Active Shape Model (ASM)
Assigning Points to a Resistor

Figure 3-1: This diagram shows the distribution of 18 points on a resistor similar to the point distribution described by Cootes et al. (1995).

Type 1: points marking parts of the object with particular application dependent significance;

Type 2: points marking image or object dependent features, such as corners or upper most points of the object;

Type 3: points that can be interpolated between points of type 1 and 2.

A shape in the training set can be then represented by a vector of the coordinates of the labelled points, \( \mathbf{X} = (x_1, y_1, z_1, \ldots, x_k, y_k, z_k)^T \) where the \( i^{th} \) point is \((x_i, y_i, z_i)\). Note that this vector is one dimensional because all three components are combined into the one vector.

3.2.2.3 Alignment of the training set

The shapes in the training set are then aligned to remove any registration errors. This is achieved by minimising the weighted sum of squared distances between equivalent labelled points by scaling, rotating, and translating each shape in the training set against a reference shape. Cootes et al. (1995) minimise the following equation,

\[
E_j = \left( \hat{X}_{\text{ref}} - M(s_j, \theta_j)[\hat{X}_j] - \hat{t}_j \right)^T W \left( \hat{X}_{\text{ref}} - M(s_j, \theta_j)[\hat{X}_j] - \hat{t}_j \right),
\]

where \( E_j \) is the error of the \( j^{th} \) shape, \( \hat{X}_{\text{ref}} \) is the reference shape, \( \hat{X}_j \) is the \( j^{th} \) shape in the training set, \( M(s, \theta)[\hat{X}] \) is the rotation (\( \theta \)) and scaling (\( s \)) matrix applied to shape \( \hat{X} \), \( \hat{t} \) is the translation vector, \( W \) is a diagonal matrix of weights for each point. The weighting matrix, \( W \), is calculated in the following way. First, the distance between points \( k \) and \( l \) is calculated for a shape, defined as \( R_{kl} \). Second, the variance in this distance over the set of shapes is determined, defined as \( V_{R_{kl}} \). The diagonal weight element is calculated is follows,

\[
w_k = \left( \sum_{l=1}^{n} V_{R_{kl}} \right)^{-1}.
\]
This weight favours the points that vary very little within the training set.

The following algorithm is then used to align the shapes.

1. Rotate, scale, and translate each of the shapes in the training set to align to the first shape of the training set.

2. Repeat:
   2.1. Calculate the mean of the aligned shapes.
   2.2. Adjust the mean to a default scale orientation and reference point.
   2.3. Rotate, scale, and translate each of the shapes in the training set to match the adjusted mean shape.

3. Until the algorithm converges

3.2.2.4 Principal component analysis

A covariance matrix is then calculated for the variations of the aligned shapes from the mean shape of the training set, see equation (3.9). This covariance matrix contains the variations of each coordinate with respect to every other coordinate across all the shapes in the training set.

\[ S = \frac{1}{N-1} \sum_{i=1}^{N} (\hat{X}_i - \bar{X}) (\hat{X}_i - \bar{X})^T \]  

(3.9)

where \( \hat{X}_i = [x_{i1}, y_{i1}, z_{i1}, \ldots, x_{ki}, y_{ki}, z_{ki}]^T \) is the \( i^{th} \) shape coordinates and \( \bar{X} = \frac{1}{N} \sum_{i=1}^{N} \hat{X} \) is the mean shape of the training set. Moreover, \((x_{ij}, y_{ij}, z_{ij})\) is the \( i^{th} \) boundary point on the \( j^{th} \) shape in the training set.

Finally, finding the eigenvectors of the covariance matrix \( S \) realises the principal components of the training set,

\[ S \hat{P}_k = \lambda_k \hat{P}_k, \]  

(3.10)

where \( \lambda_k \) is the \( k^{th} \) eigenvalue of \( S \), \( \lambda_k \leq \lambda_{k-1} \), and \( \hat{P}_k \) is the \( k^{th} \) eigenvector of \( S \). The eigenvectors are sorted in descending order according to their corresponding eigenvalues.

Thus, the most significant principal components are the first listed eigenvectors. By rule of thumb, the \( t \) most significant principal components that explain at least 90% of the variations are chosen and placed into matrix \( P \),

\[ P = \left( \hat{P}_1 \ \hat{P}_2 \ \ldots \ \hat{P}_t \right). \]  

(3.11)

Therefore, any shape in the training set can be well represented by,

\[ \hat{X}_i = \bar{X} + P \hat{b}, \]  

(3.12)

where \( \hat{b} \) is the weighting vector that indicates how much each principal component contributes to vary the mean shape into the shape \( \hat{X}_i \). Notice that each eigenvector represents one of the major shape variation modes of the training set.
3.2.2.5 Advances in the PDM

Non-linear PDM models

There is a problem with the PDM when the distributions of the \( \vec{b} \) vector weights are not Gaussian. The PDM relies upon the variation from the mean shape occurring with a uni-modal distribution. A uni-modal distribution implies that the \( \vec{b} \) vectors form convex sets where any \( \vec{b} \) vector on a line between two other \( \vec{b} \) vectors is at least as valid or likely as the two \( \vec{b} \) vectors. If the set of valid \( \vec{b} \) vectors is not convex, then the linear principal components of \( S \) no longer represent the best basis vectors of the variation space.

Sozou et al. (1994; 1995) developed two types of non-linear PDMs. The first non-linear PDM (Sozou et al., 1994) uses polynomial regression to find the non-linear principal components. Polynomial regression iteratively calculates the successive most likely polynomial paths that the \( \vec{b} \) vectors take. The second method (Sozou et al., 1995) uses an artificial neural network to find the non-linear principal components. This method was found to be the least restrictive of the two methods.

Grey level models

Cootes et al. (1996; 2001) extended the PDM approach to model grey level appearance within images. This enabled the location and identification of objects whose shape varied marginally but whose grey levels varied markedly. This is achieved by looking at the grey level of chosen patches from a training set of objects within a set of images. Each \( n_x \) by \( n_y \) patch of pixels is converted into a 1-D vector, \( \vec{G} \), of pixel grey levels. The PDM method is then applied to the \( \vec{G} \) vectors of pixel grey levels.

Combined FEM/PDM models

Cootes et al. (1994) noticed that their end PDM equation (3.12) was similar to the FEM equation (3.7). They discovered that the FEM can be used to artificially increase the PDM training set by modifying the training set’s covariance matrix, \( S \).

The modified covariance matrix becomes,

\[
S' = S + \alpha \left( \frac{1}{m} \sum_{i=1}^{m} \Phi \Lambda \Phi^T \right),
\]

where \( S \) is the PDM covariance matrix, \( \alpha = \alpha_1/m \) is the proportion of FEM variation included in the modified covariance matrix, \( m \) is the number of samples in the training set, \( \Phi \) is the modal transformation matrix, \( \Lambda \) is a diagonal matrix whose diagonal elements are the variances, \( \lambda_i \), of \( \vec{u} \) the modal transformed displacement vector. To weight the \( \Lambda \) matrix appropriately the variance \( \lambda_i = \omega_i^{-2} \), where \( \omega_i \) is the frequency of the \( i^{th} \) FEM mode.

The modified covariance matrix enables small training sets to be used until more samples can be found. The contribution of the FEM model is decreased as the number
of samples in the training, \( m \), is increased as this decreases \( \alpha \). The \( P \) matrix is found as before by applying principal component analysis to the modified covariance matrix.

### 3.2.2.6 Advantages

The advantage of using the PDM method is two fold. First, the PDM method is able to take a set of objects and quantify the major modes of variation. These modes can be used to quantify the shape variation of other objects. The second advantage of the PDM is the ability to quantify the statistical variation of a set of objects. The FEM model is able to separate objects into quantifiable modes of variation, but it is unable to take into account the statistical variation unless this is known \textit{a priori} and is included into the FEM model.

### 3.2.2.7 Disadvantages

The main disadvantage of the PDM method is that it is an inductive approach. The PDM method relies on having a valid set of training examples already existing. This is not the case if one was seeking to set-up a predictive model or do some preliminary design work. In addition, the PDM method relies on the training set having all the necessary shape variations contained within it. If all the desired shape variations are not captured in the training set, then the basis eigenvectors that make up the \( P \) matrix will no longer span the vector space of valid shapes.

### 3.3 Pattern recognition methods

This section describes the pattern recognition techniques used throughout this thesis. Watanabe (1985) defines a pattern “as opposite of a chaos; it is an entity, vaguely defined, that could be given a name.” This statement shows the difficulty in automatic pattern recognition. Humans intuitively know what patterns are, and they can know when they see one. However, describing what a pattern is and understanding how humans recognise patterns has been an active area of research for over half a century.

Duda and Hart’s (1973) seminal work separates the pattern recognition problem into three distinct components: a transducer, a feature extractor, and a classifier. The transducer gathers the data from the environment and passes it to the feature extractor. The feature extractor filters the data to find the relevant features of the data. The classifier interprets the filtered data to give a decision about which class it believes the data is from. A class is a group of patterns, objects in this thesis' case, with something in common. A class can be specified, supervised classification, or learnt from the data, unsupervised classification. This thesis will only conduct supervised classification.

There are three main approaches to pattern recognition (Schalkoff, 1992): statistical, syntactic, and artificial neural networks. Statistical classifiers develop statistical
decision boundaries using probability distributions to map the boundaries of each class. Syntactic or rule based pattern recognition uses hierarchical grammar rules to describe complex pattern classes. Artificial neural networks are massively parallel computer algorithms made up of simple processing units; they are able to learn complex relationships between the input data and the distinct pattern classes. This thesis only considered two forms of classifier, statistical and artificial neural networks, as they were deemed appropriate for the situations that arose.

3.3.1 Statistical classifiers

Statistical classifiers rely on having a good knowledge of the probability distributions of each pattern class. The flow chart of designing a statistical classifier is shown in Figure 3-2. There are two main types of statistical classifiers, parametric and non-parametric classifiers. Parametric classifiers have their probability distributions defined \textit{a priori}, and only the parameters of the distribution need to be quantified (usually mean and variance). Non-parametric classifiers do not have predefined probability distributions. In this case, the probability distributions have to be estimated using density estimation techniques, such as Parzen windows or \(k\)-nearest neighbours (Fukunaga, 1990). This thesis only considers parametric classifiers with Gaussian distributions. Once the probability distributions have been defined, a classifier is then chosen that minimises the probability of a misclassification.

The basic statistical classifier is constructed as follows. There are two classes \(\{\omega_1\) and \(\omega_2\). Therefore, a gathered piece of data, \(X\), can be classified as \(\omega_1\) or \(\omega_2\) according to which ever result is more likely,

\[
\text{if } P[\omega_1|X] > P[\omega_2|X] \text{ choose } \omega_1; \\
\text{otherwise choose } \omega_2, \tag{3.14}
\]

where \(P[\omega_i|X]\) is the conditional probability of the \(i^{th}\) class being correct given \(X\). The Bayes rule enables the conditional probability to be expressed as,

\[
P[\omega_i|X] = \frac{p[X|\omega_i]P[\omega_i]}{p[X]}; \tag{3.15}
\]

\[
p[X] = \sum_{j=1}^{2} p[X|\omega_j]P[\omega_j],
\]

where \(p[X|\omega_i]\) is the conditional probability density function of sample \(X\) occurring
given class \( \omega_i \). Thus, the classification condition can be changed to a Bayes condition,

\[
\text{if } p[X|\omega_1]P[\omega_1] > p[X|\omega_2]P[\omega_2] \quad \text{choose } \omega_1;
\]

\[
\text{otherwise } \quad \text{choose } \omega_2
\]

or

\[
\text{if } \frac{p[X|\omega_1]}{p[X|\omega_2]} > \frac{P[\omega_2]}{P[\omega_1]} \quad \text{choose } \omega_1;
\]

\[
\text{otherwise } \quad \text{choose } \omega_2.
\]

This changes condition (3.14) from being a posteriori to a priori. Equation (3.16) is called the likelihood ratio. By taking a minus-log of this equation one obtains the discriminant function, \( h(X) \),

\[
h[X] = -\ln p[X|\omega_1] + \ln p[X|\omega_2] \leq \ln \frac{P[\omega_1]}{P[\omega_2]}.
\]

3.3.1.1 Quadratic Gaussian classifiers

The Quadratic Gaussian classifier assumes that the conditional probability density functions are Gaussian with a mean \( M_i \) and covariance \( \Sigma_i \) for each class. Substitut-
ing the Gaussian or normal densities into equation (3.17), one obtains the following discriminant function,

\[ h[\bar{X}] = \frac{1}{2} (\bar{X} - \bar{M}_1)^T \Sigma_1^{-1} (\bar{X} - \bar{M}_1) - \frac{1}{2} (\bar{X} - \bar{M}_2)^T \Sigma_2^{-1} (\bar{X} - \bar{M}_2) + \frac{1}{2} \ln \left| \frac{\Sigma_1}{\Sigma_2} \right| \leq \ln \frac{P[\omega_1]}{P[\omega_2]} . \]  

(3.18)

For multi-class problems this discriminant function becomes,

\[ \text{Min}_i \left[ \frac{1}{2} (\bar{X} - \bar{M}_i)^T \Sigma_i^{-1} (\bar{X} - \bar{M}_i) + \frac{1}{2} \ln |\Sigma_i| - \ln P[\omega_i] \right] . \]  

(3.19)

This discriminant function is implemented in MATLAB and used in the experimental sheet metal forming chapter, Chapter 6.

3.3.1.2 Linear Gaussian classifiers

The Bayes linear Gaussian classifier assumes that the covariance matrices of all the classes are the same, \( \Sigma_1 = \Sigma_2 = \Sigma \). The Bayes decision then reduces to,

\[ (\bar{M}_2 - \bar{M}_1)^T \bar{X} + \frac{1}{2} (\bar{M}_1^T \bar{M}_1 - \bar{M}_2^T \bar{M}_2) < \ln \frac{P[\omega_1]}{P[\omega_2]} \text{ choose class 1;} \]

\[ \text{otherwise choose class 2.} \]  

(3.20)

The multi-class discriminant function becomes,

\[ \text{Min}_i \left[ -\bar{M}_i^T \bar{X} + \frac{1}{2} \bar{M}_i^T \bar{M}_i - \ln P[\omega_i] \right] . \]  

(3.21)

3.3.1.3 Fisher linear classifier

The Bayesian linear Gaussian classifier can be abstracted to form a general linear classifier with the following discriminant function,

\[ h[\bar{X}] = \bar{V}^T \bar{X} + v_0 \leq 0, \]  

(3.22)

where \( \bar{V} \) and \( v_0 \) are variables to be optimised to gain the best classifier design. The design is dependent on what criteria are used. The Fisher criterion measures the difference between the means of the two classes normalised by the averaged variance. This criterion removes the need for a threshold value, \( v_0 \), because of the subtraction of the two means. Therefore, the optimal value for \( \bar{V} \) is,

\[ \bar{V} = \left( \frac{1}{2} \Sigma_1 + \frac{1}{2} \Sigma_2 \right)^{-1} (\bar{M}_2 - \bar{M}_1) . \]  

(3.23)

This forms the basis of the Fisher linear classifier that is used in this thesis. This classifier is also implemented in MATLAB.
3.3.2 Artificial neural networks (ANNs)

Artificial neural networks are an attempt by humans to try and model our own brain. They were first pioneered by McCulloch and Pitts (1943) who investigated modelling of biological neurons through mathematical logic. Research has continued on artificial neural networks unabated and a neat summary of this history can be found in Haykin’s (1999) comprehensive text on neural networks.

The basic artificial neural network consists of a lattice of artificial neurons with directed connections (synapses) between the neurons. The artificial neurons are modelled to mimic the function of an actual biological neuron. The network is tuned such that certain connections within the lattice are reinforced while other connections are inhibited. The topology of the connections between the artificial neurons depends on the type of neural network used. If a network’s lattice connections always progress from the input layer to the output layer with no feedback between layers, then this network is defined as a feedforward network.

The artificial neural networks are best understood by understanding how an individual neuron works, see Figure 3-3. Each neuron consists of:

- weighted synapses which connect the inputs of the neuron to the summing junction;
- a summing junction that adds the weighted inputs and the bias together;
• activation function that filters the biased summed weighted inputs. This function also acts to diminish the output amplitude so that the output signal does not diverge greatly.

The input connections into the neuron are weighted by a value that indicates how much reinforcement/inhibition exists between a neuron of the previous layer and the current neuron. All the neuron's weighted input connections received from the previous layer are summed, including a bias term. This is defined as the activation energy,

\[
\text{Activation energy} : \quad \text{net}_i = \left( \sum_{j=1}^{N} w_{ij} o_j \right) + b_i, \quad (3.24)
\]

where \( w_{ij} \) is the weighting between the current neuron \((i)\) and the previous layer's \(j^{th}\) neuron, \( o_j \) is the output signal from the previous layer's \( j^{th} \) neuron, and \( b_i \) is the bias term. The activation energy is then transformed into the output signal through the activation function,

\[
o_i = \mathcal{F}_i[\text{net}_i] \quad (3.25)
\]

where \( o_i \) is the output signal of neuron \( i \), and \( \mathcal{F} \) is the activation function. The activation function can be any type of differentiable function, but often the function is either a linear function, a threshold function (relay) or a sigmoid function. Generally the activation function has some form of damping, such as saturation levels, so that the output signal of the neuron does not wildly diverge.

The neuron can then be trained to give the desired output, when supplied with particular input signals, by appropriately modifying the synapse weights. The single neuron, once trained, sets up a decision boundary according to the values of the these weights.

The single neuron comprises one unit of a network. Networks are defined by layers, see Figure 3-4. A layer is defined as a set of neurons that all have the same minimum number of connections between them and the input signals. A neural network is also defined as being \textit{fully connected} if a neuron from any layer in the network is connected to all the neurons in the previous layer. The input layer contains only the input values to the neural network. No processing or modification of the input values occur within this layer. The hidden layers are the layers of neurons in between the input layer and the output layer. The internal layers re-map the input signals and the resulting signals from the internal layers achieve a more separable or "classifiable" representation of the data. That is, the hidden layers allow the neural network to build more complex decision boundaries. The output layer is comprised of neurons that produce the final output signal(s) of the neural network. This thesis defines the structure of a network by listing the number of neurons in each layer, excluding the input layer. For example, the network in Figure 3-4 is denoted \{3,4\}. The number of inputs is defined separately.
Chapter 3. Methods

Figure 3-4: This diagram shows an example of a feedforward artificial neural network. The input layer performs no processing and just passes on the input signal values. This is why the input layer has square units as they are not neurons. The circle units are neurons. The hidden layer exists between the input and output layers. The output layer sends the output signal of the network out into the environment. This artificial neural network structure will be defined as \( \{3,4\} \), which indicates the number of neurons in each layer.

Essentially there are three types of feedforward artificial neural networks available for supervised classification of data: feedforward back-propagation networks, radial basis function networks, and support vector machines. While support vector machines are rigorous in their set-up of classification problems, finding the between class optimal hyperplane (VC dimension) can be difficult for complex problems. Radial basis functions were not considered due to time commitments. This thesis has applied the standard feedforward back-propagation networks within Chapters 5 and 6 as a non-linear classifier. Feedforward back-propagation networks are used because they are widely used within pattern classification and are easily applicable to complex problems.

3.3.2.1 Feedforward back-propagation networks

Rumelhart, Hinton and Williams (1986) first proposed the back propagation algorithm for artificial neural networks. Other researchers in the preceding decades had also developed back propagation methods but Rumelhart, Hinton and Williams were the first to see the possibilities of back-propagation with regard to machine intelligence. A feedforward artificial neural network that is trained via back-propagation is now a standard technique within the pattern recognition tool box.

The back-propagation algorithm cycles through the following steps. Each complete cycle is defined as an epoch and this is achieved when the entire training set has been
Figure 3-5: This diagram shows the basic approach of the back propagation algorithm. First an input vector is applied to the network with an associated target vector. The signal from the output layer of the network is compared with the target vector. The error is then propagated backward through the network from the output layer back to the input layer. The weights are then updated at each layer to minimise the error.

presented once for learning.

Step 1: Initialise the network and choose the network's weights from a random distribution with a mean of zero and a variance smaller than the saturation level of the activation function.

Step 2: Present an epoch of training examples to the network and perform steps 3 and 4. Note that the order of examples should be randomised from epoch to epoch.

Step 3: Feedforward or forward compute the output signal that each training example from the current epoch causes when applied to the network. Calculate the error between the final output and the desired response.

Step 4: Back-propagate the error through the network updating the weights of each neuron's connections using the delta rule:

\[
\text{Weight correction} = \left( \text{learning-rate parameter} \right) \cdot \left( \text{local error gradient} \right) \cdot \left( \text{output signal from previous layer neuron} \right)
\]

Step 5: Iterate steps 2, 3 and 4 until the stopping criterion is met.

A detailed description of the back-propagation method is provided by Haykin (1999).

3.3.2.2 Implementation of artificial neural networks

The first issue facing the implementation of an artificial neural network is the normalisation of the data. The activation functions commonly used in feedforward networks are sigmoid and hyperbolic tangent functions. Sigmoid functions are "S"-shaped functions with an output range between 0 and 1. Hyperbolic tangent functions are also
Chapter 3. Methods

Example hyperbolic tangent function

Figure 3-6: This diagram shows an example of the hyperbolic tangent function. Note the range of the function is from —1 to 1 while the domain is unlimited. In addition, the main variation of the output occurs for an input domain between —2 and 2. Beyond this subset of the domain the hyperbolic tangent function is saturated.

“S”-shaped functions but with an output range between —1 and 1, see Figure 3-6. The domains of both these functions are unlimited, however the main variation in the output from these functions occurs close to zero. Therefore, the input to the artificial neural network is normalised to stop the network from saturating. If the artificial neural network saturates, it becomes extremely difficult to train. This thesis normalises the input data to a range between 0 and 1 for sigmoid functions and —1 and 1 for hyperbolic tangent functions.

The second issue of implementation is the number of layers that is necessary to conduct accurate classification. With respect to the number of layers chosen for each artificial neural network, Kolmogrov (Hecht-Nielsen, 1987) proved that any function can be represented by a multilayer network of no more than 2 layers (one hidden layer and one output layer). The Kolmogrov representation theorem limits the number of layers needed to represent an arbitrary function, but does not give indications as to how many units are required. In most artificial neural network applications, the use of one or two hidden layers seems to provide near-optimal performance (Cybenko, 1989; de Villiers & Barnard, 1992). Despite many attempts to provide insight into determining the optimal number of internal units, most existing techniques are complicated and highly specialized to specific problems (Igelnik & Pao, 1995; Onoda, 1995). In most cases, practical design is still based on an iterative process of trial and error.

The feedforward back-propagated artificial neural networks are implemented using the neural network toolbox in MATLAB.
3.3.3 Error estimation of classifiers

Once a classifier has been initialised and trained, the next step is to determine its classification error rate. The basic process of calculating the classification error rate comprises of three parts. First, a series of training sets and a series of testing sets are created. Second, the classifiers are trained using the training sets. Finally, the classification error rate is calculated as a function of the errors of the testing sets. Usually there is only one data set from which to draw training and testing sets. This has led to the creation of clever classification error estimation techniques that use only one set of data, namely: \textit{n-fold cross validation}, \textit{leave-one-out} method, and the \textit{bootstrap} method.

The leave-one-out and \textit{n}-fold cross validation techniques are modified versions of the cross validation approach. Cross validation makes efficient use of small data sets by enabling each sample to be used within training and testing of a classifier. The leave-one-out method tests each sample in the data set using a classifier which is trained on the remaining \((N - 1)\) samples. This means that every sample is tested and the error result is not biased, however the variance of the error tends to be large (Jain \textit{et al.}, 2000). The \textit{n}-fold cross validation method splits the data into \(n\) equal sets. Each set is tested using a classifier which is trained on the remaining \(n - 1\) sets. This method has higher bias in the error rate than the leave-one-out method, but it is not as computationally intensive (Jain \textit{et al.}, 2000). The bootstrap method (Efron, 1982) creates many bootstrap testing sets by sampling the data set with replacement. The classifier is trained using all of the data set and then tested on the bootstrap testing sets. The error rate is modified to remove the bias component. Bootstrap estimates have lower variance in the classification error rate than the leave-one-out method (Jain \textit{et al.}, 2000). The bootstrap method is more computationally intensive, however, than the leave-one-out method.

This thesis uses a ten-fold cross validation method to calculate the error rate of the classifiers. The bootstrap and leave-one-out methods are too computationally intensive for the numbers of samples in the data sets used in this thesis. Computation time is an issue when training the artificial neural networks. The ten-fold cross validation ensures there are only ten networks to be trained. The training of ten artificial neural networks often takes several days to complete. The other two methods were calculated to need hundreds of trained networks to calculate the error rate. This was thought unfeasible.

The error rate gives a measure of the accuracy of the classifier, however one may want to know further information about the classifier. The classification confusion between classes can be seen by using a \textit{confusion matrix}. The confusion matrix is used to give an indication of the accuracy of the classifier with respect to the available data and, more importantly, the accuracy of the classifier between regions. The confusion matrix (Dunteman, 1984) is an \(N \times N\) contingency table of actual classes (columns) versus
classified classes (rows) where \( N \) equals the total number of classes. For this thesis each element of the confusion matrix is equal to the number of true \( j^{th} \) class samples (column) classified as an \( i^{th} \) class sample (row). This can be expressed mathematically as follows,

\[
element_{ij} = \sum \{ \text{decision } i|\text{class } j\}.
\]

The values off the diagonal represent misclassifications. Therefore, if one has a perfect classifier, then the confusion matrix would be a diagonal matrix.

### 3.4 Data fitting techniques

Gershenfeld (1999) states that data fitting has three components: model architecture, estimated parameters, and measured data. The measured data is self-explanatory. The model architecture is the structure and the complexity of the model used to fit the measured data. The estimated parameters are the parameters of the model that best fit the measured data.

There are many different data fitting techniques, for example, Multivariate Adaptive Regression Splines (MARS) (Friedman, 1991), Classification and Regression Trees (CART) (Breiman et al., 1984) and Generalized Additive Models (GAM) (Hastie & Tibshirani, 1990). This thesis, however, only considers two types of model, linear regression and artificial neural networks, as they are examples of both linear and non-linear methods. These two approaches are explained in the following two sections.

#### 3.4.1 Linear regression

Linear regression is a standard statistical technique used to model data in many different fields of research. Linear regression represents independent observations, \( y_i \), as a linear combination of \( k \) explanatory observations, \( \{ x_{i1}, x_{i2}, \ldots, x_{ik} \} \). That is,

\[
y = \beta_1 \bar{x}_1 + \beta_2 \bar{x}_2 + \cdots + \beta_k \bar{x}_k + \epsilon,
\]

(3.26)

where \( \bar{y} \) is a vector of independent observations to be modelled, \( \bar{x}_1 = [1, 1, \ldots, 1]^{T} \) is the bias term, \( \{ \bar{x}_2, \ldots, \bar{x}_k \} \) are the vectors of explanatory observations, \( \epsilon \) is the residual error between the model and the actual observations. Equation (3.26) can be expressed in a matrix format,

\[
\bar{y} = X \bar{\beta} + \epsilon,
\]

(3.27)

where

\[
\bar{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}, \quad X = \begin{pmatrix} 1 & \bar{x}_1 & \bar{x}_2 & \cdots & \bar{x}_k \end{pmatrix}, \quad \bar{\beta} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}, \quad \epsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_N \end{pmatrix}
\]
Chapter 3. Methods

There are two main assumptions made with respect to estimating the parameters $\beta$: 

- $\epsilon \sim N(0, \sigma^2 I)$, the residual errors have a mean of zero and a Gaussian variance distribution;
- $X$ is of rank $k$, the matrix of explanatory observations is linearly independent such that no explanatory observation can be explained by a linear combination of the other explanatory observations.

Given these assumptions, the best linear unbiased parameter estimate is,

$$\hat{\beta} = (X^T X)^{-1} X^T \tilde{y}. \quad (3.28)$$

There are many statistical measures that can be used to determine the accuracy of the linear regression model. This thesis only considers two measures: residual sum of squares (RSS) and the square of the multiple correlation coefficient ($R^2$).

The residual sum of squares provides a measure of the amount of error associated between the model, $X\hat{\beta}$, and the observed samples, $\tilde{y}$. The residual sum of squares is calculated as follows,

$$RSS = \epsilon^T \epsilon. \quad (3.29)$$

The square of the multiple correlation coefficient, $R^2$, gives an indication of the linearity of the data. $R^2$ ranges from 0 to 1, where a zero indicates that the data is noise and there is absolutely no linearity in the data, and a one indicates the data is perfectly linear and the model explains the observed samples perfectly. The square of the multiple correlation coefficient is calculated as follows (Johnston, 1987):

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}, \quad (3.30)$$

where ESS is the explained sum of squares, TSS is the total sum of squares. They are calculated as follows:

$$y^T A \bar{y} = \beta_{2-k}^T X_{2-k}^T A X_{2-k} \bar{\beta}_{2-k} + \epsilon^T \epsilon, \quad (3.31)$$

where $X_{2-k} = (\bar{X}_2, \bar{X}_3, \ldots, \bar{X}_k)$, $\bar{\beta}_{2-k} = (\beta_2, \beta_3, \ldots, \beta_k)^T$, and

$$A = I_{N \times N} - \frac{1}{N} \begin{pmatrix} 1 & \ldots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \ldots & 1 \end{pmatrix}$$

The linear regression models and the associated accuracy measures are all implemented within MATLAB.
Controllable input factors

Uncontrollable input factors

Figure 3-7: This diagram shows a general model of a process including factors which can affect the output of the process. Controllable factors, \{x_1, x_2, \ldots, x_p\}, are those which can be modified by the experimenter. The uncontrollable factors, \{z_1, z_2, \ldots, z_q\}, are the factors whose effects must be minimised on the responses of the process.

3.4.2 Non-linear data fitting

Artificial neural networks are also used in this thesis to perform non-linear data fitting. Artificial neural networks are good at learning (linear and non-linear) relationships between input vectors and output vectors. The artificial neural networks used are again feedforward back-propagated networks, as explained in Section 3.3.2. The difference, however, is that when the artificial neural network is implemented, the desired output values are normalised in the same way as the input data because of the limited ranges of the activation functions. The two normalisation procedures of the input and the desired output signals are conducted separately. A rectification function is created for the network’s output signals to convert the normalised output signal into an unnormalised or “raw” signal. Artificial neural networks are used to fit data models (inverse models) in Chapters 5, 6, and 7.

3.5 Experimental design

This section briefly describes experimental design and its importance to the gathering relevant information. Box et al. (1978) provide a more in-depth view of experimental design. Note, there are many other equivalent texts written about this subject. The procedure of experimental design is known as Design of Experiments (DoE) and it is applied within the later chapters of this thesis. The DoE approach ensures that an experiment is performed in an organised manner under partially controlled conditions to yield information about a process. The process can be modelled simply as a transformation of input material into an output product, see Figure 3-7. The output products, upon which the measurements are taken, are called the experimental units. Measure-
ments of the experimental units, also known as responses, must be observable, as well as affected by the factors of the process. The factors of the process can be split into two groups: controllable and uncontrollable. The controllable factors are those which can be modified by the experimenter. The uncontrollable factors are those that cannot be controlled by the experimenter. The effects of the uncontrollable factors are minimised by confounding the order of experiments. The intensity setting of each factor is called a level. A treatment is defined as a particular combination of factor levels. A DoE of the process then involves the following steps.

Steps involved in designing an experiment

1. Determine the factors to be included in the experiment. This can be conducted in a systematic way by performing screening experiments. Screening experiments are simple experiments that determine the important factors and other information about the process.

2. Decide on the number of levels to be used in the experiment. This may depend on the accuracy of the experiment's sensors. The greater the number of levels, the greater the resolution of the response.

3. Select the treatments to be performed in the experiment. It is important to structure the treatments correctly to reduce the uncontrollable effects in the experiment. In addition, it is necessary to decide on the number of experimental units (replications) that will be assigned to each treatment.

4. Determine the appropriate statistical analysis to calculate the effects of varying the factors on the response of the experimental units.

The uncontrollable factors in the experiment can be reduced in two ways. First, the experimental unit replicates can be placed into blocks where the conditions in each block are made as homogeneous as possible. This helps reduce the amount of variation in the response. Second, the random allocation of experimental unit replicates to treatments assists reducing systematic variation. If the experimental units are divided into blocks and the treatments are allocated at random within the blocks, then this is defined as a randomised block design.

3.5.1 Factorial experiments

The selection of treatments to be performed in the experiment can become complex with a high number of factors and factor levels. Factorial experiments or factorial design standardises this approach. A factorial experiment is one in which all levels of a given factor are combined with all levels of every other factor in the experiment. This structure of experiments allows the effect of high order interactions between factors to
be analysed, as well as the effect of single (main effect) factor responses. A factorial experiment is denoted by $N^k$, where $N$ is number of levels for each factor and $k$ is the number of factors in the experiment. The disadvantage of this method is the large number of experiments ($N^k$ experiments) that have to be performed to complete one factorial. If the higher order interactions are assumed to be negligible, then the number of experiments can be reduced by using fractional factorial experiments. This structure of experiments enables the calculation of the main effects and the low-order interactions.

3.6 Summary

This chapter has described the methods that are used later in this thesis. The PDM approach underpins this whole thesis as it provides a way of quantitatively describing shape variation of a manufacturing process. The pattern recognition techniques provide a way of finding a relationship between shape variation and the process parameters. This is extended to use data fitting techniques to determine these relationships. This will become more apparent after reading the next chapter which outlines in detail the main model developed in this thesis.
Chapter 4

The Shape Manufacturing Feedback Model

4.1 Introduction

This chapter discusses in detail the shape manufacturing feedback model that forms the basis of this thesis. The shape manufacturing feedback model combines shape description in the form of a deformable model and classification techniques to create a model that provides feedback on the quality of the part to the manufacturing process. As stated in Chapter 1, there are two forms of feedback from the shape manufacturing feedback model, a shape error function and an inverse model.

There have only been a few previous approaches within automated inspection that use deformable models, even though deformable models have been used to great effect in medical research. Etesami and Uicker (1985) used Fourier descriptors in their auto-
mated inspection model. Fourier descriptors are a compact way of describing shapes to
minimise the number of parameters to represent the shape or boundary profile. How­
ever, they find discontinuities, such as corners or sharp angles, difficult to represent.
This is because the basis functions of the Fourier descriptors are sinusoidal and smooth,
which take many harmonics to approximate a corner shape. This may be a problem for
certain types of parts with sharp corners. Secondly, the Fourier description does not
take account of significant shape variations caused by the manufacturing process. In
contrast, this thesis statistically analyses the output of the manufacturing processes to
determine a statistically significant set of basis shape variation modes that will better
suit the description of part shape.

Di Mauro et al. (1996) established an automatic inspection system based on the
PDM (Cootes et al., 1995). This system was then implemented on printed circuit
boards. This thesis will extend upon Di Mauro et al.’s (1996) system in two ways.
First, the scope of applications is increased to examine shape manufacturing processes,
specifically hot metal forging and sheet metal forming. Second, the type of quality
information is enhanced through the two types of feedback, shape error and estimated
process parameter levels. The reduction of the shape variation into a small number of
manufacturing process related variations potentially may give good data for classifica­
tion.

4.2 Shape manufacturing feedback model

This section reviews the shape manufacturing feedback model, restating the basics to
remind the reader of the main concepts. The shape manufacturing feedback model is
divided into two components. The first component describes the shape variation of
a part from the average manufactured shape, using a deformable model. The second
component uses this description to develop a relationship between the shape variation
and relevant knowledge about the manufacturing process (feedback stage). Figure 4-1
shows the detailed diagram of the two components where the feedback stage has been
split into the shape error function and the inverse shape variation–process parameter
model.

The shape variation description is performed by the PDM which describes variation
according to major modes of shape variation learnt from a training set of shapes. This
trained PDM will be defined as the Manufacturing Shape Variation Model.

The second component of the shape manufacturing feedback model determines the
feedback information about the manufactured part by analysing the output of the PDM
defeormable model (Manufacturing Shape Variation Model). This second part relies on
there being enough description of the process’ shape variation to find a relationship be­
tween the shape variation and the relevant manufacturing parameters. The assumption
made in the second model component is that specific examples in the PDM training
set can be generalised to represent the whole process. For this reason the choice of examples in the training set is critical. The choice of examples in this thesis is strictly governed by standard experimental design. One problem that can arise from this inductive approach is where two examples have the same shape but differing process parameter levels. In this case the manufacturing process does not have a valid inverse function, and it means that more information about the process is needed to correctly identify the feedback information.

The shape manufacturing feedback model can also be thought of in a mathematical way. Imagine the complex shape data as a set of features about the part \( \{X_1, \ldots, X_K\} \), such as the point data of the boundary surface:

\[
\text{Complex Shape Data} : SD = \{X_1, \ldots, X_K\}.
\]  \hspace{1cm} (4.1)

The deformable model is a function which takes this data and reduces it to a set of deformation parameters \( \{\phi_1, \ldots, \phi_N\} \) where \( N \ll K \). This is a quite important phenomenon because the complex shape data has been filtered to leave only the essential pieces of information:

\[
\text{Deformable Model} : DM (SD) = \{\phi_1, \ldots, \phi_N\}.
\]  \hspace{1cm} (4.2)

Afterwards the inverse model and the shape error function take the deformation parameters and transform them into relevant feedback information about the manufactured part:

\[
\text{Inverse Model} : IM (DM (SD)) = \text{Estimated Process Parameter Levels};
\]

\[
\text{Error Function} : EF (DM (SD)) = \text{Shape Error}.
\]  \hspace{1cm} (4.3)
Thus, the shape manufacturing feedback model can be summarised by a function based model shown in Figure 4-2. Having now described the overall nature of the shape manufacturing feedback model, each component of the model will be described in detail.

4.3 Manufacturing shape variation model

This section discusses the Manufacturing Shape Variation Model (MSVM). The MSVM is the first component of the shape manufacturing feedback model and it decomposes the manufactured shape's boundary into a series of weighted major variations from the average manufactured shape. This model describes the shape variation of the manufacturing process using the PDM (deformable model) (Cootes et al., 1995) that was explained in Chapter 3.

It must be ensured that the shape variation measured from the PDM training set contains all of the relevant variations that the manufacturing process produces. In a mathematical sense, it must be ensured that the principal modes of variation span the entire set of possible variations of the manufacturing process. Otherwise, the PDM description is no longer accurate. If it is impossible to gather all the principal modes of variation, then the unrepresented space must have a valid reason for not being described.

The DoE approach realises an organised set of results from the process. This structured approach allows the output of the process to respond to all controllable factors. Using a factorial experimental design, which was explained in Section 3.5.1, each factor and its levels are combined with every other factor. This creates a response space with as much manufacturing output variation as possible. The resolution of the output response can be improved by increasing the number of levels for each factor. In addition, by increasing the number of replicates, the variation caused by uncontrollable variations within the manufacturing process can be contained within the principal variation components if they are significant enough. The PDM training set can be then assured of containing as much of the significant process variation possible. The MSVM is, therefore, a conjunction of the PDM and DoE approaches. This combination allows the manufacturing process response to be learnt implicitly without having to rely on determining the material models for the process.
The remainder of this section describes the initialisation and the implementation of the MSVM.

### 4.3.1 Setting up the manufacturing shape variation model

The MSVM is initialised in the following way, see Figure 4-3 for an overview. First a set of "training" manufactured shapes is created according to a DoE approach (Cunningham, 1999). The DoE takes into account the required variation that is necessary from the process parameters, the constraints that may apply to the process, and any other information that may be useful in setting up the experiment. The required variation in the process parameters depends on the particular manufacturing process and the type of shape variation in which one is interested. The set of manufactured shapes is created according to a randomised set of $N^K$ factorial experiments as specified from the DoE, where $N$ is the number of levels for each factor and $K$ is the number of factors.

After the set of manufactured shapes has been created, they are measured using some form of sensor to obtain the boundary profiles of each shape. The boundary profiles are then assigned boundary points. This assignment is situation dependent although it is automatic in each case. When assigning points it is essential that all of the shape variation is captured.

Once each shape has been assigned boundary points, a mean shape of the data set is calculated. Each shape is then aligned to the mean shape. Several alignment algorithms can be used to achieve registration, for example Besl and Mckay's (1992) ICP method or the method implemented by Cootes et al. (1995) in their PDM approach. Essentially the alignment process minimises the summed distances between the measured shapes.
and the average of the measured shapes by rotating and/or translating each measured shape. This is an iterative process that eventually converges to a minimal error and produces an aligned data set.

The PDM is then performed on the aligned set of measured boundary profiles. The PDM equation (3.12) from Chapter 3 is updated for the manufacturing process as follows:

\[
\hat{X}_{\text{manufactured shape}} = \hat{X}_{\text{mean manufactured shape}} + P\tilde{b},
\]

where \(\hat{X}_{\text{manufactured shape}}\) is the list of boundary point coordinates for the PDM approximation of any manufactured shape, and \(\hat{X}_{\text{mean manufactured shape}}\) is the list of boundary point coordinates for the mean shape of the data set. Applying principal component analysis to the covariance matrix of coordinate variation realises a restricted set of eigenvectors that describe most of the variations in the set of shapes. The \(P\) matrix is created by combining together the \(t\) most significant eigenvectors. The eigenvectors are the major modes of shape variation for the manufacturing process. From these modes the corresponding \(\tilde{b}\) vectors can be determined, which can deform the mean shape into every other shape in the set of training shapes. The MSVM has now been initialised.

### 4.3.2 Implementation of the manufacturing shape variation model

The MSVM is implemented by creating a function that determines the major variations in a manufactured shape for a particular process:

\[
\mathcal{F}[\text{Manufactured Shape}] = \text{Major Shape Variations},
\]

where \(\mathcal{F}[\cdot]\) is the MSVM function that takes a measured manufactured shape and returns the major shape variations. A newly manufactured shape, not in the original PDM training set of manufactured shapes, can therefore be described by a \(\tilde{b}\) vector as seen by rearranging equation (4.4):

\[
\tilde{b} = P^+ \left(\hat{X}_{\text{measured shape}} - \hat{X}_{\text{mean manufactured shape}}\right),
\]

where \(\hat{X}_{\text{mean manufactured shape}}\) is the mean shape defined above and the variable \(\hat{X}_{\text{measured shape}}\) contains the points on the surface of the newly produced part. The \(P\) matrix holds the eigenvectors of shape variation and \(P^+\) is a pseudo inverse of \(P\),

\[
P^+ = (P^TP)^{-1}P^T.
\]

The resulting \(\tilde{b}\) vector describes the newly manufactured shape in terms of the principal components of variation. This divides every manufactured shape into a small and distinct set of quantifiable variables. The major problem with this representation, as was raised earlier, is the situation where the \(P\) matrix does not hold all the major
variations from the manufacturing process. The PDM is an approximation of the actual shape and the PDM equation is more accurately written as,

\[ \hat{X}_{\text{actual shape}} = \hat{X}_{\text{mean manufactured shape}} + P\hat{b} + \hat{\varepsilon}, \]  

(4.8)

where \( \hat{X}_{\text{actual shape}} \) is the list of the measured manufactured shape's actual coordinates, and \( \hat{\varepsilon} \) is the approximation error between the PDM shape and the actual shape. If the PDM has a good training set that captures all the major variations, then \( \hat{\varepsilon} \) is insignificant. However, if the PDM training set does not include a major source of variation, then the error \( \hat{\varepsilon} \) can be large and the PDM description of shape is no longer accurate. It is for this reason that DoE is applied to the creation of the PDM training set.

### 4.4 Shape error function

The shape error function is the first form of feedback that is provided by the shape manufacturing feedback model. The shape error function relies on two initialisations. First, the MSVM must be initialised to obtain the major shape variation modes of each manufactured shape to be tested. Second, the desired shape that will be used as the “template” must first be converted into the same shape description format as the manufactured shapes that are to be analysed. The shape error function is then ready to be applied.

The remainder of this section discusses the advantages of this approach to shape error measurement with respect to the two chosen manufacturing systems, forging and sheet metal forming, described in Section 2.6.1. The determination of the shape error function using the PDM is then outlined in Section 4.4.2. Finally, the general equations used to create the shape error are defined in Section 4.4.3.

#### 4.4.1 Advantages of the shape error function

There are several advantages that the shape error function will provide. The first advantage of this method is that the basis set of variations used to describe shape error is not pre-determined and is, therefore, not forced onto the process or its output. This is because the proposed shape error function uses the major shape variations due to process variability to describe the shape error of the components. In addition, separating the error into the major modes of variation decreases the dimensionality of the problem like discrete Fourier transforms (see Section 2.6.1) and the differential geometry methods. The second advantage is that no smoothing of the formed part's surface data is necessary before analysis. The surfaces are allowed to deform in a reasonably non-uniform manner because the surface is independently modelled using the coordinates of the sample points on the surface. This type of shape measure is
also capable of being used in two dimensions or three dimensions, although this thesis will only consider the two dimensional case. The final advantage is that the shape error data appears to have reasonable separation between variations in the process parameters which will be used in a later section to develop an inverse shape model.

4.4.2 Devising the shape error function

Given the creation of the MSVM that can determine the major modes of shape variation from the manufacturing process, the question arises what can be done with these modes? One of the important contributions of this thesis is to use the variation modes created by PDM analysis (MSVM) to develop a shape error function. The shape error function will measure the shape differences between the desired shape and the manufactured part in terms of these modes. This function is calculated in the following way:

1. find the major modes of variation of the process through an initialised MSVM;
2. obtain the desired shape's $\vec{b}$ vector;
3. obtain the manufactured part's $\vec{b}$ vector;
4. compare the two $\vec{b}$ vectors.

Let us backtrack and determine what the $\vec{b}$ vectors mean with respect to the manufacturing process. Figure 4-4 shows how the boundary points on the surface of a blank are modified during the manufacturing phase and how the weighting vector $\vec{b}$ is calculated by using the MSVM based on the PDM. The manufacturing deforming process actually modifies the shape of the blank and this can be represented in a functional form as:

$$X_{\text{deformed}} = F_{\text{manf}} \left[ X_{\text{undeformed}}, \vec{U} \right] + \eta_{\text{manf}} + \eta_{\text{measure}},$$

(4.9)

where $X_{\text{undeformed}}$ is the list of points on the boundary of the pre-manufactured blank and $X_{\text{deformed}}$ is the list of points on the boundary of the formed part. $\vec{U}$ is a vector of the ideal process parameter values. The combination of noise and errors in the manufacturing system is represented by the boundary point vector $\eta_{\text{manf}}$ and the noise from measuring the part is represented by the boundary point vector $\eta_{\text{measure}}$. $F_{\text{manf}}$ is the ideal manufacturing deforming function which has two parameters: a list of boundary surface points, $X_{\text{undeformed}}$; and a vector of input process control parameters, $\vec{U}$. Therefore, the desired shape, $\hat{X}_{\text{desired}}$ can be represented as,

$$\hat{X}_{\text{desired}} = F_{\text{manf}} \left[ \hat{X}_{\text{undeformed}}, \hat{\vec{U}} \right].$$

(4.10)
process parameters (DoE) 
\[ \hat{\mathbf{X}}_{\text{undeformed}} \rightarrow \text{Manufacturing Process} \]

manufacturing error

measuring error

\[ \hat{\mathbf{X}}_{\text{desired}} \text{ related to } \hat{\mathbf{U}} \]

( \( \mathbf{b}, \hat{\mathbf{X}}_{\text{MSVM}} \) )

Manufacturing Shape Variation Model

Align Shape

\[ \mathbf{X}_{\text{Training set}} = \{ \hat{\mathbf{X}}_1, \hat{\mathbf{X}}_2, \ldots, \hat{\mathbf{X}}_N \} \text{ related to } \Delta \mathbf{U} \]

\[ \hat{\mathbf{X}}_{\text{deformed}} \]

Figure 4-4: Block diagram showing the progress of a set of boundary nodes on the part \( \{ \hat{\mathbf{X}}_{\text{undeformed}}, \hat{\mathbf{X}}_{\text{deformed}}, \hat{\mathbf{X}}_{\text{MSVM}} \} \) before and after the manufacturing process and the MSVM. The block diagram includes the noise that will be added throughout the inspection process. \( \hat{\mathbf{U}} \) is the average set of process parameters that supposedly create \( \hat{\mathbf{X}}_{\text{desired}} \). \( \Delta \mathbf{U} \) is the variations in the process parameters away from \( \hat{\mathbf{U}} \), this is determined by the DoE approach.

Now, consider the following MSVM applied to the deformed part,

\[ \hat{\mathbf{X}}_{\text{MSVM shape}} = \hat{\mathbf{X}}_{\text{mean}} + \mathbf{P} \mathbf{b}_{\text{manf}} = \hat{\mathbf{X}}_{\text{deformed}} + \hat{\mathbf{e}}, \quad (4.11) \]

where \( \hat{\mathbf{e}} \) is the difference between the MSVM shape and the deformed shape, noting that the MSVM shape approximation accuracy depends on how many eigenvectors are included in the \( \mathbf{P} \) matrix. It is also dependent on the MSVM holding all the major variations of the manufacturing process in the \( \mathbf{P} \) matrix. Therefore, \( \hat{\mathbf{e}} \) is small if we assume that the process has not changed significantly since the calculation of the \( \mathbf{P} \) matrix.

What does this mean in terms of the \( \mathbf{b} \) vector? When equation (4.11) is rearranged the \( \mathbf{b} \) vector is given by,

\[ \mathbf{b}_{\text{manf}} = \mathbf{P}^+ \left( \hat{\mathbf{X}}_{\text{deformed}} + \hat{\mathbf{e}} - \hat{\mathbf{X}}_{\text{mean}} \right). \quad (4.12) \]

Then substituting equation (4.9) into equation (4.12) gives,

\[ \mathbf{b}_{\text{manf}} = \mathbf{P}^+ \left( \mathcal{F}_{\text{manf}} \left[ \hat{\mathbf{X}}_{\text{undeformed}}, \hat{\mathbf{U}} \right] + \hat{\mathbf{n}}_{\text{manf}} + \hat{\mathbf{n}}_{\text{measure}} + \hat{\mathbf{e}} - \hat{\mathbf{X}}_{\text{mean}} \right). \quad (4.13) \]

The \( \mathbf{b} \) vector for the desired shape can be written as follows after revising equation (4.12),

\[ \mathbf{b}_{\text{desired}} = \mathbf{P}^+ \left( \hat{\mathbf{X}}_{\text{desired}} + \hat{\mathbf{e}}_{\text{desired}} - \hat{\mathbf{X}}_{\text{mean}} \right), \quad (4.14) \]
where $\delta_{\text{desired}}$ is the difference between the desired shape and its MSVM approximation, note this is not the shape error of manufactured shape.

The Shape error is found by taking the difference between the manufactured shape and the desired shape. This can be approximated by measuring the distance between the $\tilde{b}$ vector of the desired shape and the $\tilde{b}$ vector of the manufactured shape,

$$Shape\ Error = \mathcal{M} \left( \tilde{b}_{\text{manf}} - \tilde{b}_{\text{desired}} \right),$$

where $\mathcal{M} \left( \tilde{b}_{a-b} \right)$ is the measure function that returns a weighted form of distance between vectors $\tilde{b}_a$ and $\tilde{b}_b$, note that $\tilde{b}_{a-b} = \tilde{b}_a - \tilde{b}_b$. The measure function obeys the following rules:

$$\mathcal{M} \left( \tilde{b}_c \right) : (\mathbb{R}^t) \rightarrow \mathbb{R};$$

$$\mathcal{M} \left( \tilde{b}_c \right) \geq 0;$$

$$\mathcal{M} \left( \tilde{b}_c \right) = \mathcal{M} \left( -\tilde{b}_c \right);$$

and

$$\mathcal{M} \left( \tilde{b}_{\text{zero}} \right) = 0,$$

where $\tilde{b}_{\text{zero}} = [0, \ldots, 0]^T$ is the zero $\tilde{b}$ vector, and $(\mathbb{R}^t)$ is the general case of a $\tilde{b}$ vector, that is, a vector containing $t$ real valued components, one for each eigenvector of $P$.

Therefore, shape error can be written by substituting equations (4.12) and (4.14) into equation (4.15):

$$Shape\ Error = \mathcal{M} \left( P^+ \left( X_{\text{manf}} \left[ X_{\text{undeformed}}, \bar{U} \right] + \hat{\delta}_{\text{manf}} + \hat{\delta}_{\text{measure}} + \tilde{\delta} - \hat{X}_{\text{mean}} \right) \right) -$$

$$P^+ \left( \hat{X}_{\text{desired}} + \tilde{\delta}_{\text{desired}} - \hat{X}_{\text{mean}} \right)$$

$$= \mathcal{M} \left( P^+ \left( \hat{\delta}_{\text{manf}} + \hat{\delta}_{\text{measure}} + \tilde{\delta} - \tilde{\delta}_{\text{desired}} \right) \right),$$

If one assumes that the $P$ matrix contains most of the shape variations of the manufacturing process then $\delta$ and $\delta_{\text{desired}}$ are very small in magnitude. The second assumption made is that the measurement error, $\hat{\delta}_{\text{measure}}$, is again very small in magnitude. This assumption will become increasingly certain with the advancement in the accuracies of the automated measuring systems. Therefore, $( \|\delta\|, \|\delta_{\text{desired}}\|, \|\hat{\delta}_{\text{measure}}\| ) \ll \|\hat{\delta}_{\text{manf}}\|$, which means

$$Shape\ Error \cong \mathcal{M} \left( P^+ \hat{\delta}_{\text{manf}} \right).$$

The $P^+$ matrix converts the $\hat{\delta}_{\text{manf}}$ boundary point vector into weightings of the major modes of shape variation. This is important as the measure function can then weight the more significant modes over the less significant modes. This emphasises the errors that are significant to the manufacturing process.
This analysis assumes that the input process parameter combination is the same for both the manufactured part and the desired shape. In the case where the input parameter combination is different the shape error is as follows:

\[
\text{Shape Error} \triangleq \mathcal{M} \left( P^+ \left( \mathcal{F}_{\text{manf}} \left[ \hat{X}_{\text{undeformed}}, \hat{U}_i \right] - \hat{X}_{\text{desired}} + \hat{\eta}_{\text{manf}} \right) \right). \tag{4.18}
\]

This equation decomposes the error between the desired shape and the manufactured shape into the principal modes of variation.

### 4.4.3 Shape error equations

The \( \vec{b} \) vectors for the desired shape and the manufactured part are calculated using the following equations:

\[
\begin{align*}
\vec{b}_{\text{desired shape}} &= P^+ ( \hat{X}_{\text{desired shape}} - \hat{X}_{\text{mean}} ) \\
\vec{b}_{\text{manufactured part}} &= P^+ ( \hat{X}_{\text{manufactured part}} - \hat{X}_{\text{mean}} ),
\end{align*}
\tag{4.19}
\]

where \( \hat{X}_{\text{mean}} \) is the mean shape of the training set. The vector \( \hat{X}_{\text{desired shape}} \) contains the points on the boundary of the desired part shape and the vector \( \hat{X}_{\text{manufactured part}} \) contains the points on the boundary of the manufactured part. Note that the desired shape can be created in CAD so long as the positioning of boundary points is consistent with the training set of shapes.

The shape error is then a weighted sum of the absolute differences between the manufactured part's \( \vec{b} \) vector and the desired shape's \( \vec{b} \) vector. A weighting is applied to bias the shape error in favour of whatever aim the shape error function is interested in investigating.

\[
\text{Shape Error} = \alpha_1 \left| \vec{b}(1)_{\text{manufactured part}} - \vec{b}(1)_{\text{desired shape}} \right| + \alpha_2 \left| \vec{b}(2)_{\text{manufactured part}} - \vec{b}(2)_{\text{desired shape}} \right| + \ldots \tag{4.20}
\]

where \( \alpha_i \) is the weighting factor for the \( i^{th} \) mode.

For this thesis the weighting factor will be equal to the significance each mode contributes to the overall variation. This is equivalent to dividing the eigenvalue of the particular mode by the sum of all the eigenvalues. The eigenvalue is directly proportional to the amount of variance each mode contributes to the overall variance of the training set. The reason for this weighting is because a unit difference in the first mode causes more significant changes with regard to the process than a unit change in the tenth mode. The eigenvalue weighting is applied to reflect the significance of a unit change in a component. This becomes important if the statistically significant modes do not exhibit large changes in shape. In this situation, the less significant high-order
modes may have equivalent \( \vec{b} \) vector component values as the low-order modes, and therefore the high-order modes need to be scaled down accordingly.

\[
\text{Shape Error} = \left( \frac{\lambda_1}{\sum_{i=1}^{N} \lambda_i} \right) |\vec{b}(1)_{\text{manufactured part}} - \vec{b}(1)_{\text{desired shape}}| + \left( \frac{\lambda_2}{\sum_{i=1}^{N} \lambda_i} \right) |\vec{b}(2)_{\text{manufactured part}} - \vec{b}(2)_{\text{desired shape}}| + \ldots
\]

(4.21)

where \( \lambda_i \) is the \( i^{th} \) eigenvalue of the covariance matrix \( S \) defined in Section 3.2.2.

There are other possible weighting scenarios. For example, if one is interested in shape error in which outlying errors are highlighted, then the scaling of each mode can be set equal to the inverse of the standard deviation \( (\sigma_i) \) of each \( \vec{b} \) vector component in the PDM training set \( (\alpha_i = \frac{1}{\sigma_i}) \).

4.5 Inverse shape variation–process parameter model

The inverse shape variation process parameter model is the second form of feedback associated with the shape manufacturing feedback model. The inverse model, like the MSVM, has to be initialised to learn the relationship between the shape variation and the process parameters of the manufacturing process. After the inverse model has been initialised, it will return the estimated process parameters when given a manufactured shape from the learnt process as an input.

This section first outlines the advantages for the shape variation–process parameter inverse model with respect to the forging and sheet metal forming processes described in Section 2.6.2. The theory behind the inverse model is discussed in Section 4.5.2. Finally, the initialisation and the implementation of the inverse model are presented in Sections 4.5.3 and 4.5.4.

4.5.1 Advantages of the inverse model

Most of the literature on inverse models within forging and sheet metal forming is related to die design. In the case of die design, the prime engineering problem is to design the die’s profile to produce the desired part. This type of inverse modelling uses shape optimisation to modify the shape of a die to obtain the appropriate output shape. A more complex version of this problem is designing the preform die to provide the appropriate intermediate shape that can be forged or formed into the final shape. Fourment (1996) and Zhao (1997) both developed examples of inverse models in forging to find the appropriate preform profile to assist creating the final shape. Webb and Hardt (1991), and Karafillis and Boyce (1996; 1992) developed die designs in an attempt to predict the effects of springback to create the appropriate final shape for sheet metal forming.
The second type of inverse model determines the process parameter levels that are necessary to cause a particular process response or part. Gelin and Ghouati (1995; 1996; 1998; 2001) used shape optimisation on a finite element model to find the parameter levels that produced an actual sheet metal formed part.

This thesis will extend this second approach to inverse modelling and use the output shape variation from either a desired or an average shape to predict what are the process parameter levels that created the manufactured shape. From an automatic inspection perspective, we are interested in finding out what might have caused errors in the manufactured part and in asking how we can tune the process parameters to solve this error. The essential difference to other approaches is that this work is not optimising the parameters to find what is the optimal set-up to give the desired outcome, but rather this thesis is developing a model that returns the most likely set-up that created the manufactured output.

The other main difference between many of the previous approaches and our approach is that we assume that the material model is unavailable. An inductive experimental approach is conducted to implicitly learn the material model by developing relationships between the shape variation of the process and the process parameters. This is similar to the approach of Webb and Hardt (1991).

This thesis also follows on from work on inspection completed within the Masters thesis of Brett Daniel (1998). Daniel briefly examined geometric variations in forging to determine if a part was out-of-tolerance. His work used the Point Distribution Model (PDM) approach to elicit the major modes of shape variation from a simple forged part. There was, however, no further analysis of the modes. We will extend Daniel’s investigation by relating the information from the deformable shape model back to the set-up of the manufacturing process.

4.5.2 Manufacturing inverse model – theory

This section examines the development of a manufacturing inverse model given an initialised MSVM. First, we shall discuss the structure of the inverse model. The MSVM provides a \( \bar{b} \) vector from a measured manufactured part to the inverse model and the inverse model should then return the corresponding process parameters, \( \bar{U}^* \), that created that part. This defined by the following equation,

\[
\mathcal{G} [\bar{b}] = \bar{U}^*, \tag{4.22}
\]

where \( \mathcal{G} [\cdot] \) is the inverse model.

Now rearranging equation (4.9) on page 68 gives,

\[
\bar{U}^* = \mathcal{F}^{-1}_{manf} \left[ \hat{X}_{\text{deformed}} - \hat{\eta}_{\text{manf}} - \hat{\eta}_{\text{measure}} \right], \tag{4.23}
\]

where \( \mathcal{F}^{-1}_{manf} \) is the inverse manufacturing process function. Substituting equation (4.11)
for $\dot{X}_{\text{deformed}}$ gives,

$$\ddot{U}^* = F_{\text{manf}}^{-1} \left[ \ddot{X}_{\text{mean}} + P \ddot{b} - \ddot{\epsilon} - \ddot{\eta}_{\text{manf}} - \ddot{\eta}_{\text{measure}} \right]. \quad (4.24)$$

Therefore,

$$G \left[ \ddot{b} \right] \equiv F_{\text{manf}}^{-1} \left[ b, \ddot{X}_{\text{mean}}, P, \{\dot{\epsilon}, \ddot{\eta}_{\text{manf}}, \ddot{\eta}_{\text{measure}}\} \right], \quad (4.25)$$

where $\ddot{b}, \ddot{X}_{\text{mean}}, P$ are the observable parameters and $\{\dot{\epsilon}, \ddot{\eta}_{\text{manf}}, \ddot{\eta}_{\text{measure}}\}$ are the unknown parameters of the inverse model.

Thus, the inverse model, $G \left[ \cdot \right]$, depends on the inverse manufacturing process function, $F_{\text{manf}}^{-1}$. Most manufacturing process functions are very complex and non-linear as will be learnt in later chapters. Some underlying material models are even difficult to model forwards, let alone finding the inverse, see Section 7.2 on the simulation of spring-back. The inverse model can, therefore, be extremely difficult to determine analytically from first principles using the underlying theory. In addition, in some cases the inverse model may not be a function as a single part may have been created by several possible process parameter combinations. Nevertheless, some form of relationship can be found between the shape variation and the process parameters, particularly as there are many non-linear techniques available. This relationship will be only an indicative model of the process because it is not explicitly based on the underlying material models, but in the case where these are not available this method does provide a solution.

The relationship between the shape variation and the process parameters is developed using the PDM training set. This relationship can take two forms, discrete or continuous. The discrete case uses classification techniques to assign $b$ vectors to a certain process parameter combination, $\ddot{U}_j$. The continuous case applies data fitting techniques to create a function that maps a $b$ vector to a set of real valued process parameter levels.

Therefore, the inputs to the manufacturing process are the factorial set of process parameter combinations in the PDM training set,

$$\text{Set of process parameter combinations} : U = \left\{ \ddot{U}_1, \ddot{U}_2, \ldots, \ddot{U}_{(NK)} \right\}, \quad (4.26)$$

where $\ddot{U}_i = \{x_1, x_2, \ldots, x_K\}$ is a combination of process parameter levels and $x_i$ is the $i^{th}$ process parameter set at a specified level between $\{1, \ldots, N\}$.

The output of the manufacturing process is the PDM training set which is converted into a set of $b$ vectors,

$$\text{PDM training set} : B = \left\{ \ddot{b}_{\ddot{U}_1}, \ddot{b}_{\ddot{U}_2}, \ldots, \ddot{b}_{\ddot{U}_{(NK)}} \right\}, \quad (4.27)$$

where the response $\ddot{b}_{\ddot{U}_i}$ is as follows,

$$\ddot{b}_{\ddot{U}_i} = P^+ \left( F_{\text{manf}} \left[ \ddot{X}_{\text{undeformed}}, \ddot{U}_i \right] + \ddot{\eta}_{\text{manf}} + \ddot{\eta}_{\text{measure}} + \ddot{\epsilon} - \ddot{X}_{\text{mean}} \right).$$
Chapter 4. The Shape Manufacturing Feedback Model

The inverse model can then be defined as follows:

\[ \varepsilon \left[ \bar{b} \right]: \ (\mathbb{R}^d) \rightarrow \bar{U}_j \in U, \quad \text{discrete case;} \quad (4.28) \]

otherwise, \( (\mathbb{R}^d) \rightarrow (\mathbb{R}^K) \), continuous case,

where \((\mathbb{R}^d)\) is a general \(\bar{b}\) vector, \(\bar{U}_j\) is a particular process parameter combination, and \((\mathbb{R}^K)\) is a vector of the \(K\) real valued process parameters. The learning of the relationship between the input and output sets is discussed in the next section.

It must be noted that the inverse model, \(\varepsilon[\cdot]\), is also dependent on the noise of the manufacturing process and the measurement system, and also the accuracy of the PDM description. The accuracy of the PDM description has already been discussed in Section 4.3. The advantage of this method with regard to other algorithms with regard to the noise in the manufacturing process is that it implicitly learns the structure of the noise distribution when learning the relationship between the shape variations and the process parameters. Each process has its own systematic noise which cannot be modelled using first principles unless an extensive investigation is undertaken of the process. This thesis' inductive learning approach builds up a relationship that will take the nuances of each process into account.

4.5.3 Learning the relationship between shape variation and process parameters

This section describes the general method performed to learn the inverse shape variation–process parameter model. There are three major steps to the development of the inverse shape variation–process parameter model. These steps are shown in Figure 4-5. First, a DoE method is used to determine what process parameter variation is necessary to achieve the most shape variation out of the manufacturing process. The design of the experiment is also dependent on the \(a \ priori\) knowledge that the users may have about the manufacturing process. The DoE provides two outcomes: a set of manufactured parts with shape variation caused by varying the process parameters; and a set of corresponding lists of process parameter levels associated with each manufactured part.

The second step is the analysis of the set of manufactured parts by the MSVM which returns the major shape variations of each part in vector form. The final step is to relate the list of process parameter levels with the vectors of the shape variation modes \(\bar{b}\) vectors). This step is achieved using the pattern recognition and data fitting techniques described in the previous chapter (Chapter 3). There are two ways of analysing the shape variation data. First, the data can be classified in a discrete fashion, given the process parameter combinations are a finite set of input combinations. Second, a function can be identified to link the set of PDM \(\bar{b}\) vectors, \(B\), to the set of process parameter combinations, \(U\). This can only be achieved if there is a one-to-one relationship between the shape variations and the process parameter combinations. In both
Figure 4-5: Diagram showing the steps involved in setting up the inverse shape variation–process parameter model.

cases, supervised learning is used because we already have a labelled set of data from which to investigate, that is, the PDM training set.

Essentially this method creates a multi-dimensional “shape variation” space where the number of dimensions is given by the number of eigenvectors included in the P matrix of the PDM contained within the MSVM. The \( \vec{b} \) vectors may then be segmented into hyper-regions corresponding to the combinations of the process parameter levels. Each combination of the process parameters, \( \vec{U}_i \), corresponds to a hyper-region of \( \vec{b} \) vector points in the “shape variation” space. If the manufacturing function (\( F_{manf} \)) is well behaved in a functional identification sense, then each process parameter combination will produce a clustered region of \( \vec{b} \) points in the “shape variation” space. This situation will be seen in Sections 5.7 and 6.6.6 for forging and sheet metal forming processes, respectively. The analysis of the final geometry assumes that varying the input parameters of the process produces distinct and different end geometries. Otherwise, the segmentation of the data is intractable without some further knowledge about the sheet metal forming process. The spread of each region depends on the variability of the manufacturing process (process noise). The variability of each combination can be found by measuring the spread of the corresponding \( \vec{b} \) vector hyper-region in shape space. The variation associated with each combination of the process parameters will
Chapter 4. The Shape Manufacturing Feedback Model

Figure 4-6: An example of "shape variation" ($\vec{b}$ vector) space segmented into regions of similar process parameter clusters as well as indicating the banding of the High Parameter 2 region into more precise levels.

be analysed in Section 6.6.5 for a sheet metal forming process.

Figure 4-6 shows an idealised plot with clusters of data points where only two shape variation modes, $\vec{b}(1)$ and $\vec{b}(2)$, and two process parameters with two levels each are considered. In reality there are many dimensions, however the regions of $\vec{b}$ points are easier to see in two dimensions. The search for clusters can be refined depending on the number of levels used for each process parameter. Figure 4-6 shows that the second process parameter can be refined further by dividing the high cluster into high, medium and low. Clusters of data may have a large amount of overlap, particularly if the noise in the process is large or if certain parameters have little influence over the end shape.

The final aim of the development of the inverse shape variation-process parameter model is to create a model that will identify the process parameter combinations of a newly manufactured shape by determining the region in which the shape's weighting $\vec{b}$ vector falls.

4.5.4 Implementation of the inverse model

Once the inverse model has been determined, the shape manufacturing feedback model can then be applied to real data. The newly produced part is measured and the boundary data entered into the MSVM. The resulting $\vec{b}$ vector is then analysed by the inverse model. This model takes the $\vec{b}$ vector and returns the corresponding process parameter levels. These process parameters can then be compared to the desired process parameter levels. Note that the desired process parameter levels should produce the desired part. Feedback control measures can then be introduced to bring the process back to the appropriate quality level. The feedback mechanism may depend on the complexity relationship between process parameters and output shape. A simple feedback system
Chapter 4. The Shape Manufacturing Feedback Model

Figure 4-7: Diagram showing the implementation steps of the Shape Manufacturing Feedback Model.

Figure 4-8: This diagram shows the set-up of a simple shape manufacturing feedback system. This control system is not implemented within this thesis. The determination of the MSVM and the Inverse Model, however, is investigated in the following chapters.

is shown in Figure 4-8 that could be applied to the manufacturing system. This is not explored in this thesis.

4.6 Summary

This chapter has developed the theory behind the shape manufacturing feedback model as well as its components, the shape error function and the inverse shape variation–process parameter model.

The shape manufacturing feedback model is comprised of two main sections. The first section uses the Point Distribution Model (PDM) (Cootes et al., 1995) to obtain the major modes of shape variation in the output of the manufacturing process. The second section of the model uses classification and identification methods to develop an inverse model to relate the shape variation of the output with the variation of the process parameters of the manufacturing process. The model itself has two forms of feedback to the manufacturing process. First, the shape error of the part is returned. Second, the inverse model returns the most likely process parameter levels that can
create the inspected part.

The remainder of this thesis investigates applying the shape manufacturing feedback model to simulated forging process, experimental sheet metal forming processes, and simulated sheet metal forming processes.
Chapter 5

Forging processes

Shape Odyssey 2001 continues

5.1 Introduction

This chapter applies the shape manufacturing feedback model, as discussed in the previous chapter, to several simulated forging processes. Forging is a bulk deformation process where the major dimensions of a billet are modified by forming two dies together. There are three broad types of forging: open, impression, and closed die. The open die forging allows free deformation of some areas of the billet. The deformation in impression die forging is more constrained, and the shape is fully constrained in closed
die forging.

There are only a small number of papers that directly investigate geometric shape error in forging. Perhaps this is because tolerances of forged parts may not have to be precise so that other aspects of forging have received more investigation, such as material properties. As explained previously, Wright and Bourne (1988) developed an inspection system for an open-die forged part. Fujikawa and Ishii (1995) developed a diagnostic expert system that identifies the cause of various manufacturing defects and suggests certain solutions. They have investigated the "downstream" problem of what action can be taken by the operators of the process given a particular problem. There are also a number of papers reviewed in the Chapter 2 that have investigated finding an inverse model to forging (Fourment & Chenot, 1996; Zhao et al., 1997) that have developed simple shape error functions. Daniel et al. (1997) briefly examined geometric variations in forging to determine if a part was out-of-tolerance. This work used the Point Distribution deformable Model (PDM) (Cootes et al., 1995) approach to elicit the major modes of shape variation from a simple forged part. There was, however, no further analysis of the modes. This chapter's contribution is to extend Daniel's investigation in two major ways: first, by proposing a shape error function; and second, by offering a method that creates an inverse model between the shape variation of forged parts and the forging process parameters. These feedback models are developed according to the procedures outlined in Sections 4.4 and 4.5 in the previous chapter.

This chapter is structured as follows. Section 5.2 discusses the details of the forging process, which is simulated using finite element analysis (FEA) software specific to forging. Section 5.3 discusses how the forged shapes are aligned before being analysed. The shape variation modes, and shape variation trends are then analysed for the simple and "M" shaped die pair cases in Section 5.4. The results from varying the number of boundary points are also discussed in this section. The feedback models are then initialised in Section 5.5. Section 5.6 analyses the results from applying the shape error function to the two data sets. Finally, Sections 5.7 and 5.8 investigate the results of determining two types of inverse model from the two data sets. The first type of inverse model is the discrete model and the second type is the continuous inverse model.

5.2 Description of the forging process

This section describes the forging process, shows the geometries of the dies used, and defines the operating levels of the process parameters. Two types of die pairs are used: a simple open situation where the billet is flattened between two flat dies; and a more complex impression-die situation where the billet is forged into an "M" shape. The set-up of the simple die pair is seen in Figure 5-1 and the set-up of the "M" shaped die pair is shown in Figure 5-2.
Simple forging die pair

Set-up Parameters:
- Friction
- Velocity
- Travel

Figure 5-1: The simple open die pair configuration. This set-up will be defined as the simple die pair or just simple on graphs or plots.

"M" Shape forging die pair

Set-up parameters:
- Friction
- Offset
- Travel
- Velocity

Figure 5-2: The "M" shape impression-die pair configuration. This set-up will be defined as the "M" shaped die pair or just "M" shape on graphs or plots.
Table 5.1: Table of the simple die pair process parameter levels. There are 11 levels for each parameter and 1331 combinations in total.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Friction</th>
<th>Velocity</th>
<th>Travel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Value</td>
<td>( \mu = 0.10 )</td>
<td>5.0mm/sec</td>
<td>10.0</td>
</tr>
<tr>
<td>Step Size</td>
<td>0.02</td>
<td>1.0mm/sec</td>
<td>2.0mm</td>
</tr>
<tr>
<td>Maximum Value</td>
<td>( \mu = 0.30 )</td>
<td>15.0mm/sec</td>
<td>20.0mm</td>
</tr>
</tbody>
</table>

Table 5.2: Table of the “M” shaped die pair process parameter levels. There are 5 levels for each parameter and 625 combinations in total.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Friction</th>
<th>Velocity</th>
<th>Travel</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Value</td>
<td>( \mu = 0.10 )</td>
<td>5.0mm/sec</td>
<td>31.0mm</td>
<td>-10.0mm</td>
</tr>
<tr>
<td>Step Size</td>
<td>0.05</td>
<td>2.5mm/sec</td>
<td>2.0mm</td>
<td>5.0mm</td>
</tr>
<tr>
<td>Maximum Value</td>
<td>( \mu = 0.30 )</td>
<td>15.0mm/sec</td>
<td>39.0mm</td>
<td>10.0mm</td>
</tr>
</tbody>
</table>

The finite element analysis (FEA) package Forge 2.7\textsuperscript{1} is used to simulate the forging process. A medium carbon steel (0.45\% C) is selected for the billet material because of its high usage in hot working. Forge 2.7 uses the Norton-Hoff material behaviour law that relates the deviatoric stress tensor, \( \sigma \), to the strain rate tensor, \( \dot{\varepsilon} \), as follows (Chenot & Bay, 1998):

\[
\sigma = 2K(T, \dot{\varepsilon})(\sqrt{3}\dot{\varepsilon})^{m-1}\dot{\varepsilon},
\]

(5.1)

where \( T \) is the temperature in degrees Kelvin, \( K(T, \dot{\varepsilon}) \) is the strength equation, \( m \) is the strain rate sensitivity, \( \dot{\varepsilon} \) is the effective strain:

\[
\dot{\varepsilon} = \left( \frac{2}{3} \sum_{i,j} \varepsilon_{ij} \right)^{1/2},
\]

\( \varepsilon \) is the effective strain rate and \( \dot{\varepsilon} \) is the strain rate tensor. Forge 2.7 recommends in its examples that the exponential thermal law be chosen to approximate the strength equation \( K(T, \dot{\varepsilon}) \) for hot steel working and is given by,

\[
K(T, \dot{\varepsilon}) = K_0(\dot{\varepsilon} + \tilde{\varepsilon}_0)^n \exp(-\beta T),
\]

(5.2)

where \( K_0 \) is a constant, \( \tilde{\varepsilon}_0 \) is the strain-hardening regulating term, \( n \) is the sensitivity to strain-hardening coefficient and \( \beta \) is the temperature term.

There are numerous process parameters within forging that can be modified. This chapter only investigates three process parameters in the simple die pair case and four process parameters in the “M” shaped die pair case. The process parameters chosen did not include the material properties of the billet as it is assumed that the billet’s properties remain constant across all the tests.

The three process parameters chosen for the simple case are: friction between the dies and the billet; velocity of the upper die; and travel of the upper die from initial to final position. These process parameters are chosen because they have a significant

\textsuperscript{1}Forge 2.7 is a forging simulation package produced by Transvalor S.A.
effect on the end part shape and/or their process parameter levels are controllable by the operator.

These same process parameters are also chosen for the “M” shape case. In addition, the fourth parameter is horizontal offset to the billet, this accounts for the misplacement and misalignment of the billet. Although this process parameter is not an issue in the simple die pair situation, it has a significant effect on the end “M” shape. The simple die pair case does not have any alignment problems, assuming the upper and lower dies remain level, because if the billet is placed off-centre the same end shape should result.

The process parameters of the forging model are varied according to an eleven-level, three-factor factorial experiment for the simple die pair case and a five-level, four-factor factorial experiment for the “M” shape die pair case. The ranges of the two data sets can be seen in Tables 5.1 and 5.2. These two sets of forged billets formed the basis for the two MSVMs used in this chapter. The creation of the MSVMs is performed according to the method outlined in Section 4.3.

Once the simulations are finished, 10% of the end forged shapes are randomly chosen to see if the simulations have processed to completion. Completion, in this case, means that the simulation’s punch achieves the required travel value. Occasionally, the simulation fails at an intermediate step because the internal stresses and strains cannot be resolved. The simulations with extreme (minimum and maximum) parameter values are also checked.

5.3 Billet alignment and point placement

This section describes how each set of shapes is made consistent. In addition, the number of points needed to describe the boundary cross section is examined. Finally, the way the points are distributed along the boundary is also investigated.

Each shape in a set needs to be made consistent with the other shapes in the set in two ways. First, the shapes have to be aligned to remove registration errors from being detected as shape errors. This error is removed by carefully aligning all the billets before the simulations are performed. The computer simulated forged part then remains fixed to the original alignment when the simulation ends. This ensures each forged shape has the same alignment.

Secondly, the number of points which describe the cross section or boundary surface of each forged shape needs to be made consistent so that they can be compared. Consistency, in this case, is where a constant number of points is used to describe the boundary of each shape in the training set. This problem occurs because of the automatic meshing function in Forge 2.7 which affects the number of boundary nodes that explain the surface of the forged part. Obtaining a constant number of points on each shape involves either increasing the number of points by creating redundant points, or reducing the number of points by removing irrelevant points where possible. This
operation can decrease some of the available shape variation information if a very small number of points is used to describe the boundary. Information is lost because there are not enough boundary points to measure localised variations on the part's surface. The final number of points used, therefore, must be chosen carefully. This author decided to introduce redundant points to get a consistent number of points across the training set and not remove any shape information. The original numbers of boundary points range from 90 to 150 points for the simple die pair case and 140 to 180 boundary points for the "M" shaped die pair case. The two final sets of forging data were aligned using 200 boundary points to provide reasonable accuracy without decreasing any information from the original shapes when analysing the variations in geometry. These two sets are used throughout this chapter except for the section investigating varying the number of boundary points. The affect of reducing the shape information by reducing the number of boundary points is seen in Section 5.4.3. Both the simple and "M" shape die pair sets are varied in the following series of boundary points \(\{200, 150, 100, 20, 6\}\). These number of points are arbitrarily chosen to obtain an idea of how the separation of the signatures changes, particularly at low numbers of boundary points.

Another important consideration for the two data sets is the point distribution around the boundary cross section of the forged shapes. Cootes et al. (1995) suggest that the choice of points falls into three categories:

Type 1: points with a particular application or functional significance;

Type 2: points that are application independent or image features such
as curvature extrema;

Type 3: points that are interpolated between points of type 1 and 2.

One of the motivations for this research work was to develop an automatic inspection tool. Therefore, the points chosen on the boundary cannot be selected manually. This restricts the types of points that can used to construct the set of boundary points to type 2 and type 3. Type 2 points can be found automatically through image processing techniques and type 3 points can be interpolated using computer algorithms without manual input. The selected forged shape features must be observable in each shape of the set and must also be in a consistent section of the boundary of each shape. The interpolating method must make sure that the intervening selected boundary points are comparable across the set of forged shapes. Comparable means that the actual deformation in the shape is measured, and not the changes due to the interpolation between feature points.

The point distribution method that is chosen for the simulated forged parts distributes points evenly around each part's boundary from a consistent reference point. The reference point selected is the lower floor mid-point of the billet which is aligned to the origin before each simulation, see Figure 5-3. Given the symmetry of the simple case process, the mid-point of the deformed billet is the same as the original billet. Note, however, that this method does not allow the user to arbitrarily assign points to regions of interest. The "M" shape case does not have the advantage of symmetry because the billet can be offset, but it has more recognisable features. This is because the offset billets have their point distributions dominated by the skew of the billet to one side or the other. The lack of symmetry means that the evenly distributed points around the boundary perimeter is no longer effective. Therefore, the point distribution method is modified such that the total number of points is halved; each half is then distributed evenly on the left and right sides of the "M" shaped billet, see Figure 5-3.

In summary, the number of boundary nodes of each shape is made consistent by the following process:

- determine a constant reference point across every shape that is easily recognised;
- distribute the points around the shape.

**Simple die pair:** a fixed number of points are evenly spread around the perimeter of the cross section for the simple die pair case.

**"M" shape die pair:** a fixed number of points is split in two and evenly spread around the perimeter of each half (right and left).

The reference point used as the start point for each list of boundary nodes can be seen in Figure 5-3.
5.4 Shape variation of forging

The aligned forged parts are used to create the two MSVMs (simple and "M" shaped die pair cases) using the method described in Section 4.3. The results for this chapter are based on the \( \vec{b} \) vectors obtained from the simple and "M" shaped die pair MSVMs. Each forged shape in the two data sets has an associated \( \vec{b} \) vector. This creates two sets of \( \vec{b} \) vectors that are analysed by the feedback models.

5.4.1 Major modes of shape variation

The MSVM relies on the PDM finding the major modes of deformation from statistical data obtained from the training data set. The covariance matrix, \( S \), is calculated for both sets of data. In the case of the simple die pair simulations, the first eigenvector of the matrix \( S \) held 99.5% of the shape variant information. This implies that the first mode of variation explains practically all of the shape variation of the simple die pair data set. Note, however, that some vital separation information for classification can be realised from the remaining 0.5% of the shape variant information. Similarly, the first two eigenvectors held 97.55% of the shape variant information for the "M" shaped die pair simulations.

The variation modes can be displayed visually as changes in shape from the average forged part. An over emphasised first mode of variation for the simple die pair data set is seen in Figure 5-4(a). This example indicates that the travel parameter dominates the first mode of variation because the first mode represents the distance that punch travels to squeeze the deformed billet. Similarly, the first two variational modes of the "M" shaped die pair data are shown in Figures 5-4(b) and 5-4(c). The most significant mode appears to be strongly influenced by the offset of the billet, in addition the second mode appears to be influenced by the travel of the punch.

One of the goals of applying the MSVM to the data is to reduce the shape variation of the process down to a minimum number of quantifiable modes. It is, therefore, beneficial to choose a minimum number of eigenvectors to create the \( P \) matrix. The number of modes used to explain the shape variation also depends on the separability of the data. The between class variation is important for discrete and continuous inverse models. The differences between the process parameter box plots, particularly changes in the positions of the square points (quartiles), imply that there is separability between the data subsets.

Figures 5-5, 5-6 and 5-7 are the box plots for the simple die pair case for each of the chosen process parameters: velocity, friction, and travel. For this data set the least separable parameter is velocity and yet it appears that the third mode or component of the \( \vec{b} \) vector gives reasonable separation when comparing the box plots for velocity of the punch at 5mm/sec and 15mm/sec, see Figure 5-5. Note that the third mode is the third \( \vec{b} \) vector index and its box plot result is seen in the column above the value "3"
Chapter 5. Forging processes

Simple die pair, first mode perturbed

(a) Weighting factor = 0.9951

"M" shaped die pair, first mode perturbed

(b) Weighting factor = 0.6076

"M" shaped die pair, second mode perturbed

(c) Weighting factor = 0.3678

Figure 5-4: Diagrams showing the first few grossly emphasised modes for the Simple and "M" shaped die pair data sets. The solid line indicates the average shape of the training set. The small dotted line shows the shape created by positively perturbing only the first mode of variation. The dashed line shows the shape created by negatively perturbing the first mode of variation. The weighting factor is the eigenvalue of the particular mode divided by the sum of all the eigenvalues of the data set.
Figure 5-5: Velocity process parameter box plots for the third to eleventh $b$ indices, simple die pair case. Box plots (a) and (b) show velocity = 5.0 mm/sec and 15.0 mm/sec respectively. The asterisk points are the extreme data values that occur for that particular data subset. The square points are the 25th and 75th quartiles calculated from the data subset, that is, 50% of the data subset is contained between the two square points. The solid line between the two square points is the median value and the dotted line is the zero point that indicates no variation from the average shape.

Figure 5-6: Friction process parameter box plots for the third to twelfth $b$ indices, simple die pair case. Box plots (a) and (b) show friction of $\mu = 0.10$ and $\mu = 0.30$ respectively. See Figure 5-5 for details on box plot notation.
on the $\vec{b}$ vector indices axis. It may also be possible to gain some extra classification information from some of the components up to the eleventh mode. Therefore, the box plots suggest that at most 11 modes are necessary for classification and the third mode holds much of the discriminant information when classifying velocity. The box plots for the friction parameter, simple die pair case, indicate that the third, fourth and fifth modes are influenced by friction, see Figure 5-6. There does not appear to be any further discriminant information above the twelfth mode. The box plots for the simple die pair travel parameter imply that the first mode is strongly influenced by the travel of the punch, see Figure 5-7. The other modes do not appear to have much correlation with the travel parameter. It should be noted that the difference between the minimum and maximum extreme values of the first mode of variation for the high travel parameter appears to be sizably larger than that of the low travel parameter. The reason for this difference between the box plots is explained in Section 5.6. The results of the analysis on the three simple die pair parameters are summarised in Table 5.3.

**Table 5.3:** This table lists the main mode influenced by each process parameter for the simple die pair case. The range of modes affected by each process parameter is also listed.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Main component(s) influenced</th>
<th>Range of variational modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>3</td>
<td>1 – 11</td>
</tr>
<tr>
<td>Friction</td>
<td>3,4,5</td>
<td>1 – 12</td>
</tr>
<tr>
<td>Travel</td>
<td>1</td>
<td>1 – 2</td>
</tr>
</tbody>
</table>
Figure 5-8: Velocity process parameter box plots for the first to twelfth \( \tilde{b} \) indices, "M" shape die pair case. Box plots (a) and (b) show velocity = 5.0 mm/sec and 15.0 mm/sec respectively. See Figure 5-5 for details on box plot notation.

Figure 5-9: Friction process parameter box plots for the third to eleventh \( \tilde{b} \) indices, "M" shape die pair case. Box plots (a) and (b) show friction of \( \mu = 0.10 \) and \( \mu = 0.30 \) respectively. See Figure 5-5 for details on box plot notation.
Figure 5-10: Travel process parameter box plots for the first to eighth $b$ indices, “M” shape die pair case. Box plots (a) and (b) show travel = 31.0 mm and 39.0 mm respectively. See Figure 5-5 for details on box plot notation.

Figure 5-11: Offset process parameter box plots for the first to eighth $b$ indices, “M” shape die pair case. Box plots (a) and (b) show offset = −10.0 mm and 10.00 mm respectively. See Figure 5-5 for details on box plot notation.
Figures 5-8, 5-9, 5-10 and 5-11 are the box plots for the “M” shaped die pair case for each of the chosen parameters: velocity, friction, travel, and offset. The dominant parameter for the “M” shaped die pair simulations is the offset parameter. This parameter heavily influences the first mode as can be seen in Figure 5-11. The modes above the third component do not appear to be affected by the offset parameter. The travel parameter appears to influence only the second mode but not affect the higher modes, see Figure 5-10. Similar to the simple die pair case, the “M” shaped die pair travel parameter has a greater range of its extreme values in the second mode for high travel parameter values than for low travel parameter values. This is also discussed in Section 5.6. The friction parameter appears to marginally influence the fourth mode and the eleventh mode, see Figure 5-9. The velocity parameter did not appear to influence any mode, see Figure 5-8. This will make the levels of this parameter very hard to distinguish from within the “M” shape data. The results of the analysis on the four “M” shaped die pair parameters are summarised in Table 5.4.

Table 5.4: This table lists the main mode influenced by each process parameter for the “M” shaped die pair case. The range of modes affected by each process parameter is also listed.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Main component(s) influenced</th>
<th>Range of variational modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Friction</td>
<td>4,11</td>
<td>1 – 11</td>
</tr>
<tr>
<td>Travel</td>
<td>2</td>
<td>1 – 3</td>
</tr>
<tr>
<td>Offset</td>
<td>1</td>
<td>1 – 3</td>
</tr>
</tbody>
</table>

5.4.2 Response surfaces of $\vec{b}$ vector components

The two data sets containing the calculated $\vec{b}$ vectors can be viewed as a response surface with respect to the process parameters. The surface is created by plotting a single component or mode of the $\vec{b}$ vector against several varying parameters. This gives a visual indication of the variation in the $\vec{b}$ component with respect to the process parameters. Two representative simple and “M” shape die pair response surfaces are seen in Figures 5-12 and 5-13 respectively.

The surface responses of the $\vec{b}$ vector modes give an indication of the trends of shape variation with respect to the process parameters. Figure 5-12(a) is the response surface of the most significant $\vec{b}$ component, $\vec{b}(1)$, with respect to friction and travel parameters (simple die pair case). As was noted previously, the travel effect on the first component is much greater than the friction effect for simple die pair data. However, because friction causes subtle shape changes, it has a greater effect on higher order (less significant) $\vec{b}$ components. Figure 5-12(b) shows how friction has a more dominant effect than the travel on the third component, $\vec{b}(3)$. Dominance in this case, means that
Figure 5-12: Diagram shows the simple die pair response surfaces of \( \vec{b} \) vector modes \( \vec{b}(1) \) and \( \vec{b}(3) \) when varying friction and travel while velocity = 10.0 mm/sec.

Figure 5-13: Diagram shows the "M" shaped die pair response surfaces of \( \vec{b} \) vector modes \( \vec{b}(1) \) and \( \vec{b}(3) \) when varying offset and travel while velocity = 10.0 mm/sec and the coefficient of friction \( \mu = 0.2 \).
Chapter 5. Forging processes

the shape variation mode changes significantly more when the dominant parameter is varied.

Similarly, the "M" shape data indicates that the offset parameter dominates the travel parameter in the most significant mode, see Figure 5-13(a). Furthermore, an interesting saddle surface occurs with the third mode, see Figure 5-13(b). This surface indicates there is no dominant parameter, but there is a substitution effect between the two parameters (travel and offset) with respect to the third mode values.

5.4.3 Varying the number of boundary points

The number of boundary points used to describe the forged shapes is varied to briefly investigate whether there is a loss of information. In this case the information is the separation between the \( \mathbf{b} \) vector signatures for different process parameter values. The results from reducing the number of boundary points are seen by comparing the friction parameter box plots for both simple and "M" shape die pair cases for 200, 100 and 6 boundary points in Figures 5-14 and 5-15. These numbers of points were arbitrarily chosen to obtain an idea of how the separation of the signature varied as the number of boundary points was decreased.

The decrease of boundary points reduces the amount of localised boundary sections that are measured. It is expected that this will cause the subtle shape variation information to be attenuated if not removed. The advantage of decreasing the number of boundary points is to reduce the amount of processing performed when analysing the shape data. If the attenuation in data is minimal for a particular decrease in the number of boundary points, then it is beneficial to use the lower number of boundary points to reduce analysis time and resource usage.

The previous section indicated that the friction parameter is influenced by the higher modes of variation (subtle shape variations). Hence, the friction parameter is expected to have reduced separation between the signatures of high and low process parameter levels as the number of boundary points is decreased.

In the simple die pair case, the signature of the friction parameter for high and low values is smoothed as the number of boundary points is decreased. At a low number of boundary points the large shape variation information is retained as can be seen with the differences between the box plots in the 6 boundary point box plots. The small differences, however, like that of the eleventh mode in both the 200 and 100 boundary point box plots, disappear in the 6 boundary point box plot, see Figure 5-14.

Similarly, there is definite smoothing of the signature as the number of boundary points is decreased for friction parameter in the "M" shaped die pair case. This can be seen again with the some higher modes (namely the eleventh mode) showing distinct differences in both the 200 and 100 boundary point box plots while the higher modes of the 6 boundary point box plot do not display any distinct differences, see Figure 5-15.
Figure 5-14: Diagram showing the friction process parameter box plots when varying the number of boundary points for the simple die pair case. Box plots (a) & (b), (c) & (d), and (e) & (f) have represented their forged parts with 200, 100, and 6 boundary points respectively. See Figure 5-5 for details on box plot notation.
Figure 5-15: Diagram showing the friction process parameter box plots when varying the number of boundary points for "M" shape die pair case. Box plots (a) & (b), (c) & (d), and (e) & (f) have represented their forged parts with 200, 100, and 6 boundary points respectively. See Figure 5-5 for details on box plot notation.
Table 5.5: This table contains the process parameter levels of the desired shape for both the simple and the "M" shaped die pair cases.

<table>
<thead>
<tr>
<th>Desired Shape</th>
<th>Process Parameter Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Friction</td>
</tr>
<tr>
<td>Simple die pair</td>
<td>(\mu = 0.20)</td>
</tr>
<tr>
<td>&quot;M&quot; shaped die pair</td>
<td>(\mu = 0.20)</td>
</tr>
</tbody>
</table>

Thus, the decrease of boundary points does indeed attenuate the small variations at high modes but it can amplify variations at low modes. The fidelity of the shape variation information, however, remains an open problem when the number of boundary points is decreased. In addition, the optimal number of boundary points to provide the best information is yet to be determined. The answers to these problems are probably specific to the type of system being investigated because the shape variation depends on the process being analysed.

5.5 Set-up of the forging feedback models

The shape variation of the two simulated forging processes have been analysed to determine which modes are significant. The shape error function and inverse feedback models are now initialised in this section to suit these system characteristics.

5.5.1 Shape error function

One of the important contributions this chapter makes is to use the variation modes created by the MSVMs to create a forging shape error function described by equation 4.21 in the previous chapter. According to this equation a desired shape is needed for comparison against the forged shapes.

The desired part's process parameter values are shown in Table 5.5. In fact any forged part resulting from an arbitrarily chosen process parameter combination, or indeed any shape that can be effectively described by the MSVM, can be chosen as the desired shape. This process parameter combination is chosen because it corresponds to those values recommended by Forge 2.7 for use in hot forging.

5.5.2 Discrete shape/parameter inverse modelling

Another contribution of this chapter is the development of an inverse model that relates the end shape of a forged part and the process parameters of the forging process. The model created in this section is a qualitative one and only classifies a part for each parameter into a discrete series of regions.
How the data is split into regions is described in Section 5.5.2.1. Classification techniques were used to discriminate between the different regions of the process parameters. Artificial Neural Networks (ANNs) were chosen as the classifier functions because they performed better than several other classifiers that were investigated due to reasons explained in Section 5.5.2.2.

5.5.2.1 Dividing the input parameter space into regions

All the process parameters are split into three separate regions (high, medium, low). The boundaries are chosen arbitrarily and without bias towards the data except that the desired shape is contained in the medium or normal region for each parameter. These boundaries are seen in Tables 5.6 and 5.7.

Table 5.6: The boundaries are given for the low, medium and high regions for the simple die pair case.

<table>
<thead>
<tr>
<th>Description</th>
<th>Low Region</th>
<th>Medium Region</th>
<th>High Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>Friction</td>
<td>μ = 0.10 - 0.15</td>
<td>μ = 0.15 - 0.25</td>
<td>μ = 0.25 - 0.30</td>
</tr>
<tr>
<td>Velocity</td>
<td>5.0 - 7.5 mm/sec</td>
<td>7.5 - 12.5 mm/sec</td>
<td>12.5 - 15.0 mm/sec</td>
</tr>
<tr>
<td>Travel</td>
<td>10.0 - 15.0 mm</td>
<td>15.0 - 25.0 mm</td>
<td>25.0 - 30.0 mm</td>
</tr>
</tbody>
</table>

Table 5.7: The boundaries are given for the low, medium, and high regions for the “M” shaped die pair case.

<table>
<thead>
<tr>
<th>Description</th>
<th>Low Region</th>
<th>Medium Region</th>
<th>High Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>Friction</td>
<td>μ = 0.10 to 0.15</td>
<td>μ = 0.15 to 0.25</td>
<td>μ = 0.25 to 0.30</td>
</tr>
<tr>
<td>Velocity (mm/sec)</td>
<td>5.0 to 7.5</td>
<td>7.5 to 12.5</td>
<td>12.5 to 15.0</td>
</tr>
<tr>
<td>Travel (mm)</td>
<td>31.0 to 33.0</td>
<td>33.0 to 37.0</td>
<td>37.0 to 39.0</td>
</tr>
<tr>
<td>Offset (mm)</td>
<td>-10.0 to -5.0</td>
<td>-5.0 to 5.0</td>
<td>5.0 to 10.0</td>
</tr>
</tbody>
</table>

Splitting the regions in this fashion enables the possibility of a qualitative output which, in the first instance, is easy to interpret. The coarseness of the regions can be reduced in the future to provide more quantitative information about the input parameters being observed.

5.5.2.2 Choice of classifiers

Once the parameters have been divided into regions, the type of classifier is chosen. There are several types of classifier that can be used, such as: statistical, rule based, decision trees or artificial neural networks (ANNs) (Schalkoff, 1992). The methods used have been described in chapter 3. Note that the data sets of the FEA simulations do not realize any statistical information about the forging process because the FEA
Table 5.8: This table presents typical accuracy results of applying the linear classifiers to the two forging set-ups using *ten fold cross validation*. Each confusion matrix, see page 55, splits the data points up into three regions. The left column is for low parameter points. The middle column is for medium parameter points. The right column is for high parameter points. The rows show into which region the points are classified.

<table>
<thead>
<tr>
<th>Description</th>
<th>Simple die pair</th>
<th>&quot;M&quot; shaped die pair</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Travel</td>
<td>Travel</td>
</tr>
<tr>
<td>Accuracy</td>
<td>56.4%</td>
<td>35.2%</td>
</tr>
<tr>
<td>Confusion Matrices</td>
<td>259 184 0</td>
<td>61 137 26</td>
</tr>
<tr>
<td></td>
<td>16 38 0</td>
<td>25 85 31</td>
</tr>
<tr>
<td></td>
<td>0 236 267</td>
<td>37 133 65</td>
</tr>
</tbody>
</table>

Table 5.9: This table presents typical accuracy results of applying the Gaussian quadratic classifiers when applied to the "M" shaped die pair data using *ten fold cross validation*.

<table>
<thead>
<tr>
<th>Description</th>
<th>Simple die pair</th>
<th>&quot;M&quot; shaped die pair</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Travel</td>
<td>Travel</td>
</tr>
<tr>
<td>Accuracy</td>
<td>65.6%</td>
<td>60.5%</td>
</tr>
<tr>
<td>Confusion Matrices</td>
<td>289 222 13</td>
<td>68 58 10</td>
</tr>
<tr>
<td></td>
<td>7 117 1</td>
<td>38 272 83</td>
</tr>
<tr>
<td></td>
<td>0 101 250</td>
<td>16 32 26</td>
</tr>
</tbody>
</table>

process is deterministic; a given combination of process parameters always produces the same shape and therefore the same $b$ vector.

Linear and quadratic Gaussian statistical classifiers were initially tested on the data. They gave unsatisfactory classification error levels, see Tables 5.8 and 5.9. The reason for the poor results is due to these classifiers making use of the between class variance matrices for each parameter. In this forging process example, these matrices are badly scaled. This is most likely due to the lack of any statistical variation in the data and the complex relationships that exist between the simulated forged shapes. Other statistical classifiers can be implemented, however estimates of the underlying probability distributions of each region have to be calculated before the classifiers are operational.

In contrast, ANNs are trained to learn relationships between input and output data and can therefore implicitly learn the probability distributions of the data sets. ANNs are thus chosen as the appropriate classifier to use because they provide non-linear classification without having to explicitly estimate the underlying probability distribu-

---

2For this chapter the classes are the regions defined in Section 5.5.2.1. The reason why the statistical classifiers use the variance matrices is explained in Section 3.3.1.1.
tions. This does not preclude decision trees and other non-statistical classifiers from also achieving similar results as the ANNs.

The type of ANN chosen is the feedforward back-propagated network which is fully connected between layers. The number of inputs to the system is determined by the range of modes that are influenced by the particular parameter, summarised in Tables 5.3 and 5.4 on page 91. Chapter 3 has made the point that two hidden layers are sufficient for providing near-optimal performance. In consequence, ANNs with one and two hidden layers are investigated to determine which structure gives the lowest misclassification errors. Similarly, experimentation with the number of nodes in each layer is used to find the structure with the lowest error. The starting number of nodes is set at half the number of inputs and the amount of nodes is increased or decreased depending on what minimises the misclassification error. The learning rate and momentum terms are set at 0.01 and 0.90 respectively. A maximum number of epochs is used as a stopping condition. This value is chosen to ensure convergence of the networks, and it was determined after some experimentation. Summaries of the final structures of the ANNs used to classify the data are found in Tables 5.10 and 5.11.

**Table 5.10:** The structural details of the ANN used for classifying the velocity, friction, and travel process parameters are presented for the simple die pair case.

<table>
<thead>
<tr>
<th>Description</th>
<th>Velocity</th>
<th>Friction</th>
<th>Travel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of inputs</td>
<td>11</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Hidden layers</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Neurons (hidden layers)</td>
<td>12,6</td>
<td>12,12</td>
<td>6,6</td>
</tr>
<tr>
<td>Neurons (output layer)</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Activation function</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
</tr>
<tr>
<td>Maximum epochs</td>
<td>150</td>
<td>150</td>
<td>150</td>
</tr>
</tbody>
</table>

**Table 5.11:** The structural details of the ANN used for classifying the velocity, friction, travel, and offset process parameters are presented for the “M” shaped die pair case.

<table>
<thead>
<tr>
<th>Description</th>
<th>Velocity</th>
<th>Friction</th>
<th>Travel</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of inputs</td>
<td>20</td>
<td>11</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Hidden layers</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Neurons (hidden layers)</td>
<td>12,12</td>
<td>11,6</td>
<td>3</td>
<td>3,3</td>
</tr>
<tr>
<td>Neurons (output layer)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Activation function</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
</tr>
<tr>
<td>Maximum epochs</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>
5.5.3 Continuous shape/parameter inverse modelling

This section explains the continuous models that relate shape error to the process parameters. The previous section is therefore extended such that the relationship between shape variation and process parameters is no longer qualitative but is made quantitative. The two types of models used to fit the two forged data sets of $\textit{b}$ vectors are linear regression and non-linear function fitting.

5.5.3.1 Linear regression model

The linear model is based on the following structure:

$$\bar{y} = X\bar{\beta} + \varepsilon$$

(5.3)

where $\bar{y}$ is a vector of the process parameter values for either friction, velocity, travel or offset. The process parameters values used as the output values are those explained in Tables 5.1 and 5.2 for the simple die pair case and the “M” shape die pair case respectively. The linear model’s parameters are represented by the vector $\bar{\beta}$ and the shape variation information is contained in the matrix $X$. The matrix $X$ consists of the first column vector being a vector of ones for the constant value of the linear model and the other columns are those of the $\textit{b}$ vector variance values for each mode. That is, the $i^{th}$ shape variation mode of every shape is contained in the $(i + 1)^{th}$ column vector of the $X$ matrix. The vector $\varepsilon$ is the residuals vector containing the errors or differences between the linear model and the output vector. Further analysis of the linear models is found in Appendix A.

5.5.3.2 Non-linear models

An artificial neural network (ANN) is used to determine a function to fit both sets of forging data for each process parameter. A feed-forward Levenberg-Marquardt back propagated type of ANN is used which is fully connected between all levels. Each ANN has two hidden layers and one output layer; and the number of neurons in each layer is given in Tables 5.12 and 5.13.

5.6 Results from the forging shape error function

This section applies the shape error function proposed in Section 4.4 to the data from the two forging processes. The shape error function is then used to examine certain properties and problems with respect to the two forging shape data sets. Table 5.14 shows the $\textit{b}$ vectors calculated for the desired shapes defined by Table 5.5 for both the simple die pair and the “M” shaped die pair cases.

A shape error surface can be created by varying two process parameters and calculating the shape error for each combination. This means holding one parameter constant.
Table 5.12: Table contains the structure of the ANN non-linear models for each simple die pair process parameter. The layer structure column contains the number of neurons in the first and second hidden layers and the output layer.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of inputs</th>
<th>Layer structure</th>
<th>Activation function</th>
<th>Maximum epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>20</td>
<td>3,3,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>3,3,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>3,3,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
</tbody>
</table>

Table 5.13: Table contains the structure of the ANN non-linear models for each “M” shaped die pair process parameter.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of inputs</th>
<th>Layer structure</th>
<th>Activation function</th>
<th>Maximum epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
</tr>
</tbody>
</table>
Table 5.14: Table containing the $\tilde{b}$ vector values for both the simple and “M” shaped die pair cases.

**Simple die pair desired vector**

<table>
<thead>
<tr>
<th>Mode</th>
<th>$\tilde{b}(1)$</th>
<th>$\tilde{b}(2)$</th>
<th>$\tilde{b}(3)$</th>
<th>$\tilde{b}(4)$</th>
<th>$\tilde{b}(5)$</th>
<th>$\tilde{b}(6)$</th>
<th>$\tilde{b}(7)$</th>
<th>$\tilde{b}(8)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>1.50</td>
<td>5.14</td>
<td>-0.38</td>
<td>0.11</td>
<td>0.28</td>
<td>-0.28</td>
<td>-0.43</td>
<td>0.41</td>
</tr>
<tr>
<td>Mode</td>
<td>$\tilde{b}(9)$</td>
<td>$\tilde{b}(10)$</td>
<td>$\tilde{b}(11)$</td>
<td>$\tilde{b}(12)$</td>
<td>$\tilde{b}(13)$</td>
<td>$\tilde{b}(14)$</td>
<td>$\tilde{b}(15)$</td>
<td>$\tilde{b}(16)$</td>
</tr>
<tr>
<td>Value</td>
<td>-0.12</td>
<td>-0.03</td>
<td>-0.02</td>
<td>-0.05</td>
<td>-0.06</td>
<td>-0.12</td>
<td>-0.03</td>
<td>0.02</td>
</tr>
<tr>
<td>Mode</td>
<td>$\tilde{b}(17)$</td>
<td>$\tilde{b}(18)$</td>
<td>$\tilde{b}(19)$</td>
<td>$\tilde{b}(20)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>0.04</td>
<td>-0.05</td>
<td>-0.09</td>
<td>0.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**“M” shape die pair desired vector**

<table>
<thead>
<tr>
<th>Mode</th>
<th>$\tilde{b}(1)$</th>
<th>$\tilde{b}(2)$</th>
<th>$\tilde{b}(3)$</th>
<th>$\tilde{b}(4)$</th>
<th>$\tilde{b}(5)$</th>
<th>$\tilde{b}(6)$</th>
<th>$\tilde{b}(7)$</th>
<th>$\tilde{b}(8)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>-6.11</td>
<td>-2.23</td>
<td>-5.11</td>
<td>18.38</td>
<td>3.41</td>
<td>1.76</td>
<td>0.81</td>
<td>2.05</td>
</tr>
<tr>
<td>Mode</td>
<td>$\tilde{b}(9)$</td>
<td>$\tilde{b}(10)$</td>
<td>$\tilde{b}(11)$</td>
<td>$\tilde{b}(12)$</td>
<td>$\tilde{b}(13)$</td>
<td>$\tilde{b}(14)$</td>
<td>$\tilde{b}(15)$</td>
<td>$\tilde{b}(16)$</td>
</tr>
<tr>
<td>Value</td>
<td>-0.03</td>
<td>-0.36</td>
<td>-1.04</td>
<td>-0.50</td>
<td>-0.63</td>
<td>-0.97</td>
<td>0.41</td>
<td>2.16</td>
</tr>
<tr>
<td>Mode</td>
<td>$\tilde{b}(17)$</td>
<td>$\tilde{b}(18)$</td>
<td>$\tilde{b}(19)$</td>
<td>$\tilde{b}(20)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>-0.74</td>
<td>0.59</td>
<td>1.42</td>
<td>0.32</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5-16: Diagram showing the shape error for both simple die pair and “M” shaped die data. The desired shape has the following parameters: Simple die pair case, coefficient of friction $\mu = 0.2$, velocity = 10 mm/sec, travel = 20 mm; “M” shaped die pair case, coefficient of friction $\mu = 0.2$, velocity = 10 mm/sec, travel = 35 mm, offset = 0 mm.
Figure 5-17: The shape error surfaces show that some simulations did not run to completion for the simple die pair case. The simulations that did not complete are those which break the pattern in the surface such as the canyon seen in diagram (a) for low velocity and friction $\mu = 0.26$ to 0.28, or in diagram (b) for high travel and friction $\mu = 0.26$ to 0.28.

in the simple die pair case or two parameters in the "M" shaped die pair case. The resulting shape error surfaces are plotted in Figures 5-16, 5-17, and 5-18.

The simple die pair shape error surface, Figure 5-16(a), has the travel parameter set lower than the desired shapes' travel. This means that there is an underlying error for each point on the shape error surface. The reason why the shape error is so much larger than zero is because travel is the dominant parameter. An interesting occurrence is that both decreasing the velocity or increasing the friction decreases the shape error, see Figure 5-16(a). The "M" shape die pair shape error surface, Figure 5-16(b), shows the shape error increases from the centre. The centre happens to be the desired shape. The offset parameter is dominant and this means that the variation in shape error between the extremes of the offset values and the median offset value creates a larger "valley" than that compared with the travel parameter.

When plotted, the shape error provides a visual way of checking the resulting shape variation of the end product. For example, when the shape error is plotted for the simple die pair case, Figure 5-17, it is seen that there are several holes in the shape error for high friction, high punch travel and low punch velocity. The reason for these "canyons" in the surface is that several simulations did not go through to completion. This was not found until the shape error contours were plotted because: firstly, the extreme values for the process parameters were successful; and secondly, the randomly tested samples did not fall into the set of non-completed simulations. The non-completed simulation manifests itself as a punch stopping at a travel value less than that required. The reason for the early termination is because the boundary of the forged shape is unable to be
Figure 5-18: The shape error surface in diagram (a) shows that there is larger shape errors for higher travel values in the "M" shape die pair case. Diagram (b) shows that the high travel, high velocity, medium negative offset and average friction combination has not completed.

automatically meshed for certain boundary sections. This occurrence is unusual for such a simple simulation. One would expect the automatic meshing functions to cope with the simple forging deformations. Furthermore, it is unusual that the extreme process parameter values completed correctly while some of the internal process parameter combinations terminated early. After some investigation it was found that this problem is fixed by introducing more finite element nodes on the simulated billet. These early terminations also caused a greater range of extreme values for high travel values in the simple die pair case which are seen in the box plots, see Figure 5-7 on page 91. This also impacts on the ability to identify the correct travel value in the following sections.

The shape error plot for the "M" shape die pair case, Figure 5-18(b), illustrates that the same problem has occurred as in the simple die pair case. There is a single simulation in this situation that has not processed to completion at high travel, high velocity, medium negative offset and average friction. This has created the large range of extreme values that is seen in box plot Figure 5-10 on page 93. The second effect that is shown by the shape error plot, Figure 5-18(a), is that the high travel values cause greater shape error. That is, there is greater shape error from the punch traveling past the desired level than when the punch does not achieve the desired level. One explanation for the greater error at higher travel values is that the desired shape is closer to low travel values and further away from high travel values. Figure 5-10 (page 93) shows the high travel simulations are marginally further away to the desired shape for the second mode than the low travel simulations. The desired shape has a second mode value of $-2.23$ while the high and low travel simulations have an approximate median
value of −100 and 80 respectively. Note that the average shape of each set of forged shapes has a $\mathbf{b}$ vector of zeros. Therefore, the higher travel values appear to cause greater shape distortion with respect to the desired shape which could be why there are slightly higher $\mathbf{b}$ vector values (in absolute magnitude) for high travel values than for low travel values.

In summary, the shape error function appears to give a quantitative value of shape difference between desired and other forged shapes. The shape error contours have assisted in the examination of the two forged shape data sets. It gave a visual representation in which the simulation anomalies were easily identified. Furthermore, trends regarding shape and process parameters were investigated. The next section extends this section’s brief investigation to develop some models between shape variation and the process parameters.

5.7 Results of the discrete shape/parameters inverse model

This section presents the results from the discrete inverse models proposed in Section 5.5.2. The following section (Section 5.8) then extends this study to find a continuous function between the shape variation and the process parameters.

To determine how well the ANN could be trained with respect to each data set a series of ten fold cross validation tests were performed. The ten fold cross validation test consists of splitting the data into ten equal subsets. Each subset is tested separately where the other nine subsets are used as training data. This method is less computationally demanding than the leave one out method which requires a classifier to be designed for each piece of data in the cross validation set. The data sets are of a reasonable size, 1331 simple die pair samples and 625 “M” shaped die pair samples, and therefore the leave one out method is computationally prohibitive.

Because the number of samples in each data set was not divisible by ten, 1000 random simple die pair samples and 600 random “M” shaped die pair samples were chosen to form the respective cross validation data sets. The results of the ten fold cross validation were collected in a confusion matrix. The resulting confusion matrices are found in Tables 5.15 and 5.16.

One of the ways to compare how well the classification is working is to compare the accuracies of the classifier with a random assignment classifier (James, 1985) which randomly classifies data into the number of classes or regions within the data. In the case of the three levels of each of the parameter’s data sets, the random assignment classifier would have an accuracy of 33.3%.

In the simple die pair case, the velocity data gives reasonable classification results with an accuracy of 77.0%. The high velocity region appears to be hard to classify with respect to the medium velocity region with the misclassification rate double that which occurs between the low and medium velocity region. The friction parameter has
Table 5.15: The accuracy results of applying the ANN classifiers to the simple die pair data using *ten fold cross validation*. Each confusion matrix splits the data points up into three regions. The left column is for low parameter points. The middle column is for medium parameter points. The right column is for high parameter points. The rows show into which region the points are classified.

<table>
<thead>
<tr>
<th>Description</th>
<th>Velocity</th>
<th>Friction</th>
<th>Travel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>77.0%</td>
<td>87.5%</td>
<td>95.3%</td>
</tr>
<tr>
<td>Confusion</td>
<td>253 36 26</td>
<td>236 18 17</td>
<td>239 25 0</td>
</tr>
<tr>
<td>Matrices</td>
<td>33 337 69</td>
<td>27 422 38</td>
<td>6 434 3</td>
</tr>
</tbody>
</table>

Table 5.16: The accuracy results of applying the ANN classifiers to the “M” shaped die pair data using *ten fold cross validation*.

<table>
<thead>
<tr>
<th>Description</th>
<th>Velocity</th>
<th>Friction</th>
<th>Travel</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>58.3%</td>
<td>83.5%</td>
<td>93.7%</td>
<td>88.0%</td>
</tr>
<tr>
<td>Confusion</td>
<td>2 8 3</td>
<td>89 15 4</td>
<td>109 0 6</td>
<td>81 7 0</td>
</tr>
<tr>
<td>Matrices</td>
<td>116 346 115</td>
<td>26 328 31</td>
<td>12 358 20</td>
<td>35 358 26</td>
</tr>
</tbody>
</table>

a better classification accuracy of 87.5%, but again the high friction region appears to be the most difficult to classify. The travel parameter has the highest classification accuracy of 95.3%, and it appears that the data on the margin between the lower and medium travel region is the hardest to classify.

In the “M” shaped die pair case, the offset parameter can be classified to an accuracy of 88%. This is lower than would be expected given the most significant mode appears to be largely affected by the offset parameter. There is possibly a large variance of the data occurring within each region which may make it difficult to discriminate data points situated near the boundaries of the regions. Once again the travel parameter is the easiest to classify with an accuracy of 93.7%. Unlike the simple die pair case, the misclassifications for “M” shape travel occur in the high travel region. It is interesting to note that the most accurate ANN for travel, see Table 5.11 on page 102, is one with a single hidden layer. This implies that the “M” shape travel parameter is linearly separable. The friction parameter is sufficiently classifiable with an accuracy of 83.5% given that friction changes only create subtle changes on the end forged part. The misclassifications appear to be spread evenly between the different regions. The velocity parameter is again the hardest to classify. The velocity parameter had very little influence on the end forged shape and there did not appear to be any relationship between changes in velocity and changes in shape. Perhaps the changes in the velocity parameter have more influence over the internal properties of the deformed billet which
cannot be measured by examining the shape of the final forged part. The confusion matrix shows that the classification of velocity is converging towards the optimal random classifier which is to always chose the region which is most likely to occur. That is, always choose the medium level because it is most likely to occur with a probability of 60%.

5.8 Results of the continuous shape/parameter inverse model

Section 5.5.3 explained the two types of models used to fit the data: the ordinary linear least squares regression model; and a non-linear function fitting model (neural network). This section presents results that compare the two types of models to determine which one gives the most accuracy and least amount of complexity. Adhering to the idea of Occam's razor\(^3\), the linear model is always chosen unless the neural network provides a significant improvement in accuracy or data prediction. Section 5.8.1 gives the results from both linear and non-linear models for the two forged shape data sets.

Furthermore, the fitting of models to the data does not provide any indication about how well the model fits a generalised set of forged data. Nor does it indicate how well the model predicts process parameter levels from shape variation data not used in the model creation. This is measured by using a modified cross validation technique in Section 5.8.2.

5.8.1 Fitting models to the forging data

5.8.1.1 Fitting linear models to forging data

The resulting residual sum of squares (RSS), \( \varepsilon' \varepsilon \), measures the amount of error between the linear model and the output vector. The RSS for each simple and “M” shaped die pair model is shown in Table 5.17. Unfortunately, the RSS values cannot be compared across the process parameters because the output values for each process parameter has a different range and scale. When examining the linear models RSS values for the same type of process parameters across the two forging processes it can be seen from Table 5.17 values that they are of the same order.

The square of the correlation coefficient, \( R^2 \), measures the linearity of data with respect to the model by dividing the variance of the estimated output (Var(\( X\hat{\theta} \))) or the explained sum of squares (ESS) over the variance of the actual output (Var(\( y \))) or the total sum of squares (TSS). Note that \( TSS = ESS + RSS \). Therefore, as the value of \( R^2 \) approaches 1 it implies that the data is increasingly linear. The \( R^2 \) values for the simple and “M” shaped die pair linear models are shown in Table 5.18.

\(^3\)Occam's razor essentially means that the simplest model that can explain the data should be the one that is chosen.
Table 5.17: The RSS for the simple and “M” shaped die pair linear parameter models.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Simple RSS</th>
<th>“M” Shaped RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>7323.9</td>
<td>7600.9</td>
</tr>
<tr>
<td>Friction</td>
<td>0.4262</td>
<td>0.4000</td>
</tr>
<tr>
<td>Travel</td>
<td>321.0</td>
<td>180.2</td>
</tr>
<tr>
<td>Offset</td>
<td>N/A</td>
<td>36.8</td>
</tr>
</tbody>
</table>

Table 5.18: The $R^2$ values for the simple and “M” shaped die pair linear parameter models.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Simple $R^2$</th>
<th>“M” Shaped $R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>0.4497</td>
<td>0.0271</td>
</tr>
<tr>
<td>Friction</td>
<td>0.9199</td>
<td>0.8563</td>
</tr>
<tr>
<td>Travel</td>
<td>0.9940</td>
<td>0.9640</td>
</tr>
<tr>
<td>Offset</td>
<td>N/A</td>
<td>0.9988</td>
</tr>
</tbody>
</table>

The $R^2$ values indicate that the travel data is quite linear while the velocity data appears to be non-linear for the simple die pair case. Similarly, the travel model and the offset model are therefore quite linear for the “M” shaped die pair case. Whereas, the “M” shaped die pair velocity model’s $R^2$ is so low that it would indicate the velocity data is just noise. This is consistent with the previous section which found the velocity data extremely difficult to classify into the correct level.

5.8.1.2 Fitting non-linear models to forging data

The second type of model used is the ANN. In the simple die pair case, the ANN’s RSS for each parameter has decreased in all cases with respect to the linear models, see Table 5.19. The travel model, however, has the least amount of improvement, as well as the least marginal gain by increasing the number of neurons in the network with the function fitting. The reason for this is that the travel data is very linear so a linear model fits the data so well that the non-linear models will only marginally increase the accuracy and added complexity is mainly redundant. There are very large decreases in RSS when the ANN is applied to both velocity and friction as well as large marginal gains by increasing the number of neurons in these networks. This indicates a possible higher structure that the linear model is unable to approximate for both velocity and friction.
Table 5.19: The RSS results for the ANN non-linear models for each parameter using simple die pair data.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of inputs</th>
<th>Layer structure</th>
<th>RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>20</td>
<td>3,3,1</td>
<td>1083.9</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>12,12,1</td>
<td>48.2</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>3,3,1</td>
<td>0.0793</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>12,12,1</td>
<td>0.0030</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>3,3,1</td>
<td>120.2</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>12,12,1</td>
<td>116.0</td>
</tr>
</tbody>
</table>

Similarly, the “M” shaped data indicates there is a decrease in the ANN RSS for all parameters with respect to the linear models, see Table 5.20. Once again the travel parameter appears to be the least improved by the adoption of a non-linear function fitting algorithm. The velocity parameter has an improved model but only half the gains, in percentage terms, that the simple die pair velocity parameter had when a non-linear model was applied. The error of the offset non-linear model increases as the number of neurons is increased. However, given these values are so small relative to the offset output value, which ranges from -10 to 10, the increase is probably due to rounding errors between the real and the fitted offset values.

Table 5.20: The RSS results for the ANN non-linear models for each parameter using “M” shaped die pair data.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of inputs</th>
<th>Layer structure</th>
<th>RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,1</td>
<td>2707.9</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,6,1</td>
<td>1855.5</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,1</td>
<td>0.0936</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,6,1</td>
<td>0.0023</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,1</td>
<td>51.1</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,6,1</td>
<td>44.0</td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>6,1, 5.5 x 10^{-14}</td>
<td></td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>6,6,1</td>
<td>7.5 x 10^{-4}</td>
</tr>
</tbody>
</table>

Considering the errors for both the linear and non-linear models, there has been a significant improvement for all parameters from using the non-linear models. However, in the case of the travel parameter, for both data sets, a simple non-linear model would capture more of the general trends in the data than the more complex models that had
only marginal gains in accuracy.

5.8.2 Prediction models of the forging data

Adopting the same strategy as is used in classifying the shape data in Section 5.7; the cross validation technique can be modified to give a qualitative indicator as to the prediction ability of the models. The technique developed first randomises the data list of each forging shape data set. The randomised lists are then split into ten equal subsets respectively. Each subset is held out while the rest of the data for that list is used to fit the model. After which the left-out subset is used to test the ability of the fitted model to predict data. Each subset is tested in the same way and the residual sum of squares (squared error) is summed and accumulated for each subset. The total RSS from this method can then be compared across function fitting types. The lowest RSS giving the best prediction ability.

A problem that can occur with training the ANNs to fit data is that they can over train and no longer become a generalised function for the data, but rather become a function that fits the specific data upon which it was trained. It is expected that an optimal structure can be found from this prediction method when investigating the non-linear models. As the complexity of the models is increased the models will fit more to the specific data rather than to the trends of the general data. The RSS is therefore expected to decrease towards the optimal complexity structure and there afterwards increase as the non-linear model over fits the data.

5.8.2.1 Forging data prediction using the linear regression model

The RSS's resulting from the modified cross validation technique for the simple and "M" shaped die pair cases are shown in Table 5.21.

<table>
<thead>
<tr>
<th>Parameter Model</th>
<th>Simple RSS</th>
<th>&quot;M&quot; Shaped RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>7882.9</td>
<td>8108.8</td>
</tr>
<tr>
<td>Friction</td>
<td>0.5102</td>
<td>0.4799</td>
</tr>
<tr>
<td>Travel</td>
<td>310.4</td>
<td>198.4</td>
</tr>
<tr>
<td>Offset</td>
<td>N/A</td>
<td>54.8</td>
</tr>
</tbody>
</table>

Note how similar the resulting RSS values are to the linear function fitting models in Section 5.8.1.1. This implies that the linear function fitted models have the equivalent amount of prediction power. Similarly, the RSS's resulting from the modified cross
validation technique for the "M" shape die pair case are very comparable to the linear function fitting RSS values.

### 5.8.2.2 Forging data prediction using non-linear models

Tables 5.22 and 5.23 contain the results from the simple and "M" shaped die pair case respectively, when using non-linear models to measure predictive power.

**Table 5.22:** The RSS values from ANN non-linear prediction models for each parameter are presented for the simple die pair case. The bolded RSS values are the minimum value for that parameter model.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of inputs</th>
<th>Layer structure</th>
<th>Activation function</th>
<th>Maximum epochs</th>
<th>RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>20</td>
<td>3,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>1807.7</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td><strong>1011.5</strong></td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>1330.4</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>3458.2</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>3,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td><strong>0.1382</strong></td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.1385</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.1453</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.7090</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>3,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>417.5</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td><strong>328.6</strong></td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>525.2</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>345.9</td>
</tr>
</tbody>
</table>

The results for the non-linear predictive models indicate that the best ANN structure for the velocity parameter is to have a single hidden layered ANN with 6 neurons in the hidden layer and one output neuron. The best ANN structure for the friction parameter is to have a single hidden layer with only 3 neurons in the hidden layer. The best ANN structure for the travel parameter is to have a single hidden layer with 6 neurons in the hidden layer. Most of the process parameters for the "M" shape die pair case, except the travel parameter, appear to have increased prediction power with increased complexity of the non-linear model.

### 5.9 Summary

In summary, there were two main contributions of this chapter. The first was the development of a forging shape error function that was successful for both simple and complex forging examples. The second contribution was the creation of an inverse model
Table 5.23: The RSS values from ANN non-linear prediction parameter models are presented for the “M” shaped die pair case. The bolded RSS values are the minimum value for that parameter model.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Number of inputs</th>
<th>Layer structure</th>
<th>Activation function</th>
<th>Maximum epochs</th>
<th>RSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>2673.6</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>2637.5</td>
</tr>
<tr>
<td>Velocity</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>618.6</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.0775</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.0293</td>
</tr>
<tr>
<td>Friction</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.0056</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>49.7</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>74.2</td>
</tr>
<tr>
<td>Travel</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>748.1</td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.0022</td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>6,6,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>0.1381</td>
</tr>
<tr>
<td>Offset</td>
<td>20</td>
<td>12,12,1</td>
<td>tan sigmoid</td>
<td>200</td>
<td>8.7 x10^{-6}</td>
</tr>
</tbody>
</table>

that maps forged shapes to process parameters to an average accuracy of 80%. This excludes the cases where there was no shape variation information available to make a reasonable classification. This inverse model was extended to create a continuous model between the shape variations and the process parameters.

Furthermore, an inverse shape error model was created by training an artificial neural network that could determine, to an accuracy of at least 77% for simple die pair data, whether the process parameters were too high or too low. The average accuracy of the inverse shape error model for the “M” shaped die pair data was approximately 80%. The inverse model was not accurate when determining velocity in the “M” shaped die pair case, however velocity did not have a large nor consistent effect on the forged shape. This was also reinforced when the inverse model was extended to a continuous model. The non-linear models were preferred over the linear models, but in the case of the travel parameter, a simple non-linear model was preferred over the more complex models.

It should be noted that this analysis of the final geometry assumes that varying the different input parameters of the process will produce distinct and different end geometries, otherwise the analysis will be intractable without some further knowledge. Similarly, this model relies on the inductive process, that is, it learns from specific examples and then abstracts to the general. The specific examples must therefore give enough information about the general system or otherwise the model will fail. In
conclusion, the identification of whether input process parameters are too high or low could, in the future, assist process control measures of forging processes.

This chapter's work has concentrated on simulated data. The next chapter progresses to apply the shape manufacturing feedback model to actual sheet metal data.
Chapter 6

Experimental sheet metal forming

6.1 Introduction

This chapter investigates applying the shape manufacturing feedback model to experimental sheet metal forming data. This is the second manufacturing system to which the shape manufacturing feedback model is being applied. In this case experimental data is used rather than simulated data because experimental data gives a true reflection of the process noise and measurement problems. Sheet metal forming, as a process, is where flat blank sheets of metal are plastically deformed or drawn into a specific shape using a punch and a die. Approximately half of the metal parts produced in the world are made via sheet metal forming (Schey, 1987).

The stamping of sheet metal products is a process that exhibits a large amount of unexplained variation. Blümel et al. (1986; 1988) initially investigated variation of the sheet metal process compared with the variation of material quality. They concluded
that sheet metal forming has inherent problems with process variation. More recently, Doolan et al. (1999; 2000) investigated the relationship between in-process variables, such as punch force, and quality outcomes, such as tearing and wrinkling. They developed an in-process control method for finding an operating window for punch force, as well as developing some empirical explanations for the causes of gross geometric variations in output. However, with regard to the end product of the stamping process, the geometry can be measured with increasing accuracy and speed using non-invasive techniques (Zeng et al., 2000). These data can be used to perform a more thorough and intelligent analysis of the stamping process. This chapter will extend the studies of Blümel et al. (1986; 1988) and Doolan et al. (1999; 2000) by analysing the shape variation in more detail and developing a manufacturing shape variation model (MSVM) that investigates the variation of the sheet metal process by examining the geometry of the end product. In this chapter the scope of sheet metal forming is limited to the standard “U” bend channel forming experiment. This is a standard test to measure the amount of springback for a particular material type.

The inherent variation in the sheet metal forming process causes numerous types of end product errors that need to be quantified, see Figure 6-1 for a general list of major problems according to Eary and Reed (1974). Umehara (1990) noted repeated try-outs and modifications of press dies increased lead times to production and accounted for 20% of the total die manufacturing and production cost. One of the major problems facing die designers is the issue of springback. Springback is where the deformed blank springs out of shape directly after forming due to residual stresses left within the material. As a die designer one must take into account the springback in the material when creating a new die.

The first contribution of this chapter develops a shape error function for sheet metal forming and concentrates on quantifying springback of the end product only. Within the springback prediction literature some models have been developed to make a quantifiable measurement of shape error (Karafillis & Boyce, 1996; Webb & Hardt, 1991) from the formed part’s geometric data. These methods were limited in two ways: first, there was no weighting of significant error problems over insignificant errors; and secondly, many of the methods needed to low pass filter or smooth the geometric data from the part’s surface before processing. The shape error function developed in this chapter weights significant errors over insignificant errors and does not need to filter the geometric data from the part’s surface.

The second contribution of this chapter is to develop an inverse shape variation-process parameter model that uses classification techniques to identify process set-up parameter levels by analysing the variations in geometric shape of a part from a sheet metal forming process. The vectors from the MSVM, which represent the major shape variations from the nominal sheet metal part, are linked to the appropriate levels of the process set-up parameters. The main difference between other inverse models in the
Figure 6-1: The major defects that can occur when sheet metal forming according to Eary and Reed (1974).
literature (Ghouati & Gelin, 2001; Guo et al., 2000) is that this inverse model assumes the material model is not available. The inverse model is developed from analysing the experimental data created from the MSVM.

The remainder of this chapter is structured as follows. First, the phenomenon of springback is discussed in section 6.2. The channel forming process is then described in section 6.3. The channel registration and boundary point placement are discussed in section 6.4. The feedback models are set-up in Section 6.5 which also includes the modification of the shape error equation, equation (4.21), for the use of measuring springback in Section 6.5.2. The results from the feedback models, including springback and process noise evaluation, are presented in Section 6.6. Finally, the results are discussed in Section 6.7.

### 6.2 Springback

Shape measures are used to quantify geometric variations away from the desired shape and there are many manifestations of such problems in sheet metal forming. According to Lee (1999), there are three major shape problems in sheet metal forming: wrinkling, fracture and springback. This chapter only considers springback. Springback is a type of shape error that occurs in metal forming where a deformed blank tends back to its original shape.

There are two main explanations for springback (Asnafi, 1998):

1. a non-uniform distribution of stress across the deformed sheet during forming;
2. the elastic recovery that occurs when the bending moment in the plastic region is released upon unloading of the die.

The definition of elastic recovery is schematically shown in Figure 6-2.

The simple case of springback is where a sheet is bent over a single curved radius with no tension, see Figure 6-3(a). In this case the bending moment causes a compressive strain on the inside edge of the blank and a tensile strain on the outside edge of the blank. During elastic recovery this causes an unbending moment to occur that reduces the curvature of the deformed blank. If tension is applied to the blank, the compressive stress and strain are reduced on the inside edge which in turn reduces the magnitude of the unbending moment upon unloading. Once the applied tension is greater than the yield point, then all the strain is in the plastic region; furthermore, there is no compressive strain to form the unbending moment upon unloading. The relationship between the bending moment and the elastic (unbending) moment and the tension is shown in Figure 6-4.

The situation where a blank is bent and then unbent over a radius is more complicated due to changes in the stress-strain curve, see Figure 6-2. Iwata et al. (2001)
Figure 6-2: Diagram showing the typical stress-strain curve, including the elastic and plastic regions, for both bending and the bending – unbending combination. The bending and unbending combination diagram is based on the results from Iwata et al. (2001). The yield stress, $-\sigma_Y$, on the unbending curve has been reduced due to the Bauschinger effect. In addition, the pseudo-elastic modulus is the slope of the curve just after unloading. The experimental pseudo-elastic modulus has decreased with respect to the isotropic hardening model.
recently measured experimental bending and unbending and discovered there is a large difference between the *isotropic hardening model* (Section 7.2.2.1) and the experimental bending – unbending stress-strain curve due to variation in the pseudo-elastic modulus and Bauschinger effects (Hill, 1967) in the material. The pseudo-elastic modulus is the momentary tangential slope of a material at the beginning of unloading. The Bauschinger effect is where a pre-strained material has a reduced yield stress in the opposite direction of the previous loading.

There are three main approaches for modelling or predicting springback: experimental, theoretical, and through finite element simulations. Davies (1981) was one of the first to approach springback in channel forming from an experimental perspective. This and subsequent studies have attempted to develop trends and relationships between empirical measurements of springback relative to varying certain process parameters. Duncan (1978) and then Johnson (1981) were first to approach channel forming from a theoretical point of view by using material deformation laws to try and explain the springback phenomenon. They created analytical equations to predict the amount of springback and side wall curl. Taylor *et al.* (1995), Karafillis *et al.* (1996) and successive modellers developed and improved finite element simulations of channel forming to predict springback. This chapter, however, is primarily concerned with finding a new way of describing springback and side wall curl using empirical data and will therefore concentrate on the analysis of the experimental channel forming literature.

Initially, the experimental approach developed trends from the results of pure bending and bending under tension tests. Davies (1981) found that the springback angle increased when the strength of the steel, the tool gap and die radius were increased for a pure bend system. Davies (1984) also investigated side wall curl by stretching a blank over a radius and discovered that side wall curl is better related to tensile strength than yield strength. In addition, Davies found that varying the die radius gave inconsistent results. Subsequently, several studies (Ayres, 1984; Davies, 1984; Liu, 1984) used experimental techniques to reduce the effect of springback and side wall curl. Later Umehara (1990) briefly investigated side wall curl with respect to die radius, punch load, tool gap and varying the strength of the steels. Umehara found that side wall curl increased as the die radius and tool gap was increased and decreased with blank holder force. Finally, Kim and Thomson (1989) conducted an extensive experimental analysis of springback and side wall curl for simple bending under tension using a “U” bending set-up similar to that described in Section 6.3. Kim and Thomson varied the blank holder force, tool gap, die radius, punch radius, friction and sheet thickness to develop some trends from the data. They concluded that springback increased with punch radius and tool gap and decreased with blank holder force and sheet thickness. Side wall curl, however, decreased with die radius and sheet thickness, but side wall curl increased with tool gap. This affect on side wall curl due to the die radius conflicts with Umehara’s conclusions. This chapter will rely on Kim and Thomson’s results due
Figure 6-3: The influence of tension on the stress and strain distributions for a blank formed using a single curved radius. Case (a) no tension is applied to the blank; Case (b) some tension is applied to the blank; Case (c) enough tension is applied to cause full plastic stress across the width of the blank.
Figure 6-4: Relationship between bending moment $M$ and tension $T$ in the elastic, elastic and plastic, and fully plastic regions, (Marciniak & Duncan, 1992). The assumption made is that the material is perfectly plastic above the yield point.

to the similarity in the experimental set-up. Note that the results of Kim and Thomson cannot be compared to those of Davies as the latter used pure bending, that is, no blank holder was used. The blank holder provides tension on the blank as it is being deformed.

All of the above studies develop trends by varying only one or two parameters within the one experiment. This chapter extends the analysis to three parameters by performing a factorial experiment with three major variables related to springback: blank holder force, tool gap and die radii. This factorial experiment will provide information about the interdependency between variables.

In addition, this chapter will develop a new method of measuring springback. Angle springback and side wall curl will be separated into the major modes of variation for the process. This can be viewed as vector of weighted modes, or it can be combined into one shape error function as is done in Section 6.5.2.

6.2.1 Springback and channel forming

This chapter investigates springback that occurs in channel forming, also known as “U” bend or “Hat” shaped channel forming. The experimental set-up for channel forming is discussed in the next section (Section 6.3). The channel forming experiment tests blanks for springback in both bending only and the bending – unbending combination. Springback in the channel forming experiment causes two main problems: angle springback of corner bends, and side wall curl. Angle springback is the result of a simple bend with a relatively small amount of tension applied on the lower corners of the channel. This causes the elastic component of the bending moment (within the plastic region) to be released after forming. Figure 6-5(a) shows the springback that occurs when a blank undergoes pure bending, that is, no tension is applied to the blank. Side wall
Chapter 6. Experimental sheet metal forming

Blank bent with no blank holder force

Springback, simple bend

Bending Moment (M)

Curvature (k)

Δk = springback

Side wall curl, bending and unbending

Bending Moment (M)

Curvature (k)

k_{swc} = Side Wall Curl

Figure 6-5: The simple explanation for angle springback and side wall curl. When combined these are the two effects that occur to deformed blanks when performing "U" bend channel forming.

(a) Angle Springback

(b) Side Wall Curl

Figure 6-6: The variables used to measure springback of a channel: θ is the lower corner springback angle; ρ is the side wall curl radius of curvature; and β is the upper or flange corner springback angle.
curl is a result of the upper corner of the blank being bent and then unbent under tension when passing over a die radius. This bending and unbending creates a residual bending moment in the blank, see Figure 6-5(b), that increases the bend angle unlike the simple bend where the bend angle is decreased. The definition of “curl” refers to the curvature of the blank after being bent and unbent.

For the standard “U” bend channel test, Kim and Thomson (1989) described spring-back by angles \((\theta, \beta)\) that were the difference between the actual and desired angles. Side wall curl was described by the radius of curvature \((\rho)\) of the side wall, (Figure 6-6). These measures are difficult to measure precisely. For example, it is often difficult to determine where the wall curvature begins and this may affect the calculation of the springback angle. Furthermore, using the angles and the radius of curvature is only useful for the “U” bend channel test.

6.3 “U” bend channel forming process

The “U” channel forming consists of forming a blank into a “U” shape while the edges of the blank (flanges) are held down by a blank holder.

The remainder of this section discusses:

- the set-up of the channel forming process in Section 6.3.1;
- the design of the experiment to create a set of channels for the MSVM in Section 6.3.2;
- the creation of a set of channels to monitor the noise of the channel forming process using the shape error function in Section 6.3.3.

6.3.1 Channel forming press set-up

All forming was conducted on a 30 ton Heine & Sons press with a 40mm wide punch. The experimental set-up (Figure 6-7) uses two load cells to measure the blank holder forces that were applied by the bolts. The bolts act through the top blank holder, the force sensor and lower blank holder onto the blank. The travel of the punch is measured by a linear potentiometer connected to the punch shoe.

The sample blanks were made out of Zincanneal G3N hot-dipped zinc/iron alloy coated drawing steel with a nominal sheet thickness of 0.76mm (Yield strength = 130–170MPa, Tensile = 280–320MPa, \(r_{45} = 1.2–1.6\)). Each blank was cut into (150mm x 20mm) strips with a guillotine and then the edges were de-burred. The blanks were stamped to a depth of approximately 32mm. The experimental procedure of how each blank was formed can be found in Appendix C.
Figure 6-7: Diagram (a) shows the schematic set-up of the “U” channel forming test and the five controllable process parameters: punch travel, punch radii, tool gap, die radii and blank holder force. Diagram (b) shows the photo of the actual set-up used to create the “U” channels.
Chapter 6. Experimental sheet metal forming

Table 6.1: Channel forming set-up process parameters levels.

<table>
<thead>
<tr>
<th>Set-up Parameter</th>
<th>Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blank holder force</td>
<td>B1 = 3KN</td>
</tr>
<tr>
<td></td>
<td>B2 = 7.5KN</td>
</tr>
<tr>
<td></td>
<td>B3 = 12KN</td>
</tr>
<tr>
<td>Die radii</td>
<td>D1 = 3.175mm</td>
</tr>
<tr>
<td></td>
<td>D2 = 4.7625mm</td>
</tr>
<tr>
<td></td>
<td>D3 = 6.35mm</td>
</tr>
<tr>
<td>Tool gap</td>
<td>T1 = 1.05mm</td>
</tr>
<tr>
<td></td>
<td>T2 = 1.30mm</td>
</tr>
<tr>
<td></td>
<td>T3 = 1.62mm</td>
</tr>
</tbody>
</table>

6.3.2 Experimental design

The experimental set-up was designed according to the specifications outlined in Appendix D. The design entailed conducting a pilot study investigating the effects of the five controllable sheet metal forming parameters on springback. The five controllable process parameters are:

- punch radius;
- die radius;
- tool gap;
- blank holder force; and
- lubrication.

Using statistical analysis the three most significant process parameters affecting the shape variation of the channels were:

- blank holder force;
- die radius; and
- tool gap.

The results from the statistical analysis are found in Appendix D.

This chapter investigates varying these three parameters, Blank Holder Force (B), Die Radii (D) and Tool Gap (T), to obtain geometric variations in the deformed channel. These parameters were varied according to a three-level, three-factor factorial experiment and the levels of the parameters are given in Table 6.1. Note that this chapter describes each set of repetition channels by its associated parameter levels. That is, the low blank holder force, medium die radii and large tool gap set of channels is denoted by “B1D2T3”. The MSVM is then developed via this set of standard 2-D drawn “U” bend channels.

After forming, each sample was scanned using a flatbed scanner to obtain its 2-D cross section. The scanned cross section was then imported into the analysis program and a chain coding (edge detection) algorithm, see Appendix F for details, was used to segment out the upper (inner) surface or boundary of the actual channel.
6.3.3 Noise in channel forming

This chapter also investigates using the shape error function to find out more about the noise in forming processes. Two entirely separate sets of channels were created in addition to the previous section's data set. The two sets consist of 20 channels each for the combinations of [low blank holder force (B1), small die radii (D1), small tool gap (T1)] and [medium blank holder (B2), medium die radii (D2), medium tool gap (T2)] parameter levels respectively.

For an example of the noise within the forming system we note that the tool gap was created by placing metal shims of the appropriate distance between the die and lowered punch and locking the die into the correct position. The medium tool gap used three shims to create the gap whereas the low tool gap used only two shims. There is greater potential for error when aligning three shims as opposed to aligning two shims. The results due to this error variation will be investigated in Section 6.6.5.

6.4 Channel alignment and point placement

Once the set of formed channels is created, each scanned channel in the set needs to be made consistent with the other channels in the set in two ways. First, each channel has points distributed across the boundary of the cross-section in a consistent manner. Second, the channels are aligned to remove registration (alignment) errors.

6.4.1 Boundary point distribution

There are many ways in which the boundary points can be distributed across the cross-sections of each channel. However, two issues must be addressed.

The first issue is that there should be a fixed number of points that describes the boundary surface of each channel. This means that each channel has corresponding boundary points that can be compared. This involves either increasing the number of points by creating redundant points or reducing the number of points by removing irrelevant points where possible. The former strategy was chosen so as not to remove any shape information.

The second issue is how the boundary points are distributed across each channel. Corresponding boundary points on different channels must represent the same region of the "general" channel for accurate comparisons between channels.

The boundary point distribution system is developed in the following way:

1. select reference point(s) from which the distribution system can be based;

2. choose a distribution system.
The remainder of this section will discuss the selection of the reference point in Section 6.4.1.1. This is followed by a description of the methods attempted to distribute points in Section 6.4.1.2. Finally, the chosen distribution method will be discussed in detail in Section 6.4.1.3.

### 6.4.1.1 Selection of a reference point

The first step to developing a boundary point distribution system is to find a reference point. The main reference points for the "U" channel are:

- the ends of the flanges;
- the corners (upper and lower) of the channel;
- the mid-point of the floor.

![Diagram showing the inconsistency of the flanges and corners of the channel, while showing the consistency of the mid-point of the floor.](image)

**Figure 6-8:** Diagram showing the inconsistency of the flanges and corners of the channel, while showing the consistency of the mid-point of the floor.

The problem of having the ends of the flanges as reference points is that there is no consistency in length nor angle of the flange, see Figure 6-8. Similarly, the position and angles of each corner are also inconsistent. The most consistent point of the set of channels is the mid-point of the floor of each channel. The forming process has a very minor effect on the floor of the channel, although there is occasionally a slight bow in the floor. The bow tended to occur at low blank holder forces but only varied the vertical height of the mid-point by at most 0.5mm, whereas the variations of the flanges of the channel could vary up to 10mm. The mid-point is also more consistent than the other features because the width of the punch is the same for all the samples.

The channel floor's mid-point is found by bisecting the imaginary line between the two corners of the channel, see Figure 6-9. The corners of the channel are determined by using orthogonal regression to fit five lines to the two flanges, two side walls and the floor of the channel. Orthogonal regression differs from the standard regression
Chapter 6. Experimental sheet metal forming

6.4.1.2 Choosing a point distribution method

After the channels have been scanned and edge detected, the number of points describing each channel varies. The chain code (see Appendix F) does the edge detection and follows the boundary of the channel and records the pixels that it finds on the boundary. The resulting "chains" are lists of $x$ and $y$ coordinates,

$$\hat{X}_{\text{chain}} = \begin{bmatrix} x_1 - x_{\text{mid}} & y_1 - y_{\text{mid}} \\ \vdots & \vdots \\ x_n - x_{\text{mid}} & y_n - y_{\text{mid}} \end{bmatrix}$$

where $(x_{\text{mid}}, y_{\text{mid}})$ is the mid-point and by subtracting the mid-point from the chain code, the mid-point is made the origin of the channel. The value of $n$ varies with each chain.

Several methods of boundary point distribution systems were then examined:

- Arbitrary fixed length distances between points;
- Equidistant distances between points;
- Fixed numbers of points between recognisable features.

The problem with the arbitrary fixed length method is that the flange lengths on each side of the channel are not necessarily going to be equivalent and the fixed length distance between neighbouring points will not be able to properly reflect this. The arbitrary length may not be the appropriate spacing that will enable a boundary point to be on the end of each flange. The problem with the equidistant spacing method from the mid-point is that the channels are not all symmetric, that is, one flange can...
be longer than the other. If one flange is longer than the other, then one flange will have more points than the other. This will lead to a correspondence problem because points on one channel will represent one feature while on another feature they will represent something different.

The final method proposed is to assign a fixed number of points between recognisable features. The most recognisable features for the “U” channel are: ends of flanges, corners and the mid-point. We use the mid-point as a reference point and this is the first chosen feature. The second feature chosen is the end of the channel’s flange. These were chosen because they are easy to find and they indicate the most about the shape in terms of symmetric information. After the appropriate features have been found (mid-point, end of flange), a fixed number of points are assigned in an equidistant fashion between the mid-point and the ends of the flanges, see Figure 6-10(a).

This method solves the problem of the asymmetric channels as the points adjust to the size of the flange. However, if the number of points is small and the degree of asymmetry large then there will still be some correspondence problems. In this case certain corresponding points will represent part of one region in one channel (for example, the upper corner) and represent a different region in another channel (for example, side wall), see Figure 6-10(a). If this problem continues to occur then the corners can be included as features and then three segments of points could be used for each side (see Figure 6-10(b)): mid-point to lower corner; lower corner to upper corner; and upper corner to end of the flange.

**Figure 6-10:** Fixed number of points between features where the points are assigned in an equidistant fashion between the mid-point and the ends of the flanges. The second diagram shows how the “U” channel could be segmented, if necessary, into three regions.
6.4.1.3 Details of chosen point distribution method

The "chain" for each channel is modified via the "fixed number of points between features" point distribution system, described in the previous section, to make each channel have the same amount of points.

First the mid-point is found. The lengths from the mid-point to the left and right ends of the flanges are calculated by summing the straight line distances between each point between the two features. A fixed number of points are then placed in an equidistant fashion along the channel on the left and right sides. The equidistant length is found by dividing the length from the mid-point to the end of the flange by the number of fixed number of points.

\[
\begin{align*}
N &= 2 \times \text{fixed number} + 1 \\
(x_{\text{fixed number}+1}, y_{\text{fixed number}+1}) &= \text{the mid-point}.
\end{align*}
\]

The points are then distributed with equal numbers of points being positioned on each half (left and right) of the channel. The channels are described using 201 boundary points \((N = 201)\). The final point distribution is shown in Figure 6-11 where every second boundary point is marked with an "x" for clarity.

6.4.2 Alignment of the channels

The second issue of consistency is the alignment of the channels now that the number of points has been fixed and the points have a high degree of regional correspondence. The alignment of the channels is one of the most important steps in the analysis of the channels because the aligned channels should only show the variations which are due
to the process parameter changes or noise in the process. This means that the aligned channels should minimise the rotation and translation error types of errors that can be seen in Figure 6-12.

![Diagram showing the three types of alignment errors.](image)

**Figure 6-12**: Diagram showing the three types of alignment errors.

Initially the mid-point of the floor of each channel is found using the method described in Section 6.4.1.1 and this is used as the first alignment, see Figure 6-13. By aligning the mid-points of all the channels, the Y translation error is removed from the channel data set. The next alignment process therefore only needs to adjust the rotational aspect of the channels and make small adjustments to the horizontal (X) translations of the channels.

There are many methods of alignment that can be used following the initial alignment of the channels, and following are several methods that were considered:

- **Method 1**: minimising the sum of weighted squared distances of all the points with respect to the points of the average shape;
- **Method 2**: minimising the weighted difference between the slopes of the fitted lines of the floors of the U shapes and the average shape;
- **Method 3**: minimising the sum of weighted distances from the points of each channel to the fitted lines of the average channel.

The problem with Method 1 is that some of the flanges are longer than the average shape and this confuses the alignment process by placing too much significance on the flanges and not the floor of the channels. Method 2 solves this problem by comparing the slopes of the flanges, placing the flanges on an equal significance as the floor and the side walls. The problem with Method 2 is that it does not enable the X translation modification because the slopes of the channel do not change with an X translation. Method 3 minimises the orthogonal distance between the aligned channel's points and
the average channel's fitted lines. This removes the problem of the significance of the flanges by finding the minimum distance of the points on the flange to the fitted lines of the average shape. It also reduces the significance of the longer flanged channels by removing the flange correspondence errors that dominate the errors in Method 1. Finally, it also enables the X translation alignment which can fine tune the set of channels in the horizontal (X) direction.

The registration error for an individual channel then is gradually removed by an iterative process which rotates and translates the channels to minimise the error between the mean shape of the channels and each particular channel using Method 3. The registration error is calculated by first finding the mean channel's orthogonal lines that match the five main features, two flanges, two side walls and the floor.

$$\left[ \begin{array}{c} \text{Average Channel's} \\ \text{Orthogonal Lines} \end{array} \right] (\text{ACOL}) = \left\{ \text{lines: } \mathcal{F}_{\text{orthogonal regression}} \left( \hat{X}_{\text{mean channel}} \right) \right\}$$

The registration error is then defined as the sum of the minimum distances between each shape's boundary points and the Average Channel's Orthogonal Lines (ACOL). Because the floor of the channel is the most consistent feature, and this feature should be matched over other features, the error away from the average channel's floor was weighted to a value five times the error of the side wall or the flange. Each channel can then be separately rotated and translated to minimise this error. The minimisation of the registration error is achieved by the following process.

*Registration steps*

1. Calculate the average channel and the ACOL.

2. Loop around each channel in the data set.
2.1. Minimise channel registration error by rotating and translating the channel:

\[
\text{Channel Registration Error (} \theta, T) = \sum_i^N \text{Dist} \left[ \hat{X}(x_i, y_i) \ W(\theta, T), \ ACOL \right],
\]

\[
\min_{\theta, T} \text{[Channel Registration Error (} \theta, T)]
\]

where Dist[., .] is a function that calculates the weighted minimum distance between a point and a line. If the minimum distance is with respect to the floor line, then the distance is weighted by a factor of 5. \( N \) is the number of points that describe the channel, note \( N = 201 \). \( W \) is the combined translation and rotation matrix which translates and rotates the point coordinates by \( \theta \) and \( T \) respectively.

3. Calculate the total registration error,

\[
\text{Total Registration Error} = \sum_i^M \min \text{. Channel Registration Error} \left( \hat{X}_{(\text{channel } i)} \right),
\]

where \( M \) is the number of channels in the data set and \( \hat{X}_{(\text{channel } i)} \) is the point list for the \( i^{th} \) channel.

4. If the total registration error has not yet converged
then return to Step 1,
otherwise end.

This process is continued until the error between the average shape and the rest of the channels converges to an acceptably small level.

6.5 Setting up the feedback models

The aligned channels are used to create the experimental channels' MSVM using the PDM approach (Cootes et al., 1995) described in Section 4.3. The feedback models (shape error function and inverse shape variation - process parameter model) then need to be initialised to obtain the results that are given in Section 6.6.

6.5.1 Shape error function

The shape error function was created using the equations developed in Section 4.4. The desired shape to be used as a comparison was drawn in CAD to have medium tool gap (\( T2 = 1.30 \text{mm} \)) and medium die radii (\( D2 = 4.7625 \text{mm} \)). Note that this desired shape is an artificial channel and this means that sizeable shape errors may exist even when compared to the experimental T2D2 channels.
Figure 6-14: Diagram shows the “springback free” channel compared with the average channels of the same parameter combination, high tool gap (T3) and high die radii (D3).

6.5.2 Springback shape error equations

Given the definition of the general shape error equation, equation (4.21), this section describes how this measure is extended to measure springback. The springback error equation is the same as the error equation for an individual channel except that the desired channel is now an artificially created channel without springback errors (springback free) but with the same corresponding process parameters (such as, die radii, punch radii and tool gap). Therefore, when calculating the springback error for each channel in the data set, a separate “springback free” artificial channel is created for each process parameter combination. That is, the artificial channels have no side wall curl, nor any splaying of the floor angle, nor the decrease in flange angle. The “springback free” channels were created in CAD for all corresponding combinations of die radii, punch radii and tool gap levels that were used to create the channel data set. The resulting artificial channels were plotted and scanned like the original channels in the data set. The “springback free” channels were then aligned to the average channel of the data set and their $\vec{b}$ vectors were calculated using the experimental channels' MSVM.

Hence, each channel in the original channel data set was compared against its “springback free” equivalent channel using the shape error measure equation as follows:
where the "springback free" channel has the same corresponding parameter values as the manufactured channel but the artificially created channel has no springback error. That is, each "springback free" channel is the desired channel for that particular parameter combination. An example of a "springback free" channel is shown in Figure 6-14. It includes the channels with the same process parameter combinations with which it will be compared.

### 6.5.3 Set-up of the inverse models

The inverse models are created using classification under supervised learning where the shape variations of the channels are related to the set-up parameter levels that formed each channel. Hence a type of classifier must be chosen to develop the relationship between the shape variations and the process parameters. There are several types of classifier that could be used: statistical, rule based, decision trees or artificial neural networks (ANNs) (Schalkoff, 1992).

The linear Fisher criterion and quadratic Gaussian statistical classifiers (Fukunaga, 1990) are tested on the data. The results of these statistical classifiers will be compared against an ANN. The type of ANN chosen was the feedforward back-propagated network which was fully connected between layers. The number of inputs to the system is determined by experimentation that give the lowest classification error. In most ANN applications, the use of one or two hidden layers seems to provide near-optimal performance, see Section 3.3.2. In consequence, ANNs with one and two hidden layers were investigated to determine which structure gives the smallest misclassification errors. Similarly, experimentation was done with the number of nodes in each layer to find the structure with the smallest error. The starting number of nodes for each layer was set at one node and the amount of nodes was increased to one and a half times the number of inputs. A summary of the final structure of the ANNs used to classify the data is found in Table 6.2.

### 6.6 Results

The results of this chapter are based on the 135 channels ($3^3$ combinations with 5 channels in each combination) created from the "U" channel forming process outlined in Section 6.3. Figure 6-15 shows the 5 channels for the combination B1D1T1 and
Table 6.2: The best ANN classifiers’ structural details.

<table>
<thead>
<tr>
<th>Description</th>
<th>Blank Holder Force</th>
<th>Die Radii</th>
<th>Tool gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of inputs</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Hidden layers</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Neurons (hidden layers)</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Neurons (output layer)</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Activation function</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
<td>tan sigmoid</td>
</tr>
<tr>
<td>Maximum epochs</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>

Figure 6-15: This diagram shows the B1D1T1 experimental channels and the associated $\vec{b}$ vectors. The solid line represents the average B1D1T1 channel and the dotted lines are the 5 B1D1T1 channels.
the associated $\vec{b}$ vectors with 20 variation modes. The $\vec{b}$ vectors for each channel are the result of using the MSVM on each channel. Due to the large number of channel combinations and $\vec{b}$ vector modes, only the relevant modes or channel combinations will be presented in this Results section.

The shape variation modes are presented in Sections 6.6.1 and 6.6.2 to give the reader an indication of the trends in the experimental channels' MSVM. It also indicates which process parameters are influential on the MSVM. The results for the first feedback model, shape error, are then presented in Section 6.6.3. These results are compared with manually measured results to check the validity of the shape error function. The applications of the shape error function and resulting shape error data are revealed in Section 6.6.5. Finally, Section 6.6.6 states the results received from the inverse shape variation - process parameter model.

6.6.1 Shape variation modes

The accuracy of the shape variation/process parameter model is dependent on the modes of geometric variation from the sheet metal process. The grossly emphasised first three modes of geometric variation are shown in Figure 6-16. The first mode indicates that the most significant change between the channels is the length of the flange and the decrease of the flange/side wall angle. The second mode tracks the imbalance between the lengths of the flanges. The third mode appears to track the side wall curl of the channels. The significance of each mode is given by the weighting factor under each of the emphasised modes. The first mode is approximately five times more significant than the third mode.

6.6.2 Response surfaces of the variational modes

The shape variational modes of the channels can be viewed as response surfaces if the average $\vec{b}$ vector value for a particular mode is plotted while varying two process parameters, see Figure 6-17. The most significant mode's, $\vec{b}(1)$, response surface (see Figure 6-17(a)) indicates that the blank holder force has a dominant effect on the most significant mode, particularly at high blank holder forces. The term dominant implies that blank holder force influences that particular $\vec{b}$ vector mode much more than the other parameters. In the response surface figures, dominance is seen as a steep slope (large variation of the mode value) of the surface as a particular process parameter is varied. The tool gap parameter does not appear to have a consistent effect on the most significant mode.

The die radii and tool gap parameters do not have a such a prominent effect on the shape of the channel as the blank holder force. It may be possible that die radii and tool gap may be more consequential in subtle shape changes of the channel. The die radii parameter dominates the fourth shape variation mode, $\vec{b}(4)$, as can be seen in
Figure 6-16: The grossly emphasised PDM variational modes of the data set. Note that the positive perturbation is the dotted line and the negative perturbation is the dashed line and the significance of each mode is reflected in the weighting factor below each graph. The first through to the third modes track (a) flange length; (b) flange imbalance; (c) side wall curl, respectively.
Figure 6-17: (a) The surface response of the first $\vec{b}$ vector mode; (b) the surface response of the fourth $\vec{b}$ vector mode; and (c) the surface response of the ninth $\vec{b}$ vector mode.
Figure 6-17(b). Note also that the change from the small die radii to the average die radii appears to have more affect on the fourth mode than changing to the large die radii.

The tool gap parameter is the least dominant parameter and this can be seen in Figure 6-17(c). This figure shows the response surface where the tool gap is the most effective and yet the blank holder force still has a significant influence on the $\tilde{b}(9)$ response surface. The lack of an effect on the resulting shape variation modes implies that it will be difficult to develop an accurate relationship between the tool gap parameter and shape variation modes.

### 6.6.3 Shape error surfaces

The focus of this section is to examine the shape error surfaces that occur when the forming process parameters are varied and to discuss the validity of the shape error function. The validity of the model is investigated by comparing the manual error measurements against the shape error function outcomes.

The resulting shape error from the channel data set is dependent on the modes of geometric variation from the sheet metal process. The first mode is approximately 5 times more significant than the third mode and this affects the resulting shape error value. The shape error is graphed as error surfaces in Figure 6-18. The shape error surfaces are created by plotting the average channel's shape error for all the parameter combinations of die radii and tool gap while holding the blank holder force at a fixed level. Note that the average channel for each parameter combination is the equivalent channel created from the average $\tilde{b}$ vector of the five repetition channels for each parameter combination.

From the shape error surfaces it can be inferred that increasing the blank holder force decreases the shape error as the range of the mean $\tilde{b}$ vector shape error decreases from Figure 6-18(a) to Figure 6-18(c). There appears, however, to be inconsistent relationships between shape error and the other two parameters.

A comparison of the manual error measurements of all the average channels in the channel data set was undertaken to check the validity of the shape error function. It revealed that the shape error measure gives reasonable results with respect to manual inspection, particularly for low and medium blank holder force. At high blank holder force, however, the results were not as promising. The manually measured springback features, for example flange angles, are collated in Table 6.3. Due to the large number of process parameter combinations only a few representative sets of combinations will be analysed. In the following sections the analysis is split into the three blank holder force levels because blank holder force is the most significant process parameter.
Figure 6-18: Shape error surfaces where the shape error is plotted against varying tool gap and die radii while the blank holder force is held at a particular level.
### Table 6.3: Manually measured flange angles and lengths and floor angles for the average and desired channels.

<table>
<thead>
<tr>
<th>Channel Set</th>
<th>Flange Angle, $\beta$</th>
<th>Flange Length</th>
<th>Floor Angle, $\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Left</td>
<td>Right</td>
<td>Left</td>
</tr>
<tr>
<td>Desired</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>93.6°</td>
<td>93.9°</td>
<td>33.8mm</td>
</tr>
<tr>
<td>B1D1T1</td>
<td>87.6°</td>
<td>87.3°</td>
<td>30.1mm</td>
</tr>
<tr>
<td>B1D1T2</td>
<td>88.3°</td>
<td>88.4°</td>
<td>29.6mm</td>
</tr>
<tr>
<td>B1D1T3</td>
<td>88.5°</td>
<td>88.5°</td>
<td>29.1mm</td>
</tr>
<tr>
<td>B1D2T1</td>
<td>88.8°</td>
<td>88.4°</td>
<td>30.3mm</td>
</tr>
<tr>
<td>B1D2T2</td>
<td>89.7°</td>
<td>89.0°</td>
<td>30.5mm</td>
</tr>
<tr>
<td>B1D2T3</td>
<td>89.7°</td>
<td>89.5°</td>
<td>30.7mm</td>
</tr>
<tr>
<td>B1D3T1</td>
<td>88.7°</td>
<td>88.7°</td>
<td>30.3mm</td>
</tr>
<tr>
<td>B1D3T2</td>
<td>89.2°</td>
<td>89.5°</td>
<td>31.0mm</td>
</tr>
<tr>
<td>B1D3T3</td>
<td>90.1°</td>
<td>90.3°</td>
<td>30.1mm</td>
</tr>
<tr>
<td>B2D1T1</td>
<td>88.8°</td>
<td>88.8°</td>
<td>31.0mm</td>
</tr>
<tr>
<td>B2D1T2</td>
<td>89.7°</td>
<td>90.0°</td>
<td>31.2mm</td>
</tr>
<tr>
<td>B2D1T3</td>
<td>90.1°</td>
<td>90.5°</td>
<td>31.0mm</td>
</tr>
<tr>
<td>B2D2T1</td>
<td>90.2°</td>
<td>89.8°</td>
<td>31.2mm</td>
</tr>
<tr>
<td>B2D2T2</td>
<td>90.6°</td>
<td>90.5°</td>
<td>31.3mm</td>
</tr>
<tr>
<td>B2D2T3</td>
<td>91.2°</td>
<td>91.0°</td>
<td>31.5mm</td>
</tr>
<tr>
<td>B2D3T1</td>
<td>89.6°</td>
<td>90.0°</td>
<td>31.7mm</td>
</tr>
<tr>
<td>B2D3T2</td>
<td>90.5°</td>
<td>90.4°</td>
<td>31.3mm</td>
</tr>
<tr>
<td>B2D3T3</td>
<td>91.2°</td>
<td>91.2°</td>
<td>31.2mm</td>
</tr>
<tr>
<td>B3D1T1</td>
<td>90.6°</td>
<td>91.3°</td>
<td>33.0mm</td>
</tr>
<tr>
<td>B3D1T2</td>
<td>91.5°</td>
<td>92.1°</td>
<td>33.0mm</td>
</tr>
<tr>
<td>B3D1T3</td>
<td>92.1°</td>
<td>92.6°</td>
<td>33.3mm</td>
</tr>
<tr>
<td>B3D2T1</td>
<td>91.7°</td>
<td>91.8°</td>
<td>34.3mm</td>
</tr>
<tr>
<td>B3D2T2</td>
<td>92.9°</td>
<td>92.9°</td>
<td>33.7mm</td>
</tr>
<tr>
<td>B3D2T3</td>
<td>93.4°</td>
<td>93.3°</td>
<td>33.2mm</td>
</tr>
<tr>
<td>B3D3T1</td>
<td>91.2°</td>
<td>91.1°</td>
<td>33.8mm</td>
</tr>
<tr>
<td>B3D3T2</td>
<td>92.6°</td>
<td>92.8°</td>
<td>33.6mm</td>
</tr>
<tr>
<td>B3D3T3</td>
<td>93.1°</td>
<td>93.0°</td>
<td>33.5mm</td>
</tr>
</tbody>
</table>
Figure 6-19: Comparing the low blank holder force average channels sets (B1D1T3 and B1D3T3) and their associated absolute $\vec{b}$ vector errors with respect to the desired shape.

6.6.3.1 Low blank holder force

First, consider two representative low blank holder force sets, B1D1T3 and B1D3T3, where the die radii is varied. Figure 6-19 displays the differences where the low die radii (B1D1T3) channel and the high die radii (B1D3T3) channel are a fine dotted line and a dashed line respectively. The difference between the average channels of the low die radii and high die radii sets are that the high die radii channel has:

- more rounded upper flange corners;
- a better flange angles and lengths.

The absolute $\vec{b}$ vector error (absolute difference between the desired channel's and the particular channel's $\vec{b}$ vector) between the two channels indicates that the low die radii channel has a higher first mode error but the high die radii channel has a higher third mode error which accords with the high die radii channel having a longer flange but more rounded corners. The high die radii channel has much lower shape error than the low die radii channel and this accords with the manually measured errors of the difference in the flange angles and lengths.

6.6.3.2 Medium blank holder force

Now consider the difference between the average channels for representative sets of medium level blank holder force. Figure 6-20(a) displays the desired channel (solid line), the small die radii and low tool gap (B2D1T1) average channel (fine dotted line) and the medium die radii and tool gap (B2D2T2) average channel (dashed line). The
two channels are quite similar except that the small die radii and low tool gap channel has:

- a slightly tighter corner radii;
- a less splayed tool gap (floor angle);
- the flange angle is further away from the desired angle;
- the flange length is not as long as the medium die radii and tool gap channel on the left flange but is longer on the right.

The absolute $\vec{b}$ vector error indicates that the small die radii and low tool gap channel has a larger error in the first mode with respect to the medium die radii and tool gap channel. This implies the low tool gap channel is shorter in the flanges than the medium die radii and tool gap channel with respect to the desired shape but the higher second mode error in the medium tool gap channel indicates this channel is more imbalanced than the low tool gap channel. The medium tool gap and die radii channel has lower shape error than the low tool gap and die radii channel and this result agrees with the manually measured results.

6.6.3.3 High blank holder force

Finally, the average channels of two representative high blank holder force sets, B3D3T2 and B3D2T2, are compared where the die radii is varied. The two channels (B3D3T2 and B3D2T2) are very similar as seen in Figure 6-21. The high die radii channel has the following differences:
Figure 6-21: Comparing the high blank holder force average channels sets (B3D3T2 and B3D2T2) and their associated absolute $\vec{b}$ vector errors with respect to the desired shape.

- a slightly smaller flange angle;
- a longer right flange;
- the corner radii are markedly more rounded.

The absolute $\vec{b}$ vector error suggests that the medium die radii channel has higher error in the first two modes. This implies that the high die radii channel has longer and more balanced flange lengths. However, the high die radii channel has a much higher error in the third mode, which indicates the high die radii channel has more rounded corners or a lot of side wall curl. The shape error for the high die radii channel is smaller than the shape error for the medium die radii channel. This result is ambiguous with respect to the manual measurements because although the high die radii channel is more balanced in the flange length, its flange and floor angles are worse.

6.6.4 Springback error surfaces

The springback error surfaces calculated from the channel data set are shown in Figure 6-22. Note that the major difference between the springback error surfaces and the general shape error surfaces is that the shape error function uses only one desired shape, whereas the springback error surfaces use a different desired ("springback free") channel for each parameter combination.

Once again there is a general trend that an increase in the blank holder force causes a decrease in the springback error similar to the general shape error function in section 6.6.3. This trend, however, is not as strong as the general shape error plots
Figure 6-22: Springback error surfaces where the springback error is plotted against varying tool gap and die radii while the blank holder force is held at a particular level.
Table 6.4: Effects of the process parameters on the manually measured values of springback (flange angle and length springback error). Effect = $\frac{\text{Contrast}}{n^2}$ and Sum of Squares (Significance) = $\frac{(\text{Contrast})^2}{n^2}$, where $n =$ number of replicates (5) and Contrast = sum of the positive treatments minus the sum of the negative treatments.

<table>
<thead>
<tr>
<th>Left Flange Angle ($\beta$)</th>
<th>Right Flange Angle ($\beta$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Parameter</td>
<td>Effect on Error</td>
</tr>
<tr>
<td>BHF</td>
<td>-10.2</td>
</tr>
<tr>
<td>DR</td>
<td>-1.3</td>
</tr>
<tr>
<td>TG</td>
<td>-0.3</td>
</tr>
<tr>
<td>BHF &amp; DR</td>
<td>1.5</td>
</tr>
<tr>
<td>BHF &amp; TG</td>
<td>-0.6</td>
</tr>
<tr>
<td>DR &amp; TG</td>
<td>-1.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Left Flange Length</th>
<th>Right Flange Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Parameter</td>
<td>Effect on Error</td>
</tr>
<tr>
<td>BHF</td>
<td>-10.3</td>
</tr>
<tr>
<td>DR</td>
<td>-2.7</td>
</tr>
<tr>
<td>TG</td>
<td>1.8</td>
</tr>
<tr>
<td>BHF &amp; DR</td>
<td>-0.3</td>
</tr>
<tr>
<td>BHF &amp; TG</td>
<td>-1.7</td>
</tr>
<tr>
<td>DR &amp; TG</td>
<td>-0.3</td>
</tr>
</tbody>
</table>
because a reasonable amount of overlap exists between the springback error ranges in Figures 6-22(a) and 6-22(b). Moreover, the blank holder force trends in springback error agree with the manual measurements. The trends from the manually collected traditional measurements are tabulated in Table 6.4. In addition, there is also a minor trend that increasing the die radii decreases the springback error as seen in Figures 6-22(a) and (b). This is also consistent with the manual springback trends calculated for this data set using traditional measurements of flange angle ($\beta$). The tool gap parameter does not show any consistent trend when viewing Figure 6-22. This also accords with the manual springback trends.

Figures 6-23, 6-24, and 6-25 respectively show the comparison of representative low, medium, and high blank holder force channels against their “springback” free equivalent channels.

6.6.5 Evaluation of process noise

One of the problems facing manufacturers is the ability to maintain their production within a geometric dimensional tolerance window over a period of time. One application of the shape error function is to conduct a time series analysis of the formed product’s shape error. The shape error function provides a quantitative value that can track the quality of the product being manufactured over time. The trends in this data can be analysed to obtain the optimal time to tune the manufacturing process so to keep the process within the tolerance window. For example, the shape error value could be used in a statistical process control of the process.

This section investigates the shape error analysis of the two sets of 20 experimental channels created for the parameter settings B1D1T1 and B2D2T2. There are two major graphs that are used investigate process noise: the histogram; and the time series error graph.

6.6.5.1 Histogram graph

The histogram gives an indication of the noise distribution of the end product. The smaller the shape error distribution implies a more precise process. These plots enable the process control engineers to determine what variations/noise exists for different combinations of the process parameters. They could then investigate various methods to decrease this variation for specific parameter combinations. The shape error histogram also gives an indication of whether the normal operating point of the process has changed. One can determine if the process needs re-tuning by plotting the shape error frequency histogram of the measured manufactured parts compared to a “desired” shape of the normal operating average part. Figure 6-26 shows a process where the normal operating point has not changed. The shape error histogram is one in which the frequency of the parts decreases as the shape error range increases. Figure 6-27 shows
Figure 6-23: Comparison of springback errors across average channels sets B1D1T3, B1D2T3, and B1D3T3. The channels with drawn with the solid line are the “springback free” channels and the other channels (dashed lines) are the average channel for the particular set. The springback $\bar{b}$ vector is the absolute difference between the $\bar{b}$ vectors for the “springback free” channel and the average actual channel.
Figure 6-24: Comparison of springback errors across average channels sets B2D1T2, B2D2T2, and B2D3T2.
Figure 6-25: Comparison of springback errors across average channels sets B3D1T1, B3D2T1, and B3D3T1.
Figure 6-26: This diagram shows the case when the normal operating condition has not changed. It assumed that the parts are distributed in a normal distribution. This means that the number of parts that are extremely dissimilar to the normal operating part are very few. "x" indicates the normal operating condition, "o" indicates the measured manufactured parts.

Figure 6-27: This diagram shows the case when the normal operating condition has changed. It assumed that the parts are distributed in a normal distribution. This means that as the shape error range is increased the frequency of manufactured parts increases at least once. "x" indicates the normal operating condition, "o" indicates the measured manufactured parts.
a process where the normal operating point has changed, and potentially the process may need re-tuning. In this case, the shape error histogram no longer monotonically decreases as the shape error range increases.

6.6.5.2 Time series error graph

The time series error graph plots shape error versus production order (experimental order in the case of this chapter). This graph shows if there is any plant (equipment) wear or process parameter drift occurring within the process. If wear or process parameter drift occurs the shape error of the product is expected to rise over time and this can be found by statistically examining the time series error graph for upward trends in shape error. This is done by applying linear regression to the shape error values versus experimental order. If the slope of the error is positive and within the appropriate level of confidence, typically 95%, then there exists some fatigue or wear within the system. This method assumes that the shape error values are normally distributed around the mean trend line.

6.6.5.3 B1D1T1 process noise

The 20 B1D1T1 channels and the associated \( \vec{b} \) vectors are shown in Figure 6-28. The solid line and dotted lines in Figure 6-28 are, respectively, the average channel shape for the MSVM’s B1D1T1 channel set and the 20 channels formed using B1D1T1 settings to analyse the noise in the system. The graph of the \( \vec{b} \) vectors indicates that there is a major amount of variability in the second and third modes, which range from \(-80\) to
Figure 6-29: (a) First two components of the $\vec{b}$ vectors for the B1D1T1 channels ("x" is the MSVM's average B1D1T1 channel); (b) the B1D1T1 shape error frequency histogram; (c) the B1D1T1 shape errors in experimental order.
Figure 6-30: The B2D2T2 channels and the associated $\vec{b}$ vectors. The solid line represents the MSVM's average B2D2T2 channel and the dotted lines are the 20 B2D2T2 channels to analyse process noise.

20 and $-20$ to $80$ respectively. The first two components of the 20 B1D1T1 channels are graphed together in Figure 6-29(a). The circles represent the components from the 20 channels and the "x" represents the first two components of the MSVM's average B1D1T1 channel. The shape error function is then applied to the 20 B1D1T1 channel set where the desired shape is the MSVM's average B1D1T1 channel. The resulting shape error is plotted as a histogram, see Figure 6-29(b).

6.6.5.4 B2D2T2 process noise

A second set of 20 channels was created with B2D2T2 settings to compare the noise of the forming process for different parameter settings. Again, the MSVM's average B2D2T2 channel and the 20 other channels are graphed in Figure 6-30. The B2D2T2 channels appear to have more tool gap variance than the B1D1T1 channels as there is visible blurring of the side walls, see Figure 6-30(a). The side wall variation is associated with the third and higher modes of variation and indeed, it is these modes which have most of the variation. In addition, modes 8 and 12 have relatively large variations away from the MSVM's average B2D2T2 channel, see Figure 6-30(b). The first two modes of the $\vec{b}$ vectors of the B2D2T2 noise analysis channels are plotted in Figure 6-31(a). They appear to be evenly spread around the MSVM's average B2D2T2 $\vec{b}$ vector, represented by an "x".
Figure 6-31: (a) First two components of the $\tilde{b}$ vectors for the B2D2T2 channels (“x” is the MSVM’s average B1D1T1 channel); (b) the B2D2T2 shape error frequency histogram; (c) the B2D2T2 shape errors in experimental order.
6.6.6 Results from the inverse shape variation-process parameter model

The channel data is first analysed using a linear cluster separation measure to indicate the general success of a linear classification. This simple measure is calculated by taking the trace of the between class variance matrix multiplied by the inverted within class variance matrix, see equation (6.2). A higher value of the linear measure implies that the set of data is more separable. The data is split into the three levels for each parameter and the linear measure estimates the potential that each parameter can be classified correctly to the appropriate parameter level.

\[
\text{Linear Measure} = \text{trace} \left( B(W)^{-1} \right),
\]

where the within class variance is defined by:

\[
W = \sum_{i=1}^{L} \text{Prob}_i \text{E} \left[ \left( \bar{X} - \bar{M}_i \right) \left( \bar{X} - \bar{M}_i \right)^T \right] \forall \bar{X} \in \omega_i
\]

and the between class variance is defined by:

\[
B = \sum_{i=1}^{L} \text{Prob}_i \left( \bar{M}_i - \bar{M}_{Total} \right) \left( \bar{M}_i - \bar{M}_{Total} \right)^T.
\]

In addition, \( \omega_i \) is the \( i \)th class, \( \text{Prob}_i \) is the a priori probability of \( \omega_i \), \( \bar{X} \) is a data vector from the set of classification data, \( \bar{M}_i \) is the mean of the \( i \)th class, \( \Sigma_i \) is the covariance matrix of the \( i \)th class and

\[
\bar{M}_{Total} = \sum_{i=1}^{N} \text{Prob}_i \bar{X}
\]

\[
= \sum_{i=1}^{L} \text{Prob}_i \bar{M}_i
\]

is the mean of the data set. Note that \( L \) is the total number of classes in the data set and \( N \) is the number of pieces of data in the classification set.

The resulting values from the measure are:

- Blank holder force 22.0238
- Die radii 20.0000
- Tool gap 6.2938

The tool gap parameter is therefore the most difficult to classify as was already noted in a previous section (Section 6.3.3). Several classifiers are then used to determine the original levels of the process set-up parameters from the shape metric data. A ten-fold
cross validation technique is used to measure the accuracy of the classifiers. The set of channels is randomised and split into ten groups. Each group is held out in turn while the model is trained on the remaining nine groups. The held out group is then used as a test set to measure the model's classification accuracy. The results from the models for each parameter can be seen in Table 6.5.

<table>
<thead>
<tr>
<th></th>
<th>Linear classifier</th>
<th>Quadratic classifier</th>
<th>ANN</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blank holder force</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classifier accuracy</td>
<td>70.77%</td>
<td>97.69%</td>
<td>94.62%</td>
</tr>
<tr>
<td>Confusion matrices</td>
<td>44 30 0</td>
<td>43 1 0</td>
<td>41 1 0</td>
</tr>
<tr>
<td></td>
<td>0 5 0</td>
<td>1 42 0</td>
<td>1 38 1</td>
</tr>
<tr>
<td></td>
<td>0 8 43</td>
<td>0 1 42</td>
<td>0 4 44</td>
</tr>
<tr>
<td><strong>Die radii</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classifier accuracy</td>
<td>69.23%</td>
<td>94.62%</td>
<td>93.85%</td>
</tr>
<tr>
<td>Confusion matrices</td>
<td>43 38 0</td>
<td>39 1 0</td>
<td>39 1 0</td>
</tr>
<tr>
<td></td>
<td>0 3 0</td>
<td>2 41 2</td>
<td>2 40 1</td>
</tr>
<tr>
<td></td>
<td>0 2 43</td>
<td>0 2 43</td>
<td>0 4 43</td>
</tr>
<tr>
<td><strong>Tool gap</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classifier accuracy</td>
<td>68.46%</td>
<td>70.00%</td>
<td>73.85%</td>
</tr>
<tr>
<td>Confusion matrices</td>
<td>43 8 0</td>
<td>37 8 0</td>
<td>37 5 0</td>
</tr>
<tr>
<td></td>
<td>0 5 2</td>
<td>4 24 12</td>
<td>5 26 10</td>
</tr>
<tr>
<td></td>
<td>1 30 41</td>
<td>2 12 31</td>
<td>1 13 33</td>
</tr>
</tbody>
</table>

### 6.7 Discussion

There are a number of issues raised from the results section with regard to: the general shape error results; the springback error results; the process noise results; and the inverse shape variation - process parameter model results. This section discusses these issues in that same order.

#### 6.7.1 Shape error function

The examination of the low blank holder force channels illustrates an issue that needs further explanation with regard to the shape error function. The flange error tends to dominate the errors in the corner radii. This occurs for two reasons. First, the flange length is the feature that varies the most when the manufacturing/forming process is varied. Second, the number of boundary points used to describe the flange is greater than that describing the corner radii. The first explanation is a valid reason why the flange length and angle dominate the shape error as this is the most significant variation and hence the most significant component of error of the forming process. However, the second reason depends on functional requirements of the shape error. If the quality
inspector is only concerned with the closeness of the shape with respect to the major source of variation, then the shape error function need not be changed. On the other hand, if the corners are of functional value then the boundary point positioning system has to be changed to place a higher concentration of nodes in the corner radii areas and thus artificially raise the significance of the corners. The higher number of points in the flange distorts the dominance of the variance in the flange.

The dominance of the first two modes is a very minor issue for low and medium blank holder force channel as the flanges are shorter and the other differences become more significant. Nevertheless, when the blank holder force is high, the differences between the channels are minimal and the small changes between the flange lengths create relatively large changes in shape error. For example, the dominance of the flange length is seen in the representative high blank holder force channels, with the high die radii channel having less error than the medium die radii channel because the flanges are more balanced, even though the corners are more rounded. As previously stated, this situation can be diminished, if necessary, by changing the point distribution system such that there is an increase in the number of points in the corner areas relative to the flanges. This effectively is a type of weighting that affects the end shape error result. Thus, the shape error function can be manipulated to concentrate on areas of interest but not in an automatic fashion.

### 6.7.2 Springback error function

The springback error function is perhaps more beneficial than the previous traditional methods of measuring springback because it produces only one global error measurement rather than separate measurements from corner angles and side wall curl. The springback error function is a fusion of particular springback errors that are scaled in significance with regard to what the forming process "determines" is important. Another advantage is that the springback error function methodology can be applied to other forming processes not just "U" channel forming, unlike the traditional "U" channel forming measures.

There are two main trends seen in the springback error surfaces:

- springback error decreases with increasing blank holder force;
- springback error decreases with increasing die radii.

These trends agree with the theory for springback and sidewall curl discussed in Kim and Thomson (1989). However, the lack of a consistent trend with regard to the tool gap is interesting. This does not agree with the theoretical and experimental results of Kim and Thomson (1989). Part of the problem may be the large amount of variation in the actual tool gap value which could lead to a large variation in the final channel shape. The trends from the manually calculated effects using traditional springback
error measures and the visual evidence from the springback error surfaces indicate that the tool gap has a very minor affect on springback within our channel data set.

### 6.7.3 Evaluation of process noise

This section analyses the results obtained from the process noise channels data sets.

#### 6.7.3.1 B1D1T1 process noise channels

The decreasing frequency of the B1D1T1 channels as the shape error range is increased, shown in Figure 6-29(b), suggests the average of the B1D1T1 noise analysis samples has not changed from the MSVM's B1D1T1 channels. This assumes that noise distribution is a “bell” curve around the average channel and the frequency of channels will decrease as the shape error range increases away from the average channel.

The time series graph for the B1D1T1 channels, when inspected visually, appears to have increasing shape error peaks. To prove that there is parameter drift, a linear regression must be performed on the shape error data and then a $t$ test conducted on the regression variable. A linear regression performed on the shape error values gave the following linear model:

$$\text{Shape Error} = 12.06 + 0.17 \bar{X}_{\text{experimental order}}$$

where $X_{\text{experimental order}}$ is the vector $[1, 2, \ldots, 20]^T$. Two hypotheses can be asserted:

$$H_0 : \beta = 0;$$

$$H_1 : \beta \neq 0,$$

where $\beta$ is the regression variable. $H_0$ is rejected, at 5% significance level, if the following holds true:

$$\left| \frac{b}{s/\sqrt{\sum_i(X(i))^2}} \right| > t_{0.025} \sim t(n-2), \quad (6.3)$$

where $b$ is the estimated regression variable, $s$ is the estimated standard deviation of the residuals, and $t_{0.025}$ is the $t$ distribution with $(n-2)$ degrees of freedom at a probability of 0.025.

The statistic, equation (6.3), was calculated to be 1.42 and $t_{0.025} = 2.101$ for 18 degrees of freedom. Therefore, $H_0$ is accepted and it is concluded that there is no positive trend in the shape error with a 95% confidence level. Note that $t_{0.1} = 1.330$, which implies that there is at least some trend in the shape error, but not enough for it to be significant.

#### 6.7.3.2 B2D2T2 process noise channels

The “bell” shape of the histogram for the B2D2T2 channels, see Figure 6-31(b), implies that the shape of the average channel may have changed between the noise set-up and
MSVM set-up. This is because there are more channels between the shape error range 10 to 20 than from 0 to 10. The procedure of setting up the different tool gap levels as discussed in Section 6.3 noted that the medium tool gap channels may have more shape error due to increased alignment problems. Possibly the tool gap is slightly different between the two data sets which is causing the “bell” shaped histogram.

The B2D2T2 time series error graph visually does not show any apparent trends, see Figure 6-31(c). The linear regression model is:

\[ \text{Shape Error} = 14.35 + 0.04 X_{\text{experimental order}}, \]

where \( X_{\text{experimental order}} \) is the vector \([1, 2, \ldots, 20]^T\). By applying the same t test on the linear regression model of the time series shape error the following value is found for equation (6.3), 0.3852, note that, \( t_{0.025} = 2.101 \). Therefore, parameter drift is not a factor in the variations that are occurring in the B2D2T2 channels.

### 6.7.4 Inverse shape variation–process parameter model

The linear classifier appears to be able to classify the channels into the high and low parameter levels for all parameters with almost 100% accuracy, see Table 6.5. However, the medium level obviously is too close to either the low or high parameter levels. The blank holder force and die radii parameters appear to have data clusters that are close together for low and medium levels. Intuitively this is correct for blank holder force as the higher forces have much greater impact on the end shape particularly as the blank is very near to necking and tearing. This is supported by Figure 6-17(a) that shows a greater disparity between high and medium levels than medium and low levels with respect to blank holder force.

The confusion that occurs with the tool gap data is between the medium and high levels. This is primarily due to the errors in the tool gap set-up. The medium tool gap used three shims to create the gap whereas high tool gap used only two shims. There is greater potential for errors when aligning the three shims. Moreover, this error will always cause an increase of the actual tool gap with respect to the desired level. This has also caused a consistent problem for the other classifiers when discriminating between high and medium levels. Until better experimental procedures are implemented this error will be difficult to reduce when classifying.

The quadratic classifier assumes that the variances of the different classes/levels are not the same and therefore scales each class by its variance before making a classification decision. This appears to have removed many of the problems that the linear classifier had when classifying the medium level classes for each parameter, see Table 6.5. The accuracies of the quadratic classifier for both the blank holder force and die radii are extremely high. The accuracy in predicting the tool gap, however, is not significantly better than that of the linear classifier. The quadratic classifier’s confusion matrix
is very different, with respect to the linear case, because there is much less accuracy in predicting the high and low levels, but the medium level is predicted with a much greater accuracy.

The ANN classifier is more inconsistent than the other classifiers. Sometimes it classifies better than the quadratic classifier but typically it is slightly worse, see Table 6.5. The reason for this is probably due to the classes having a lot of overlap within the shape variation modes which may confuse the ANN's learning. This is also supported by the very minimal number of neurons in the best and most consistent ANN classifiers, see Table 6.2. The one hidden layer and one neuron in each layer enables the ANN to learn the basic trends of the data without learning the detail. The more complex ANNs are perhaps "over trained" and learn the specific patterns of the training data and are not able to abstract to the testing data.

Again the tool gap parameter is the hardest to classify (73.85%). However, the ANN does classify slightly more accurately than the quadratic classifier but not significantly better. Similarly, the medium and high tool gap channels are hard to discriminate accurately as has been previously mentioned.

6.8 Summary

In summary, there were four main contributions of this chapter. The first contribution was to apply the proposed shape error function to sheet metal forming. The shape error function was applied to a set of channels produced from a "U" bend channel test used to measure the amount of springback for a particular material. The shape error function was in strong agreement with manual measurements of error for low and medium blank holder force. At high blank holder force there were only minor differences between channels, and the shape error function separated these channels by the forming process' most significant error (flange length). Furthermore, the shape error surfaces indicated that the blank holder force had a strong influence on the resulting shape error, whereas the two other parameters, the die radii and the tool gap, had an inconsistent influence on the shape error.

The second contribution was to extend the shape error function to measure springback error in the channels. The springback error surfaces were influenced by the blank holder force, but there also was a minor trend for the springback error to decrease as the die radii was increased. These trends are consistent with the theory of bending under tension for side wall curl (Kim & Thomson, 1989). The trends with respect to tool gap were insignificant which is inconsistent with the literature. However, the springback error function results did correspond with the manually measured results from the data set which may imply that the experimental set-up may have to be changed to obtain better tool gap data.

The third contribution was the application of the shape error function to channel
data to extract further information about the sheet metal forming process. The inherent variability present in the stamping process results in a corresponding variation in the geometric shape of the part produced. From a quality perspective, the error between the desired shape and the actual shape produced needs to be monitored and controlled. This is particularly relevant for process engineers who need a basis to describe the decline in the geometric quality of the product. The histogram of the shape error and the time series error graphs provide enough quality information to conduct statistical quality control methods.

The final contribution of this chapter was to create inverse shape variation - process parameter models. The inverse models were created using linear, quadratic and ANN classifiers that related the $\vec{b}$ vectors to the process parameter levels. The quadratic classifiers were the most accurate predictors as well as being the most consistent classifier over all the parameters with average accuracies of: Blank holder force - 97.69%; Die radii - 94.62%; Tool gap - 70.00%. The ANN classifier was inconsistent but its typical accuracies were below the quadratic classifier: Blank holder force - 94.62%; Die radii - 93.85%; Tool gap - 73.85%. The linear classifier confused the medium level parameter consistently with either the low level, for blank holder force and die radii, or the high level, for tool gap. Consequently, the linear classifier was the most inaccurate classifier at approximately 70.00% for all parameters. The nature of the data was such that it suited the quadratic classifier over the ANN. The ANN did slightly better on the more complex tool gap data, however it was not a significant improvement.

Given the similar classification accuracies from analysing forging data (see Chapter 5) it may be generalised that the shape manufacturing feedback model will provide reasonable classification data in most cases for other shape manufacturing processes.
Chapter 7

Simulated sheet metal forming

Shape Odyssey 2001 continues

7.1 Introduction

This chapter investigates applying the shape manufacturing feedback model to simulated sheet metal forming data. The sheet metal forming process of the previous chapter has been modelled using a finite element simulation. The resulting simulated channels are examined for shape error by using a shape error function. This function is an extension of the previous chapter’s shape error function. It is important to note that the aim of this chapter is not to model springback in minute detail, but to determine whether the shape manufacturing feedback model is appropriate to measure the accuracy of simulated processes.

There is a large scope for using inspection ideas and methods for measuring the accuracy of finite element models. Currently, finite element models are validated against a benchmark problem. The benchmark problem is a well constructed experimental problem with accurate experimental results that the finite element models can be compared against. There are two main methods of comparing simulated parts versus experimental parts, by using either shape, or strain information. This chapter will only consider the comparison of shape. In general, the shape of a simulated part is plotted against
the actual part for the reader to visually examine the differences. A quantitative measure of the global error between the simulation and the actual part, to this author’s knowledge, has not been implemented. There are, however, quantitative measures that have been developed to calculate the error for only a particular region of the part. For example, the local shape error measures for “U” channel forming simulations are the same as those devised for the experimental channels (see Figure 6-6). That is, the bend angles of the upper and lower corners are measured, as well as the side wall radius of curvature. The main contribution of this chapter is, therefore, to provide a new global shape measure for comparing simulations to experimental data using the shape manufacturing feedback model. In particular, a global measure of shape error is developed for “U” channel simulations.

Past simulation benchmarks have only specified a single set-up state of a process, where each process parameter is fixed at a specified level. This was mainly due to the computational time involved in determining a simulation result. Computers are, however, becoming increasingly powerful each year and additional memory is relatively inexpensive. An improvement that could be made to the current benchmarking method is to create a benchmark in which finite element modellers have to simulate a number of set-up states for the one process. This would determine how flexible each finite element model was to changes in the set-up and whether the code was worthwhile for the process as a whole. Given that the shape manufacturing feedback model is based upon knowing the outcomes of a process under varying conditions, it would be natural to apply the shape manufacturing feedback model to this new benchmark method to give quantitative results on the accuracy of the simulations.

The remainder of this chapter is structured as follows. First, a review of the simulation of springback with regard to the “U” bend channel forming process is undertaken in Section 7.2. The set-up of the simulation used in this chapter is explained in Section 7.3. The resulting simulation channels from the simulation are aligned using the methods outlined in Section 7.4. Section 7.5 outlines the set-up of the simulation shape error function. The shape variation of the simulated channels are then compared to that of the experimental channels in Section 7.6. The global shape error between the simulated and the experimental channels is discussed in Section 7.7.

7.2 Simulation of springback

Simulations of sheet metal forming have been a serious area of research since the 1980s. By the late 1980s, Rebolo et al. (1989) had developed a 3-D sheet metal forming simulation that excluded springback effects. The simulation of springback, however, has remained a difficult problem, and current finite element modellers are still struggling to find an accurate simulation method for the prediction of springback (Iwata et al., 2001).
One of the first serious attempts in predicting springback by finite element simulation was Karafillis and Boyce's (1992) seminal paper on predicting springback in a “U” channel. This was followed by the NUMISHEET'93 (Numerical simulation of 3-D sheet metal forming processes) conference which set the prediction of springback in “U” channel as one of the conference’s benchmark problems. The NUMISHEET'96 conference extended this benchmark problem to three dimensions by proposing a new springback benchmark problem of forming a 3-D “S” rail. From the NUMISHEET conferences and other springback literature there are two main issues that the finite element modellers have investigated with respect to springback. The first issue is the type of finite element method used to model springback. The second issue is what to include in the underlying material models used within the simulations.

7.2.1 Finite element methods

There are two main finite element methods that researchers have used to model springback, namely static implicit and dynamic explicit simulations. The theory behind the two methods is outlined in Sections 7.2.1.1 and 7.2.1.2, and for more detailed examination, Lee (1999) presents a clear review of the two methods. Initially the static implicit techniques were used to model the general sheet metal forming process to great success in 2-D (Rebelo et al., 1992) in the 1980s. In the beginning of the 1990s, however, it was found that the static implicit method required a large amount of memory and computational time. The computational power necessary at that time was unfeasible for complex 3-D problems. The dynamic explicit method was then applied to sheet metal forming which decreased the computational time but with a slight loss in accuracy (Lee et al., 1999).

The simulation of springback in sheet metal forming was split into two components: the forming of the initial shape (forming); and the removal of the punch and die to give springback (unloading). The dynamic explicit method, however, does not adequately handle the second component (unloading) because it is a static problem, whereas the first component (forming) is a dynamic problem. Karafillis and Boyce (1996) proposed combining the static implicit and dynamic explicit methods together, where the dynamic explicit method was used to form the sheet metal part (forming) and the static implicit method was used to unload the tool set to simulate springback. Currently, this combined method has remained the main approach finite element modellers have used to solve the springback problem.

It was also found that not only does the type of finite element method affect the end result, but also the parameters chosen within each simulation. Mattiason et al. (1995) showed that the numerical parameters of the finite element simulation, such as element size and punch velocity, influenced the final springback result. This result revealed the lack of reliability and stability in the simulation of springback. Lee and Yang (1998)
also conducted an investigation into numerical parameters influencing springback simulation. They performed a fractional factorial experiment using five numerical parameters: contact damping parameter, penalty parameter, blank element size, and number of corner elements. Their experiments indicated that blank element size and number of corner elements had significant effects on the springback simulations.

### 7.2.1.1 Static implicit simulations

Implicit analysis updates the displacement of the mesh nodes according to a Newton iterative search of the statics equation in the following standard form:

\[
\Delta \tilde{u}^{(i+1)} = \Delta \tilde{u}^{(i)} + K_t^{-1} \cdot \left( \tilde{F}^{(i)} - \tilde{f}^{(i)} \right),
\]

where \( K_t \) is the current tangent stiffness matrix, \( \tilde{F} \) is the applied load vector and \( \tilde{f} \) is the internal force vector.

When there is a large amount of contact and bulk material transformation, such as in sheet metal forming, then the implicit simulations often take a long time to run due to convergence problems. This is partly due to the difficulty of inverting the stiffness matrix.

### 7.2.1.2 Dynamic explicit simulations

The explicit analysis updates the displacement of the mesh nodes according to a dynamic equation using the explicit central difference integration rule:

\[
\tilde{u}^{(i+1)} = \tilde{u}^{(i)} + \Delta t^{(i+1)} \cdot \tilde{a}^{(i+\frac{1}{2})},
\]

\[
\tilde{a}^{(i+\frac{1}{2})} = \tilde{a}^{(i-\frac{1}{2})} + \frac{\Delta t^{(i+1)} + \Delta t^{(i)}}{2} \cdot \tilde{u}^{(i)}.
\]

where \( \tilde{a}^{(i-\frac{1}{2})} \) is known from the previous time increment and the acceleration, \( \tilde{a}^{(i)} \), is calculated by,

\[
\tilde{a}^{(i)} = M^{-1} \cdot \left( \tilde{F}^{(i)} - \tilde{f}^{(i)} \right).
\]

The matrix \( M \) is the mass matrix which is made diagonal to ease its inversion by assuming the mass is "lumped" together at the nodes without much loss in generality.

The major problem of explicit codes is that the stable time increments are very small compared to the actual forming time. To compensate for this problem, the speed of the process is increased by increasing the die velocities. There is also the problem of inertia effects which can upset the stability of the explicit solution. The difficulty of simulating springback using the explicit code is because it is a static problem and the explicit code takes a long time to dampen its vibration of the unloaded part when finding equilibrium.
7.2.2 Material models

A brief introduction to material models is outlined in this section followed by its relevance to finite element modelling.

7.2.2.1 Sheet metal forming material models

There are many phenomena which affect the material models of sheet metal, such as recrystallisation, material aging and damage, and springback. However, we will only consider strain hardening in this section to show the complexity of the problem.

The simple explanation for strain hardening is that it is the increase in stress needed to produce further strain in the plastic region (Ashby & Jones, 1991). The plastic region is where some of the applied strain becomes a permanent deformation. Strain hardening is believed to occur from trapped dislocations in the material not being able to flow when an external stress is being applied. Strain hardening can be removed by annealing the material.

In general, the flow stress of the material is a function of the strain of the material,

$$\sigma_f = f(\epsilon),$$

(7.1)

where $\sigma_f$ is the flow stress of the material, and $\epsilon$ is the strain of the material. A typical stress-strain curve is shown in Figure 7-1. The exact shape of the stress-strain curve often relies on the type of material. If one assumes the material is isotropic, responds to stress similarly in any direction and yielding is independent of the orientation of the material, then several approximations of the stress-strain curve can be made (Marciniak & Duncan, 1992). Most of these approximations are non-linear in nature.

**Power law**: $\sigma_f = K\epsilon^n$, this approximation does not have the sharp initial point as the true curve. It may also overestimate the actual curve for large strains.
7.2.2.2 Finite element modelling and material models

The second major issue in springback simulation is the nature of the underlying material models used in the finite element model. Karafillis and Boyce (1992) first used an isotropic elastic-plastic material model with isotropic strain hardening. Shu and Hung's (1996) paper on simulating springback in the double bend process concluded that their simulations would not improve until the Bauschinger effect was included in the material models. The Bauschinger effect (Hill, 1967) is where a pre-strained material has a reduced yield stress in the opposite direction of the previous loading. Shi and Zhang (1999) investigated the effects of varying material models and parameters on springback. They concluded that springback was influenced by: the yield criterion of the material; the anisotropy of the material; and the strain hardening behaviour (Bauschinger effect). Finally, Iwata et al. (2001) obtained good results from including both the Bauschinger effect and the decrease in the pseudo-elastic modulus in their material models. The pseudo-elastic modulus is the momentary tangential slope of a material at the beginning of unloading. They conducted experiments to determine the material behaviour before doing the simulations. Figure 6-2 shows the effects of the Bauschinger effect and the decrease in the pseudo-elastic modulus.

![Figure 7-2: Diagrams showing the Power-law, Pre-strained material, and Linear strain hardening approximations of the stress-strain curve.](image)

(a) Power law: \( \sigma_f = Ke^n \)  
(b) Pre-strained material:  
\[
\sigma_f = K(e_0 + e)^n
\]  
(c) Linear strain hardening: \( \sigma_f = Y_0 + P\epsilon \)

**Pre-strained material**: \( \sigma_f = K(e_0 + e)^n \), this approximation assumes the material has been pre-strained so as to ignore the elastic region of the material.

**Linear strain hardening**: \( \sigma_f = Y_0 + P\epsilon \), this approximation assumes a linear relationship between stress and strain and should only be used when considering small changes in strain.
Figure 7-3: This diagram shows the G3N steel stress-strain curve that is used within the channel forming simulation.

7.3 Set-up of channel forming simulation

This section describes the set-up of the ABAQUS\(^1\) simulation to model the channel forming process of the previous chapter. The simulation set-up is first described in Section 7.3.1 and then the design of simulations which were conducted is outlined in Section 7.3.2.

7.3.1 Simulation set-up

The simulation of the channel forming experiment is based upon Taylor's \textit{et al.} (1993) ABAQUS simulation for the NUMISHEET'93 Conference springback benchmark. Our simulation aims to model the experimental channel forming set-up in Chapter 6. The material is modelled as an elastic-plastic material with isotropic elasticity and uses the Hill anisotropic yield criterion for plasticity. The Bauschinger effect and the decrease in the pseudo-elastic modulus were not included in simulation model. The importance of these effects were only presented in the literature in this past year after our simulation models and simulated channels had been created. Moreover, the main investigation of this chapter was not to create perfect simulations of springback, but to determine whether the shape manufacturing feedback model could provide appropriate feedback about the simulation of springback.

The blank is 150mm by 15mm and is 0.76mm thick. The blank material model is based on G3N steel using tensile test data from Herron and Lanzon (1997). The stress-strain curve that is used within the channel forming simulation can be seen in

---

\(^1\)ABAQUS is a standard finite element analysis package (ABAQUS, 1998).
Figure 7-3. The additional properties of the blank material are:

- Young’s modulus = 200 GPa;
- Poisson’s ratio = 0.3;
- Density = 7800;
- Yield stress ≈ 150 MPa;
- Anisotropic yield criterion:
  \[\begin{align*}
  R_{11} &= 1.0, & R_{22} &= 1.245, & R_{33} &= 1.0, \\
  R_{12} &= 1.0, & R_{13} &= 0.9456, & R_{23} &= 1.0.
  \end{align*}\]

Note that direction 1 is along the length of the blank, direction 2 is in the opposite direction of the punch, and direction 3 is in the plane of the blank.

The blank is made up of 150 first order shell elements with the assumption that the forming process undergoes plane strain at the element level. The forming process is symmetric about the centre of the punch and therefore only half the forming process was modelled. The coefficient of friction is set at \(\mu = 0.1\) for all contact surfaces. This value is within the range given by Schey (1987) for sheet metal working. The coulomb model of friction is used as it is the default option and was used by Taylor’s et al. (1993) benchmark.

The channel forming process is simulated using two main steps: step 1, ABAQUS/Explicit; step 2, ABAQUS/Standard (implicit).

**Step 1: ABAQUS/Explicit simulation**

The ABAQUS/Explicit simulation is carried out in two sub-steps. The first step applies the blank holder force of the prescribed level to the blank. The force is ramped during this step to minimise inertia effects. The second step moves the punch down 32 mm by applying a triangular amplitude function for the punch velocity. The triangular amplitude function started and ended with a velocity of zero and peaked at the middle time period with a velocity of \(-15\) m/sec, see Figure 7-4.

![Figure 7-4: This diagram shows the velocity of the punch used during the channel forming ABAQUS/Explicit simulation.](image-url)
Step 2: ABAQUS/Standard simulation

ABAQUS/Standard (implicit) simulation is used to model the springback in the channel after the punch is removed. The ABAQUS/Standard simulation imports the formed channel from the previous explicit step. The explicit step stressed channel is gradually brought to a quasi-equilibrium state in a single ABAQUS/standard step. At the end of the step, the calculated displacement is the springback and the remaining stresses give the residual stress state.

7.3.2 Design of experiment

The design of the simulation experiment is exactly the same as the design of experiment for experimental channel forming (see Section 6.3.2) so that a comparison can be made between the two manufacturing systems. A three-level, three-factor factorial experiment is performed by varying the blank holder force, die radii, and tool gap parameters. The range of values for these parameters can be seen in Table 6.1 in Chapter 6.

The blank holder force is varied by modifying the amount of force that the virtual blank holder applies to the blank. The die radii and tool gap are varied by redrawing the simulated die-set such that the die radii and tool gap are set at a particular level.

7.4 Channel alignment and point placement

After the simulations are completed, the resulting channel data is aligned and points are assigned to the boundary of the channels. The nodes of each deformed blank are used as a description of the boundary of the simulated channel. Each simulated channel's boundary is combined with the mirror image of the right flange to create the whole channel because only half the process was modelled due to symmetry. A fixed number of boundary points is then distributed over the simulated channel according to the method specified in Section 6.4.1.3. The simulated channels are then processed in two ways:

- first, the simulated channels are aligned to each other to create a shape variation training set to develop a MSVM for the simulated channel forming process; this MSVM is used for investigating shape variation within the simulation process; and
- second, the same simulated channels are aligned to the average channel of the experimental channel data set from Chapter 6 to create a second set of aligned simulated channels that can be compared to the experimental results of Chapter 6.

Note, the alignment process of the simulated channels is the same as that used in Section 6.4.2.
7.5 Setting up the simulation shape error model

This section outlines a method of describing the global shape error between an actual experimental channel and its corresponding simulated channel. The actual experimental channel is considered, in this case, to be the “desired” channel. The shape error function calculates the total shape error in a simulated channel in terms of the shape variations of the actual process. This is the main contribution of this chapter.

The simulations are first tuned to match the actual experimental result of the modelled process. This is achieved by reducing the shape error and other material conditions (if known) by tuning internal variables within the simulation. The simulation shape error function is then used to measure the difference in shape. This error function is similar to the springback error function developed in Section 6.5.2 (see equation (6.1) on page 138), where the average experimental channel and the corresponding simulated channel replace the “springback free” channel and the manufactured channel, respectively. The simulation shape error is calculated as follows:

\[
\text{Simulation Error} = \sum_{i=1}^{n} \frac{1}{\lambda_i} | \tilde{b}(1)_{\text{simulated channel}} - \tilde{b}(1)_{\text{experimental channel}} | + \sum_{i=1}^{n} \frac{1}{\lambda_i} | \tilde{b}(2)_{\text{simulated channel}} - \tilde{b}(2)_{\text{experimental channel}} | + \ldots ,
\]

(7.2)

where the experimental channel has the same corresponding parameter values as the simulated channel. Note that the \( \tilde{b} \) vectors are calculated using the experimental channels’ MSVM and \( \lambda_i \) is the \( i^{th} \) eigenvalue of the experimental channel’s covariance matrix, \( S \).

The advantage of this shape error function is that the error is calculated with regard to the variations of the actual process. That is, the shape error is weighted to find the significant variations in shape that are different from the actual process. This global simulation shape error function can also be abstracted to any simulation process if the appropriate experimental data is available. The \( \tilde{b} \) vector of the simulated shape is compared to the \( \tilde{b} \) vector of the corresponding average experimental shape in the same way as the simulated channels in equation (7.2).

Thus, the simulated shape error is calculated using the following steps:

Step 1: align the simulated channels to the average channel of the experimental data set.

Step 2: calculate the \( \tilde{b} \) vector for each simulated shape using the MSVM of the experimental channel data set.

Step 3: determine the weighted absolute difference between each simu-
lated channel’s \( \vec{b} \) vector and the corresponding experimental channel’s \( \vec{b} \) vector. In our case the experimental channel’s \( \vec{b} \) vector was the average \( \vec{b} \) vector for that parameter combination because there were five channels for each combination.

### 7.6 Comparison of simulation and experimental shape variation

The results for this chapter are based on the set of 27 simulated channels (\( 3^3 \) process parameter combinations). This section examines the resulting shape variation trends from the set of simulated channels. A simulated channel MSVM was also created from the set of simulated channels. The shape variation trends from the simulated channel MSVM can then be compared to the experimental channels’ MSVM trends that were examined in the previous chapter. If there is good correspondence between the simulated and actual processes, then the shape variation trends should be similar. Otherwise, the simulation is not an accurate model of the actual process.

#### 7.6.1 Comparison of the variational modes

The over-emphasised first three modes of geometric variation of the simulated channels are shown in Figure 7-5. The most significant mode of the simulated channels emphasises the length of the flanges. The second mode tracks the side wall curl of the simulated channel. The third mode is quite insignificant (weighting factor = 0.0023) and appears to track noise and possibly some side wall curl.

The MSVM of the simulated channel data set can now be compared to the MSVM of the experimental channel data set (Figure 6-16) by examining the most significant modes of shape variation. The experimental channel shape variation modes have been analysed previously in Section 6.6.1 and it was noted that the most significant mode tracked the flange length and angle of the upper corner. The next significant mode was process dependent where the imbalance between the flanges was tracked. The third significant modes tracked the side wall curl of the channels. We will assume that the experimental variation modes are the true modes of variation. The comparison between the simulation and experimental modes (Figures 6-16 and 7-5) realised the following differences:

- the most significant mode for the simulated channels does not include any upper corner angle variation;

- there is no simulated channel mode that tracks the imbalance of the flange; and
Figure 7-5: The grossly emphasised PDM variational modes of the simulated channel data set. Note that the positive perturbation is the dotted line and the negative perturbation is the dashed line and the significance of each mode is reflected in the weighting factor below each graph.
• the amount of side wall curl is reduced in the simulated channels as
  the weighting of the second mode is 0.0409, whereas the equivalent
  experimental mode (third mode) has a weighting of 0.1185.

It should be noted that the imbalance in the simulated channels' flanges would have
to be specifically computer modelled as an imbalance in blank holder force or different
friction conditions between the two die blocks in the simulations for it to appear as a
shape variation. Due to time commitments, this extra modelling was not performed.

7.6.2 Comparing only the simulated channels

The shape variation of the simulated channels can also be seen by viewing the simulated
channels themselves. Figure 7-6 shows the difference between the two combinations of
low and high blank holder force channels and medium and high blank holder force
channels. Figure 7-6(a) reveals that the low and high blank holder force channels are
very close in terms of shape. The differences are that the low blank holder force channel
has:

• slightly more side wall curl;
• the upper corner flange angle is greater; and
• the flange length is longer.

Figure 7-6(b) indicates a large difference between the medium and high blank holder
force channels, which is much larger than the difference between low and high blank
holder force. The medium blank holder force channel has the following differences:

• more side wall curl;
• much longer flanges; and
• possibly the upper corner flange angle is smaller.

The major problem with the simulations is that the low and high blank holder
force channels appear to be very similar. This is a problem because the experimental
channels from the previous chapter indicated that the low blank holder force channels
had the most springback; and they were the channels that had the most differences
to the high blank holder force channels. This problem with the simulations can be
seen specifically when comparing the flange lengths. The flange lengths of the low
blank holder force simulation channels were longer than the flange lengths of the high
blank holder force channels. This shape variation is incorrect when compared to the
flange length variation in the experimental channels. The problem is only magnified
when the medium blank holder force channels are examined. This incorrect flange
stretching in the simulations implies that the model of the die block / blank / blank
holder interactions must be reviewed in both the explicit and implicit pieces of code. This problem was not investigated due to time commitments. If one is to attempt to solve this problem, then there are two initial approaches that can be made. First, the finite element parameters, such as, number of elements in the channel, can be varied to try and obtain a better simulation. Second, the material model parameters can be investigated to make sure they are correct. Otherwise, the underlying material models may be incorrect and will have to be reviewed.

The problem with flange length error is not only confined to our simulations. Joannic and Gehn (1999) simulation results for “U” channel forming also have a large difference in the flange length when compared to the experimental results. They were also interested in varying the blank holder force parameter to determine the accuracy of their method. Their focus, however, was to obtain greater accuracy of the lower corner angle and side wall curl rather than the shape of the whole channel. Nevertheless, not all simulations have problems with flange length error. Pourboghrat et al. (1998) appeared to get good agreement with the experimental results of aluminium channel forming. They did not, however, vary the blank holder force a great deal.

### 7.7 Shape error of the simulated channels

This section examines the shape error between the simulated channels and their corresponding experimental channels, where the shape error is calculated by the simulation shape error function. The difference between this section and the previous section is
that the shape variation in the experimental channels is used to explain shape errors in the simulated channels, rather than the shape variation of the simulated channels. The experimental channels' MSVM is applied to the simulation channel data set to assist developing the simulation shape error function. The simulation shape error function results are then analysed to determine the robustness of the simulation process when varying the process parameters.

The comparison of the experimental and the simulated channels are performed in two parts. First, Section 7.7.1 compares the experimental shape variational surfaces, viewed in the previous chapter within Section 6.6.2 (Figure 6-17, page 142), to those of the simulated process. Second, the shape error surfaces of the simulated channels are investigated in Section 7.7.2.

### 7.7.1 Simulation variation modes based on experimental data

The surface responses of the simulated channels' first, fourth and ninth $\vec{b}$ vector modes, calculated using the experimental channels' MSVM, are seen in Figure 7-7. The surface responses chosen for the above vector modes are the same as those chosen in Section 6.6.2 so that a comparison can be made.

The first mode, Figure 7-7(a), appears to be dominated by the blank holder force. That is, the surface response does not change significantly when the tool gap parameter is varied. The $\vec{b}(1)$ values for low and high blank holder force are similar and this agrees with the results from the previous section (Section 7.6). The medium blank holder force channels have extremely high $\vec{b}(1)$ values. The fourth mode, Figure 7-7(b), is also dominated by the blank holder force, but this time against the die radii parameter. The medium blank holder force channels have very large negative $\vec{b}(4)$ values. The ninth mode, Figure 7-7(c), has equal influence from both tool gap and blank holder force as can be seen with the diagonal contour lines below the response surface.

#### 7.7.1.1 Differences in shape variation trends

This section discusses the differences in the shape variation trends between the simulated and the experimental channels by comparing Figures 6-17 and 7-7.

The differences in the shape variation trends for the first mode, $\vec{b}(1)$, are as follows:

- there are no negative first mode values for the low blank holder force simulated channels;
- the medium blank holder force simulated channels $\vec{b}(1)$ values are extraordinarily high (simulated $\approx 700$ vs experimental $\approx 0$);
- simulation $\vec{b}(1)$ surface does not consistently increase with blank holder force.
Figure 7-7: The surface responses of the first, fourth, and ninth $\vec{b}$ vector modes for the simulated channels when using the experimental channels’ MSVM to create the simulated channels $\vec{b}$ vectors.
The differences in the shape variation trends for the fourth mode, $\delta(4)$, are as follows:

- the blank holder force parameter is dominant for the simulation channels, whereas the die radii parameter is dominant for experimental channels;
- the medium blank holder force values for the simulated channels are extraordinarily negative (simulated $\approx -200$ vs experimental, $-10$ to $30$)

The differences in the shape variation trends for the ninth mode, $\delta(9)$, are as follows:

- the slope of the surface is wrong, the $\delta(9)$ surface should increase when the blank holder force is increased. The simulated channels appear to show the opposite.

In summary, the low and medium blank holder force simulated channels have large shape variation trend differences with respect to their experimental counterparts. Indeed, the medium blank holder force simulated channels have extremely large differences. In general the simulation shape variation surface trends are incorrect when compared to the experimental surfaces. This would suggest that the finite element models used for this chapter have yet to capture the essential variation features of the channel forming process.

### 7.7.2 Shape error between the experimental and simulated channels

This section focuses on the results from applying the simulation shape error function to the simulated channel data set. This allows the comparison of the simulation channels against the experimental channels. Figure 7-8 presents the shape error surfaces, calculated using simulation shape error function (equation (7.2)), for all the combinations of the process parameter levels. Table 7.1 lists the manually measured lengths and angles of all the simulated channels. This table can be compared with Table 6.3 which contains all the manually measured lengths and angles of the experimental channels in the previous chapter. Note that the left and right flange lengths for the simulated channels are not the same as one would expect from symmetric channels. The algorithm that calculated the flange length used orthogonal regression in the same fashion as that used to find the mid-point of the channels in Section 6.4.1.1. This algorithm approached the channels from left to right and this is why the right flange length is shorter than the left. When working from from left to right, the left flange orthogonal line starts at the tip of the flange and progresses towards the upper left corner; whereas for the right flange orthogonal line, the algorithm starts from the right upper corner and progresses towards the tip of the right flange. The difference between the two is that the orthogonal line for the right flange starts from further down the right corner which makes for a flatter line and shorter estimated flange distance. There is also a
similar effect on the side wall corners, which also affects the calculation of the length of the right flange. The maximum error in the flange lengths, however, is only 0.2 mm and therefore is not a significant problem.

Figure 7-8: Simulation shape error surfaces where the shape error is plotted against varying tool gap and die radii while the blank holder force is held at a particular level. Shape error is the difference between the experimental and simulated channels according to their $\vec{b}$ vector representations where the shape variation modes are those from the experimental channel data set.

The simulation shape error surfaces do not show any consistent trends across all levels for any of the process parameters. The blank holder force parameter, however, has the most significant effect on the simulated channels. The simulated channels will, therefore, be examined by splitting the channels into their the blank holder force levels. The high blank holder force channels have, on average, the least amount of shape error and the medium blank holder force channels have the most shape error. The low blank
Table 7.1: The manually measured flange angles and lengths and floor angles for the simulated channels.

<table>
<thead>
<tr>
<th>Channel Set</th>
<th>Flange Angle, $\beta$</th>
<th>Flange Length</th>
<th>Floor Angle, $\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Left</td>
<td>Right</td>
<td>Left</td>
</tr>
<tr>
<td>B1D1T1</td>
<td>$92.9^\circ$</td>
<td>$93.6^\circ$</td>
<td>$34.0\text{mm}$</td>
</tr>
<tr>
<td>B1D1T2</td>
<td>$93.7^\circ$</td>
<td>$94.4^\circ$</td>
<td>$34.1\text{mm}$</td>
</tr>
<tr>
<td>B1D1T3</td>
<td>$95.6^\circ$</td>
<td>$95.7^\circ$</td>
<td>$33.7\text{mm}$</td>
</tr>
<tr>
<td>B1D2T1</td>
<td>$92.7^\circ$</td>
<td>$92.3^\circ$</td>
<td>$34.3\text{mm}$</td>
</tr>
<tr>
<td>B1D2T2</td>
<td>$92.6^\circ$</td>
<td>$92.4^\circ$</td>
<td>$33.8\text{mm}$</td>
</tr>
<tr>
<td>B1D2T3</td>
<td>$92.7^\circ$</td>
<td>$92.5^\circ$</td>
<td>$33.8\text{mm}$</td>
</tr>
<tr>
<td>B1D3T1</td>
<td>$91.4^\circ$</td>
<td>$91.0^\circ$</td>
<td>$33.8\text{mm}$</td>
</tr>
<tr>
<td>B1D3T2</td>
<td>$91.8^\circ$</td>
<td>$91.4^\circ$</td>
<td>$33.8\text{mm}$</td>
</tr>
<tr>
<td>B1D3T3</td>
<td>$93.3^\circ$</td>
<td>$92.8^\circ$</td>
<td>$33.5\text{mm}$</td>
</tr>
<tr>
<td>B2D1T1</td>
<td>$94.1^\circ$</td>
<td>$94.1^\circ$</td>
<td>$49.3\text{mm}$</td>
</tr>
<tr>
<td>B2D1T2</td>
<td>$93.4^\circ$</td>
<td>$93.4^\circ$</td>
<td>$48.9\text{mm}$</td>
</tr>
<tr>
<td>B2D1T3</td>
<td>$94.2^\circ$</td>
<td>$94.2^\circ$</td>
<td>$48.2\text{mm}$</td>
</tr>
<tr>
<td>B2D2T1</td>
<td>$94.0^\circ$</td>
<td>$94.0^\circ$</td>
<td>$49.0\text{mm}$</td>
</tr>
<tr>
<td>B2D2T2</td>
<td>$95.1^\circ$</td>
<td>$95.1^\circ$</td>
<td>$46.3\text{mm}$</td>
</tr>
<tr>
<td>B2D2T3</td>
<td>$93.1^\circ$</td>
<td>$92.8^\circ$</td>
<td>$45.8\text{mm}$</td>
</tr>
<tr>
<td>B2D3T1</td>
<td>$99.4^\circ$</td>
<td>$99.4^\circ$</td>
<td>$47.6\text{mm}$</td>
</tr>
<tr>
<td>B2D3T2</td>
<td>$93.9^\circ$</td>
<td>$93.8^\circ$</td>
<td>$48.3\text{mm}$</td>
</tr>
<tr>
<td>B2D3T3</td>
<td>$93.3^\circ$</td>
<td>$93.3^\circ$</td>
<td>$45.1\text{mm}$</td>
</tr>
<tr>
<td>B3D1T1</td>
<td>$90.8^\circ$</td>
<td>$90.7^\circ$</td>
<td>$32.3\text{mm}$</td>
</tr>
<tr>
<td>B3D1T2</td>
<td>$89.0^\circ$</td>
<td>$89.1^\circ$</td>
<td>$32.7\text{mm}$</td>
</tr>
<tr>
<td>B3D1T3</td>
<td>$89.9^\circ$</td>
<td>$89.9^\circ$</td>
<td>$32.8\text{mm}$</td>
</tr>
<tr>
<td>B3D2T1</td>
<td>$88.6^\circ$</td>
<td>$88.5^\circ$</td>
<td>$33.1\text{mm}$</td>
</tr>
<tr>
<td>B3D2T2</td>
<td>$90.0^\circ$</td>
<td>$89.8^\circ$</td>
<td>$33.5\text{mm}$</td>
</tr>
<tr>
<td>B3D2T3</td>
<td>$90.0^\circ$</td>
<td>$89.6^\circ$</td>
<td>$32.8\text{mm}$</td>
</tr>
<tr>
<td>B3D3T1</td>
<td>$90.0^\circ$</td>
<td>$90.0^\circ$</td>
<td>$45.3\text{mm}$</td>
</tr>
<tr>
<td>B3D3T2</td>
<td>$91.4^\circ$</td>
<td>$91.2^\circ$</td>
<td>$34.0\text{mm}$</td>
</tr>
<tr>
<td>B3D3T3</td>
<td>$91.1^\circ$</td>
<td>$90.8^\circ$</td>
<td>$33.7\text{mm}$</td>
</tr>
</tbody>
</table>
holder force channels have the most consistent shape because there are no extreme shape error values.

7.7.2.1 Low blank holder force channels

The low blank holder force channels, Figure 7-8(a), appear to decrease their shape error with respect to the experimental channels as the die radii increases. There is no consistent trend that can be assigned to the tool gap parameter. A comparison of the manually measured data for the low blank holder force simulated channels against the experimental channels revealed the following:

- the lengths of the simulated channels' flanges are much longer than the corresponding experimental channels; the average simulated flange length = 33.8mm, and the average experimental flange length = 29.8mm;

- the flange angles of the simulated channels are much higher than the experimental channels; the average simulated flange angle = 92.9°, and the average experimental flange angle = 88.9°;

- the floor angles of the simulated channels are lower than the experimental channels; the average simulated floor angle = 94.5°, and the average experimental channel floor angle = 96.4°.

Figure 7-9: Comparing the low blank holder force simulated channels (B1D1T1 and B1D3T3) (dotted lines) and their associated absolute $\vec{b}$ vector errors with respect to their corresponding experimental channels (solid lines). The reconstructed simulated channel using the simulated channels $\vec{b}$ vector is shown using a dashed line. The simulated channel modal error is the absolute difference between the simulated and the corresponding average experimental $\vec{b}$ vectors.
Figure 7-9 compares two representative simulated low blank holder force channels against their corresponding experimental channels. The simulated channel is displayed using a dotted line while the experimental channel is displayed using a solid line. The dashed line, which is graphed over the dotted line, is the reconstructed simulated channel using its $\vec{b}$ vector. There is some loss of shape information when a shape is converted into $\vec{b}$ vector space because not all the variation modes are included in the $\mathbf{P}$ matrix. The reconstructed line is included in the plots to verify the $\vec{b}$ vector still adequately represents the channel. This check is performed to determine whether the simulated channels are still being correctly represented because the shape variation modes are from the experimental MSVM, not from the simulated channels themselves. The simulated B1D1T1 channel, see Figure 7-9(a), has two aspects which are too large: upper angle and flange length. The floor angle is also too small. These problems are tracked in the $\vec{b}$ vector modes, as there is a large first mode error, which tracks flange length and flange angle, and the third mode error is also large, which tracks floor angle and side wall curl. The simulated B1D3T3 channel has less shape error than the simulated B1D1T1 channel, see Figure 7-9(b). The B1D3T3 flanges are much too long but the flange angle and the floor angle error is improved. This is seen in $\vec{b}$ vector representation as a reduction in the first mode error and a reduction in the third mode error.

7.7.2.2 Medium blank holder force channels

The medium blank holder force channels, Figure 7-8(b), have an inconsistent marginal trend of the shape error decreasing with increasing die radii. The shape error surface also has a very high average simulation shape error and there some channels which have extreme differences with their corresponding experimental channels. The comparison of the manually measured data of the simulated medium blank holder force channels against the experimental channels revealed the following:

- the lengths of the simulated channels’ flanges are substantially longer than the corresponding experimental channels; the average simulated flange length = 47.9mm, and the average experimental flange length = 30.8mm;

- the flange angles of the simulated channels are much higher than the experimental channels; the average simulated flange angle = $94.5^\circ$, and the average experimental flange angle = $90.2^\circ$;

- the floor angles of the simulated channels are higher than the experimental channels; average simulated floor angle = $98.6^\circ$, and the average experimental channel floor angle = $95.6^\circ$.

Figure 7-10 compares two representative simulated medium blank holder force channels against their corresponding experimental channels. It can be seen that the exper-
Figure 7-10: Comparing the medium blank holder force simulated channels (B2D2T2 and B2D2T3) (dotted lines) and their associated absolute \( \vec{b} \) vector errors with respect to their corresponding experimental channels (solid lines). The reconstructed simulated channel using the simulated channels \( \vec{b} \) vector is shown using a dashed line.

Experimental MSVM has some difficulty accurately representing these simulated channels. This is indicated by the dashed line, the MSVM channel approximation, not fully covering the simulated channels near the upper corners of the channels. The rest of the simulated channel is, however, explained correctly. In addition, the simulated B2D2T2 channel has a major problem in that the flange length is extremely long compared to the experimental channel, see Figure 7-10(a). The flange and floor angles are close to the experimental channel but are still too high. These problems are tracked in the \( \vec{b} \) vector modes, where the first mode error is huge and the third mode error is also abnormally high. The simulated B2D2T3 channel has more shape error than the simulated B2D2T2 channel, see Figure 7-10(b). The flanges are again substantially longer than the experimental comparison channel, but in this case the flange angle is too small and either the floor angle or the side wall curl is much too large. This is seen in the \( \vec{b} \) vector representation as a large increase in the first and third modal errors.

### 7.7.2.3 High blank holder force channels

The high blank holder force channels, Figure 7-8(c), appear to have little shape error with respect to the experimental channels. There is no consistent trend from either die radii or tool gap. The simulated B3D3T1 channel has a much larger error than any of the other high blank holder force simulated channels. The comparison of the manually measured data for the simulated high blank holder force channels against the experimental channels revealed the following:

- the lengths of the simulated channels' flanges are marginally longer
than the corresponding experimental channels; the average simulated flange length = 34.4mm, and the average experimental flange length = 33.2mm;

- the flange angles of the simulated channels are smaller than the experimental channels; the average simulated flange angle = 90.0°, and the average experimental flange angle = 92.2°;

- the floor angles of the simulated channels are lower than the experimental channels; the average simulated floor angle = 93.3°, and the average experimental channel floor angle = 94.2°.

**Figure 7-11:** Comparing the high blank holder force simulated channels (B3D3T1 and B3D3T2) (dotted lines) and their associated absolute $\hat{b}$ vector errors with respect to their corresponding experimental channels (solid lines). The reconstructed simulated channel using the simulated channels $\hat{b}$ vector is shown using a dashed line.

Figure 7-11 compares two representative simulated high blank holder force channels against their corresponding experimental channels. The simulated B3D3T1 channel has a major error in the flange length, see Figure 7-11(a). This channel's flange angle is too large and it also appears to have a small amount of reverse side wall curl. These problems are tracked in the $\hat{b}$ vector modes as large first and third mode errors. The simulated B3D3T2 channel has a very small amount of shape error, see Figure 7-11(b). The flanges are only slightly longer than the experimental channel and the flange and floor angles are marginally too small. This is seen in the $\hat{b}$ vector representation with small errors for each mode except the fourth mode. This mode tracks the shape of the upper corners.
7.7.2.4 Discussion of the simulation shape error results

This section discusses the major results of the simulation shape error function from the previous section. It was seen that the $\mathbf{b}$ vector error does track the manually measured errors very well. The first mode tracked the error in the flange length, the third mode tracked the error in the sidewall curl, and the fourth mode tracked the error in the upper corner of the channel. These errors in the $b$ vectors were combined in the simulation shape error function and the results were seen in Figure 7-8.

Figure 7-8(a) looks remarkably similar to Figure 6-18(a) on page 144. Figure 6-18(a) plotted the shape error between the low blank holder force experimental channels versus an artificially created desired channel that had no springback. Essentially, the desired springback free channel was similar to a high blank holder force channel. The similarity between the two figures implies that the simulated channels for low blank holder force are similar to channels which have no springback or high blank holder force. The channel forming simulation, therefore, should be modified to increase the amount of springback released in the unloading of the channels. This could be achieved by increasing the residual stress released in the implicit simulation or increasing the amount of residual stress remaining at the end of the explicit simulation.

Figure 7-8(b) indicated that the medium blank holder force simulated channels were not accurate. This agreed with the conclusions in Section 7.6.2. The main errors were the extremely long flange lengths which appeared as large $b(1)$ errors and large side wall curl errors, $b(3)$ errors. These results indicate that the channel forming simulation has to be improved by better modelling of the flanges.

Figure 7-8(c) shows that the high blank holder force simulated channels have very little error when compared to the experimental channels. The improvement that can be made to the simulation, in this case, is to fine tune the simulation by minimising the simulation shape error.

The simulation shape error function, which combines the weighted errors of the $\mathbf{b}$ vectors, gives agreement in the trends from the manually measured errors. Thus, the analysis of the shape error in simulations can be handled in an automatic way using a shape manufacturing feedback model which has been trained previously on the appropriate experimental data.

7.8 Summary

This chapter has applied the shape manufacturing feedback model to simulated sheet metal forming data. The shape manufacturing feedback model used the experimental data from the previous chapter as a comparison. The main contribution of this chapter is the automatic, quantitative, global error analysis that the shape manufacturing feedback model brings to the comparison of simulations with their corresponding ac-
tual processes. The simulation shape error function uses the actual experimental shape variations to measure the difference in shape between the experimental and simulated channel. The importance of using the actual shape variations is that it weights the significant errors in the actual manufacturing process above the insignificant errors. Thus, the low and medium blank holder force simulated channels in this chapter had longer flange lengths relative to their corresponding experimental channels and this increased their errors significantly. This chapter's simulations did not model the actual shape variation trends to any reasonable degree. Several suggestions were made about how to improve the simulations, such as, increasing the springback released from low blank holder force simulation channels, and varying the simulation and material model parameters, but the investigation of these suggestions was out of the scope of this thesis.

The shape manufacturing feedback model, through the simulation shape error function, provides an overall measure of a simulation's accuracy. Indeed, the simulation shape error function can be a very useful tool for investigating the current benchmarks by enabling the finite element modellers to have a quantitative, global assessment of their simulations with respect to the actual process. Furthermore, the benefits of using the shape manufacturing feedback model can be truly gained if the benchmarking of simulations is improved by widening the number of process parameter combinations investigated within a benchmark. Currently, a benchmark considers only a single process parameter combination of the process. This chapter has shown that a simulation may give reasonable results for a single combination, in our case high blank holder force, but they may not give accurate results for other process parameter combinations.
Chapter 8

Conclusions

Shape Odyssey 2001 finishes

8.1 Introduction

This chapter first reviews the major contributions and results of the work provided in this thesis. After which, there is a discussion of the future work that this thesis has opened for further research.

8.2 Major contributions and results

This section is divided into three areas: first, the general conclusions of the thesis are discussed; this is followed by the specific results for both forging and sheet metal forming.
8.2.1 General conclusions

The question posed at the beginning of this thesis was:

**Whether it is possible to characterise the geometric variation from a manufactured product to develop relevant feedback information for manufacturing control.**

This question was divided into two components: first, characterising shape variation; and second, providing feedback information. This thesis proposed that the shape variation can be well characterised by the Manufacturing Shape Variation Model (MSVM). This model is a combination of the Point Distribution Model (PDM) (Cootes et al., 1995) and the Design of Experiments (DoE) methodology. The advantages of the MSVM are:

- the material model of the part does not need to be explicitly known as the MSVM implicitly learns this from the output of the process;
- the variation of the part is described by the principal variations of the manufacturing process which emphasise the significant variations of the process, unlike Fourier descriptors or other such models; and
- the characterisation of part variation from the average part is compact because only a small number of deformable model parameters are needed to describe the part.

These advantages were seen when the MSVM was applied to the shape manufacturing processes of forging, experimental and simulated sheet metal forming in Chapters 5, 6, and 7.

The feedback information models that were developed in this thesis were restricted to two forms, shape error functions and inverse models. The shape error function provides a quantitative value of the global error between the manufactured part and its desired shape. The advantages of the shape error function are:

- the shape errors that are significant to the manufacturing process are weighted higher than the other shape errors;
- the boundary information of the inspected part does not need to be low pass filtered or smoothed.

The shape error function was applied to good effect in forging and experimental sheet metal forming in Chapters 5 and 6. In addition, the shape error function was implemented to investigate the accuracy of the simulation of an actual sheet metal forming process in Chapter 7. The results from this application were very promising.

The second form of feedback provides information on the possible causes of the shape error. This information is provided by an inverse shape variation–process parameter
model that estimates the combination of process parameter levels sufficient to create the part being inspected. The estimated process parameter combination can then be used in a manufacturing control system to provide an indication of which process parameters need to be modified. The shape variation–process parameter inverse model is a new approach to feedback systems. Previously, most inverse models have used the underlying material models to drive their inverse algorithms (Fourment & Chenot, 1996; Fourment et al., 1996; Ghouati & Gehn, 2001). In contrast, the shape variation–process parameter inverse model learns a relationship between the variation of the output shape and the input settings. This method will be advantageous in situations where the material models are unknown or have not been fully determined, such as springback. Inverse models of reasonable accuracy were developed for forging and sheet metal forming in Chapters 5 and 6.

Finally, the shape manufacturing feedback model, as a whole, has been put into perspective within the relevant literature on automatic manufacturing control. Research into automatic control of manufacturing processes, excluding in-process control, is still largely unexplored (Hardt, 1993). Webb and Hardt (1991) developed one of the first process-tuning sheet metal forming control systems. Since this work, research into process-tuning systems has been limited. The problem is that automatic process-tuning combines the three difficult areas of industrial inspection, feedback and process rectification. The shape manufacturing feedback model bridges the gap between inspection and feedback. In the future, work can be performed to integrate process rectification into the model, as will be discussed in Section 8.3.

In summary, there were several major contributions from this thesis, namely:

1. a novel method for measuring shape error of forged products;
2. the development of an inverse model that learns a relationship between the final shape of a simulated forged part and its associated forging process parameters to a reasonable accuracy;
3. the extension of the forging shape error function to the inspection of actual sheet metal formed parts with discontinuities in shape, such as corners, which previous sheet metal shape error measures have had to filter out;
4. the development of a novel simple empirical springback quantifier;
5. the development of an inverse mapping between shape variation of sheet metal channels and the process parameter settings of the press;
6. the development of a shape error function to determine the accuracy of finite element sheet metal simulations.
In conclusion, characterising geometric shape variation in shape manufacturing processes can be used to develop process parameter feedback models.

8.2.2 Forging, results and contributions

There were two main contributions of applying the shape manufacturing feedback model to simulated forging processes. The first contribution was the development of the forging shape error function. It gave accurate results for both simple and complex forging geometries. The second contribution was the creation of an inverse shape variation-process parameter model that had an average accuracy of 80% except where there was no shape variation information available to make a reasonable classification.

8.2.3 Sheet metal forming, results and contributions

8.2.3.1 Experimental sheet metal forming

The shape manufacturing feedback model was then applied to a sheet metal forming process, experimental “U” channel forming. This application of the shape manufacturing feedback model brought about three main contributions. The first was the creation of a channel shape error function. The shape error function was validated against manually measured shape errors. The function gave good agreement to the manually measured shape errors, although the channel shape error function was biased toward weighting the flange error. The flange variation was the most significant form of variation within the channel forming process and this was highlighted by the channel shape error function. The second contribution was to extend the channel shape error function to measure springback error. The springback error function was also validated against manually measured shape errors. The springback error function also corresponded well with the manually measured results. The springback error results did show that the channels were influenced by the blank holder force and the die radii which agreed with Kim and Thomson’s (1989) results. The influence of the tool gap, however, was insignificant, which was inconsistent with Kim and Thomson’s (1989) experimental results. The lack of influence in the tool gap was partly explained by the experimental set-up. It was further found that the channel shape error function was useful in determining quality control measures of the end product. The channel shape error function was utilised to decide whether the manufacturing process needed re-tuning. The final contribution to experimental sheet metal channel forming was to create inverse shape variation-process parameter models. The inverse models were created using linear, quadratic and ANN classifiers that related the \( \vec{b} \) vectors to the corresponding process parameter levels. The quadratic classifiers were the most accurate predictors, and the most consistent classifiers over all the parameters. The quadratic classifiers had average accuracies of: Blank holder force – 97.69%; Die radii – 94.62%; Tool gap – 70.00%. The
nature of the data was such that it suited the quadratic classifier over the ANN. The
ANN did better on the more complex tool gap data, however it was not a significant
improvement.

8.2.3.2 Simulated sheet metal forming

The shape manufacturing feedback model was also applied to simulated sheet metal
forming data. The shape manufacturing feedback model was used to compare the
simulation results of “U” channel forming against the experimental results of the same
process. The simulations of the sheet metal channel forming process were only accurate
for the channels with high blank holder force. The aim of the simulations, however,
was to provide a data set for which the shape manufacturing feedback model could
be applied, not to obtain an exact model of the process. The main contribution to
simulated sheet metal forming was the creation of a simulation shape error function.
This function used the experimental channel variations as a basis to measure the global
difference between the experimental and simulated channels. The experimental chan­
nel variations weight the significant errors over the insignificant errors with respect to
the actual process. The simulation shape error function can also be used for simula­
tion benchmark problems to give finite element modellers quantitative feedback on the
overall shape error of their simulation.

8.3 Future work

This section outlines the future research areas that can explored from the work in
this thesis. The areas of research are divided into three sections: first, the general
improvements that can be made to the shape manufacturing feedback model; second,
the specific improvements that can be made within forging and sheet metal forming
with respect to the model; and finally, a proposed manufacturing control system based
on the shape manufacturing feedback model.

8.3.1 General improvements

The main area of future research arising from the work in this thesis is to progress the
shape manufacturing feedback model into three dimensions. The PDM approach has
been already implemented in three dimensions (Hill et al., 1993; Daniel et al., 1997).
The extension to three dimensions raises the difficulty of assigning boundary points to
the surface of a 3-D part. The automatic meshing algorithms used in finite element
modelling may provide an initial solution that could be modified to suit the 3-D shape
manufacturing feedback model. The important issue in this case is to get a boundary
point mesh that is consistent across the training set, such that each point represents
the same region of variation on the part’s surface. In addition, the number of points
that are needed to optimally describe the surface is still an open problem and worth researching.

The second improvement that can be made to the shape manufacturing feedback model is to modify the PDM approach. The are a number of advances that have been made with respect to the PDM approach, see Section 3.2.2.5. Non-linear principal component analysis could be applied within the PDM approach which may realise a better set of \( \hat{b} \) vectors that have greater separation between the process parameter combinations. This assumes that there are non-linear relationships in both the shape variation and the process parameter combinations.

The PDM approach can also be improved by implementing the FEM/PDM combined model (Cootes & Taylor, 1994), see Section 3.2.2.5. This is particularly advantageous in situations where there are only a few training examples available and the material models are known. The use of FEM modes enables the material model of the part to be initially included in the PDM description of the shape.

8.3.2 Specific improvements

There are specific improvements that can be made to the shape manufacturing feedback model with respect to forging, experimental sheet metal forming, and simulated sheet metal forming.

The results from this thesis have shown great promise for simulated forging data and the next step is to investigate actual forging data using the shape manufacturing feedback model. Unfortunately, no actual forging data was available for this thesis.

The next step for experimental sheet metal forming is to apply the shape manufacturing feedback model to 3-D parts. This is particularly relevant to measuring springback in 3-D parts as there is, to this author's knowledge, no standard method of measuring 3-D springback. This is of fundamental importance to the automotive industry who are using material that is increasingly prone to springback. In addition, there is an opportunity to extend the shape manufacturing feedback model to process parameter optimisation of simulations. The shape manufacturing feedback model could assist in finding the appropriate process parameter levels that will match simulated parts to actual parts. The results of this optimisation could then be compared to those of Gelin and Ghouati (2001) who also determine the parameter levels corresponding to an actual part.

The shape manufacturing feedback model can also be extended to optimise the process parameters for simulated sheet metal forming process. This can be achieved by implementing a modified shape variation–process parameter inverse model that returns the error in each process parameter. A MSVM could be trained for simulated sheet metal forming data using different simulation types, including different numbers of nodes and integration types. The modified inverse shape variation–simulated process
Three Stages

1. Initial Learning
2. Final Learning
3. Inspection

Parameter feedback model could be implemented, which would return the process parameter levels that are needed to give the experimental channel outcome. The process parameter error would be the difference between the feedback model parameter estimates and the parameter levels used in the simulation. Finally, this feedback model would form part of an optimisation loop such that the process parameter error decreased until a simulation matched the actual result.

8.3.3 Shape manufacturing control systems

This section extends the shape manufacturing feedback model towards the final goal of manufacturing control. The shape manufacturing feedback model transforms shape variation into feedback information for the process parameters. This completes a closed loop between the output of a shape manufacturing process and the tuning of this process. It opens up the possibility of applying control theory to shape manufacturing systems.

A possible scheme to pursue is that of a three-stage shape manufacturing control system based on the shape manufacturing feedback model, see Figure 8-1.

The first stage uses a finite element model to approximate the manufacturing process. The shape manufacturing feedback model is applied to determine the relationship between the estimated end geometry and process parameters. The finite element model gives the control system knowledge of the gross features of the input/output relationship.

The second stage introduces the real manufacturing process to the shape manufacturing feedback model. This enables the control system to gain knowledge about the
finer details of the output/process parameter relationship. In addition, this stage will gain statistical information about the manufacturing process.

The third stage actually implements the control of the manufacturing system. The shape manufacturing feedback model applies the relationships learnt in stages one and two to provide feedback about the errors in the process parameters. In addition, an adaptive background process is implemented to monitor changes due to wear and fatigue. This background process updates the shape manufacturing feedback model when necessary.
References


201


Efron, B. 1982. The Jackknife, the Bootstrap and Other Resampling Plans. Philadelphia: SIAM.


tion of PDMs using polynomial regression. *Pages 397–406 of: British Machine
Vision Conference.*

Machine Vision Conference.*

dimensional inspection with coordinate measuring machines. *IEEE Transactions

Spyridi, A.J., & Requicha, A.A.G. 1993. Automatic Planning for Dimensional Inspec-

models. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 14*(11),
1061–1075.

approach for automated defect detection and neural classification of web textile


Taylor, L.M., Cao, J., Karafillis, A.P., & Boyce, M.C. 1993. Numerical Simulations of
Sheet Metal Forming. In: A. Makinovchi, et al. (ed), *Proceedings of 2nd Interna-
tional Conference, NUMISHEET'93.*


2213.

Tonapi, S.S., & Srihari, K. 2000. Implementation of the pin-in-paste process in a con-
tract manufacturing environment. *International Journal of Production Research,
38*(17), 4535–4553.


*International Journal of Mechanical Sciences, 32*, 891.


Appendix A

Analysis of linear models for forging data

The basic linear model structure, see equation (5.3), creates the following linear regression models for both forging data sets ("M" shape and simple die pair cases):

\[
\text{Process parameter value} = \begin{pmatrix}
\tilde{\beta}(0) \\
\tilde{\beta}(1) \\
\vdots \\
\tilde{\beta}(18) \\
\end{pmatrix}
+ \begin{pmatrix}
\text{std err}(0) \\
\text{std err}(1) \\
\vdots \\
\text{std err}(18) \\
\end{pmatrix}
\begin{pmatrix}
\bar{b}(1) \\
\bar{b}(19) \\
\vdots \\
\bar{b}(20) \\
\end{pmatrix}
\]  

(A.1)

where \( \tilde{\beta} \) is a vector of the model's parameters and below each model parameter is the standard error of the parameter. In addition, the values of \( \tilde{\beta} \) and the standard errors of each model parameter value are calculated assuming the \( t_{\alpha/2} = 1.960 \) for a confidence of 95%, \( \alpha = 5\% \).

The resulting linear regression models are shown in Tables A.1, A.2, A.3, A.4, A.5, A.6, and A.7. Note that some of the model parameters are not significant and this is indicated by the standard error being larger than the parameter value.

According to Johnston (1987), a model parameter vector, \( \tilde{\beta} \), can have a set of linear hypotheses restrictions that are expressed as,

\[
R \tilde{\beta} = \bar{r},
\]

where \( R \) is the restriction matrix \((q \times k)\) with \( q \) restrictions for the \( k \) model parameters, note \( q \leq k \). In addition, the vector \( \bar{r} \) is the \( q \)-element restriction vector that assigns the model parameter restrictions to a particular value. Usually the restriction vector is a vector of zeros which imply the restricted parameters are hypothesised to be zero or insignificant. If the normal ordinary least squares assumption is made that \( \tilde{\beta} \sim N(\text{True } \beta, \sigma^2 (X'X)^{-1}) \), then the hypothesis described by the restriction matrix and
### Table A.1: Velocity linear regression model, simple die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>10.0000</td>
<td>-0.0006</td>
<td>-0.0032</td>
<td>0.6408</td>
<td>-0.0804</td>
<td>0.0402</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.1270</td>
<td>0.0018</td>
<td>0.0294</td>
<td>0.0587</td>
<td>0.1088</td>
<td>0.1896</td>
</tr>
<tr>
<td>Index</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.1003</td>
<td>-0.2445</td>
<td>-0.0200</td>
<td>1.2232</td>
<td>-2.3074</td>
<td>-2.9448</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.2322</td>
<td>0.2909</td>
<td>0.4083</td>
<td>0.5085</td>
<td>0.5483</td>
<td>0.6898</td>
</tr>
<tr>
<td>Index</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-2.6847</td>
<td>-3.5339</td>
<td>2.3817</td>
<td>8.6505</td>
<td>-4.7904</td>
<td>1.3967</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.8721</td>
<td>0.9829</td>
<td>1.2808</td>
<td>1.4450</td>
<td>1.5517</td>
<td>1.6954</td>
</tr>
<tr>
<td>Index</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model parameter</td>
<td>4.6264</td>
<td>-7.4298</td>
<td>-9.4296</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard error</td>
<td>1.7526</td>
<td>1.8721</td>
<td>2.0207</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table A.2: Friction linear regression model, simple die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>0.2000</td>
<td>0.0000</td>
<td>-0.0005</td>
<td>-0.0248</td>
<td>0.0027</td>
<td>0.0059</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0010</td>
<td>0.0000</td>
<td>0.0002</td>
<td>0.0004</td>
<td>0.0008</td>
<td>0.0014</td>
</tr>
<tr>
<td>Index</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0109</td>
<td>0.0097</td>
<td>-0.0082</td>
<td>0.0284</td>
<td>-0.0368</td>
<td>-0.0639</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0018</td>
<td>0.0022</td>
<td>0.0031</td>
<td>0.0039</td>
<td>0.0042</td>
<td>0.0053</td>
</tr>
<tr>
<td>Index</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0389</td>
<td>-0.0669</td>
<td>0.0783</td>
<td>0.1097</td>
<td>-0.0338</td>
<td>-0.0280</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0067</td>
<td>0.0075</td>
<td>0.0098</td>
<td>0.0110</td>
<td>0.0118</td>
<td>0.0129</td>
</tr>
<tr>
<td>Index</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.1219</td>
<td>-0.0859</td>
<td>-0.1234</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0134</td>
<td>0.0143</td>
<td>0.0154</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table A.3: Travel linear regression model, simple die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>20.0000</td>
<td>-0.0870</td>
<td>0.0568</td>
<td>-0.0410</td>
<td>-0.0260</td>
<td>-0.2309</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0266</td>
<td>0.0004</td>
<td>0.0062</td>
<td>0.0122</td>
<td>0.0228</td>
<td>0.0397</td>
</tr>
<tr>
<td>Index</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.2544</td>
<td>0.2127</td>
<td>0.1343</td>
<td>-0.0754</td>
<td>-0.2211</td>
<td>0.2270</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0486</td>
<td>0.0609</td>
<td>0.0855</td>
<td>0.1064</td>
<td>0.1148</td>
<td>0.1472</td>
</tr>
<tr>
<td>Index</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.4482</td>
<td>0.0725</td>
<td>0.9560</td>
<td>-0.4285</td>
<td>2.2258</td>
<td>-0.0044</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.1826</td>
<td>0.2058</td>
<td>0.2682</td>
<td>0.3026</td>
<td>0.3249</td>
<td>0.3578</td>
</tr>
<tr>
<td>Index</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.9198</td>
<td>0.4559</td>
<td>-1.0690</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard error</td>
<td>0.3669</td>
<td>0.3919</td>
<td>0.4231</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table A.4: Velocity linear regression model, “M” shape die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>10.0000</td>
<td>0.0002</td>
<td>0.0005</td>
<td>0.0001</td>
<td>-0.0111</td>
<td>-0.0183</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.2666</td>
<td>0.0032</td>
<td>0.0037</td>
<td>0.0306</td>
<td>0.0258</td>
<td>0.0752</td>
</tr>
<tr>
<td>Index</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.0005</td>
<td>-0.0200</td>
<td>-0.0161</td>
<td>0.0539</td>
<td>-0.0180</td>
<td>0.1041</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0942</td>
<td>0.1042</td>
<td>0.1146</td>
<td>0.1183</td>
<td>0.1319</td>
<td>0.1545</td>
</tr>
<tr>
<td>Index</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0200</td>
<td>0.1042</td>
<td>-0.2019</td>
<td>0.0187</td>
<td>-0.3214</td>
<td>0.0358</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.1658</td>
<td>0.1896</td>
<td>0.1974</td>
<td>0.2299</td>
<td>0.2383</td>
<td>0.2766</td>
</tr>
<tr>
<td>Index</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0766</td>
<td>0.0857</td>
<td>-0.0968</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard error</td>
<td>0.3030</td>
<td>0.3352</td>
<td>0.3712</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table A.5: Friction linear regression model, “M” shape die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>0.2000</td>
<td>0.0000</td>
<td>0.00003</td>
<td>-0.0001</td>
<td>0.0008</td>
<td>0.0004</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0021</td>
<td>0.0000</td>
<td>0.00003</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0005</td>
</tr>
<tr>
<td>Index</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0036</td>
<td>0.0003</td>
<td>0.0004</td>
<td>-0.0034</td>
<td>0.0025</td>
<td>-0.0288</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0007</td>
<td>0.0009</td>
<td>0.0009</td>
<td>0.0009</td>
<td>0.0010</td>
<td>0.0012</td>
</tr>
<tr>
<td>Index</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.0095</td>
<td>-0.0080</td>
<td>0.0183</td>
<td>0.0005</td>
<td>0.0060</td>
<td>0.0041</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0013</td>
<td>0.0015</td>
<td>0.0015</td>
<td>0.0018</td>
<td>0.0018</td>
<td>0.0021</td>
</tr>
<tr>
<td>Index</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0028</td>
<td>-0.0148</td>
<td>-0.0005</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0024</td>
<td>0.0026</td>
<td>0.0030</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table A.6: Travel linear regression model, “M” shape die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>35.0000</td>
<td>-0.0016</td>
<td>-0.0394</td>
<td>0.0030</td>
<td>-0.0116</td>
<td>-0.0249</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0428</td>
<td>0.0005</td>
<td>0.0006</td>
<td>0.0035</td>
<td>0.0040</td>
<td>0.0116</td>
</tr>
<tr>
<td>Index</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Model parameter</td>
<td>0.0291</td>
<td>-0.0293</td>
<td>0.0022</td>
<td>-0.0730</td>
<td>-0.0025</td>
<td>-0.0815</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0135</td>
<td>0.0160</td>
<td>0.0176</td>
<td>0.0182</td>
<td>0.0201</td>
<td>0.0238</td>
</tr>
<tr>
<td>Index</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0097</td>
<td>-0.0059</td>
<td>-0.0476</td>
<td>-0.0020</td>
<td>0.0307</td>
<td>-0.0254</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0256</td>
<td>0.0294</td>
<td>0.0304</td>
<td>0.0347</td>
<td>0.0367</td>
<td>0.0426</td>
</tr>
<tr>
<td>Index</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model parameter</td>
<td>-0.0196</td>
<td>-0.0536</td>
<td>0.0397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0466</td>
<td>0.0516</td>
<td>0.0571</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table A.7: Offset linear regression model, "M" shape die pair

<table>
<thead>
<tr>
<th>Index</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>0.0000</td>
<td>0.0782</td>
<td>-0.0051</td>
<td>-0.0001</td>
<td>0.0357</td>
<td>0.0085</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0000</td>
<td>0.0002</td>
<td>0.0003</td>
<td>0.0012</td>
<td>0.0018</td>
<td>0.0053</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Index</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>-0.0010</td>
<td>0.0154</td>
<td>0.0560</td>
<td>0.0386</td>
<td>-0.0877</td>
<td>0.0506</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0058</td>
<td>0.0072</td>
<td>0.0080</td>
<td>0.0082</td>
<td>0.0092</td>
<td>0.0107</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Index</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>0.0188</td>
<td>0.1585</td>
<td>0.0676</td>
<td>0.0613</td>
<td>0.0405</td>
<td>-0.0958</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0116</td>
<td>0.0132</td>
<td>0.0137</td>
<td>0.0160</td>
<td>0.0166</td>
<td>0.0192</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Index</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model parameter</td>
<td>-0.0597</td>
<td>-0.1015</td>
<td>0.2060</td>
</tr>
<tr>
<td>Standard error</td>
<td>0.0211</td>
<td>0.0233</td>
<td>0.0258</td>
</tr>
</tbody>
</table>

The restriction vector have the following distribution:

\[
\left( R\hat{\beta} - \hat{\tau} \right) \sim N \left( \tilde{0}, \sigma^2 R (X'X)^{-1} R' \right).
\]

Therefore, an F ratio can be formed such that if \( R\hat{\beta} = \hat{\tau} \) is true, then

\[
\frac{\left( R\hat{\beta} - \hat{\tau} \right)' \left[ R (X'X)^{-1} R' \right]^{-1} \left( R\hat{\beta} - \hat{\tau} \right) / q}{\hat{\varepsilon}^2 / (n - k)} \sim F(q, n - k),
\]

where \( q \) is the number of restrictions, \( k \) is the number of model parameters and \( n \) is the number of observations in the data matrix.

The (simple) velocity parameter had the following individual insignificant model parameters: \( \{\tilde{\beta}(1), \tilde{\beta}(2), \tilde{\beta}(4), \tilde{\beta}(5), \tilde{\beta}(6), \tilde{\beta}(7), \tilde{\beta}(8), \tilde{\beta}(17)\} \). An F-test was conducted to determine if all these parameters were simultaneously insignificant.

Null Hypothesis : \( \tilde{\beta}(1) = \tilde{\beta}(2) = \tilde{\beta}(4) = \tilde{\beta}(5) = \tilde{\beta}(6) = \tilde{\beta}(7) = \tilde{\beta}(8) = \tilde{\beta}(17) = 0 \)

\( F(8, 1000) = 1.95 \) (At the 95\% confidence level)

\{Velocity\} F statistic = 1.0951 \( \Rightarrow \) Accept Null Hypothesis

(A.3)

The (simple) friction parameter had the following individual insignificant model parameter: \( \{\tilde{\beta}(1)\} \).

The (simple) travel parameter had the following individual insignificant model parameters: \( \{\tilde{\beta}(9), \tilde{\beta}(13), \tilde{\beta}(17)\} \). An F-test was conducted to determine if all these parameters were simultaneously insignificant.
Null Hypothesis: $\tilde{\beta}(9) = \tilde{\beta}(13) = \tilde{\beta}(17) = 0$

\[ F(3, 1000) = 2.61 \text{ (At the 95% confidence level)} \tag{A.4} \]

\{ Travel \} \text{ } F \text{ statistic } = 0.8016 \Rightarrow \text{ Accept Null Hypothesis}

The ("M" shape) velocity parameter had the following individual insignificant model parameters: \{ $\tilde{\beta}(1), \tilde{\beta}(2), \tilde{\beta}(3), \tilde{\beta}(5), \tilde{\beta}(6), \tilde{\beta}(7), \tilde{\beta}(8), \tilde{\beta}(9), \tilde{\beta}(10), \tilde{\beta}(11), \tilde{\beta}(12), \tilde{\beta}(13), \tilde{\beta}(15), \tilde{\beta}(17), \tilde{\beta}(18), \tilde{\beta}(19), \tilde{\beta}(20) \}$. An F-test was conducted to determine if all these parameters were simultaneously insignificant.

Null Hypothesis: $\tilde{\beta}(1) = \tilde{\beta}(2) = \tilde{\beta}(3) = \tilde{\beta}(4) = \tilde{\beta}(5) = \tilde{\beta}(6) = \tilde{\beta}(7) = \tilde{\beta}(8) = \tilde{\beta}(9) = \tilde{\beta}(10) = \tilde{\beta}(11) = \tilde{\beta}(12) = \tilde{\beta}(13) = \tilde{\beta}(15) = \tilde{\beta}(17) = \tilde{\beta}(18) = \tilde{\beta}(19) = \tilde{\beta}(20) = 0$

\[ F(18, 625) \approx F(16, 1000) = 1.65 \text{ (At the 95\% confidence level)} \tag{A.5} \]

\{ Velocity \} \text{ } F \text{ statistic } = 0.3280 \Rightarrow \text{ Accept Null Hypothesis}

The ("M" shape) friction parameter had the following insignificant model parameters: \{ $\tilde{\beta}(1), \tilde{\beta}(3), \tilde{\beta}(5), \tilde{\beta}(7), \tilde{\beta}(8), \tilde{\beta}(15), \tilde{\beta}(20) \}$. An F-test was conducted to determine if all these parameters were simultaneously insignificant.

Null Hypothesis: $\tilde{\beta}(1) = \tilde{\beta}(3) = \tilde{\beta}(5) = \tilde{\beta}(7) = \tilde{\beta}(8) = \tilde{\beta}(15) = \tilde{\beta}(20) = 0$

\[ F(7, 625) \approx F(7, 1000) = 1.202 \text{ (At the 95\% confidence level)} \tag{A.6} \]

\{ Velocity \} \text{ } F \text{ statistic } = 0.6442 \Rightarrow \text{ Accept Null Hypothesis}

The ("M" shape) travel parameter had the following individual insignificant model parameters: \{ $\tilde{\beta}(3), \tilde{\beta}(8), \tilde{\beta}(10), \tilde{\beta}(12), \tilde{\beta}(13), \tilde{\beta}(15), \tilde{\beta}(16), \tilde{\beta}(17), \tilde{\beta}(18), \tilde{\beta}(20) \}$. An F-test was conducted to determine if all these parameters were simultaneously insignificant.

Null Hypothesis: $\tilde{\beta}(3) = \tilde{\beta}(8) = \tilde{\beta}(10) = \tilde{\beta}(12) = \tilde{\beta}(13) = \tilde{\beta}(15) = \tilde{\beta}(16) = \tilde{\beta}(17) = \tilde{\beta}(18) = \tilde{\beta}(20) = 0$

\[ F(10, 625) \approx F(10, 1000) = 1.84 \text{ (At the 95\% confidence level)} \tag{A.7} \]

\{ Travel \} \text{ } F \text{ statistic } = 1.0261 \Rightarrow \text{ Accept Null Hypothesis}

The ("M" shape) offset parameter had the following individual insignificant model parameters: \{ $\tilde{\beta}(0), \tilde{\beta}(3), \tilde{\beta}(6) \}$. An F-test was conducted to determine if all these parameters were simultaneously insignificant.

Null Hypothesis: $\tilde{\beta}(0) = \tilde{\beta}(3) = \tilde{\beta}(6) = 0$

\[ F(3, 625) \approx F(3, 1000) = 2.61 \text{ (At the 95\% confidence level)} \tag{A.8} \]

\{ Offset \} \text{ } F \text{ statistic } = 0.0475 \Rightarrow \text{ Accept Null Hypothesis}
Appendix B

Forging simulation code

Fichier des données générales pour Forge2

.Fichier
Fmay = simptv301.may ! Mesh file
Fout = simptv301.out ! Die file
Fres = simptv301.res ! Result file
Finc = simptv301.inc ! Increment file
Faux = simptv301.aux ! Auxiliary file
! Freo = simptv301.reo
delete
.Fin Fichier

.Unites
mm-KPa-mm.kg.s ! Unit system for computation
.Fin Unites

.Rheologie
thermoecroui: Exp Beta T, ! Exponential thermal law
beta = 2.5e-03, ! Temperature term
n = 1.74e-01, ! Strain-hardening
Eb0 = 1.0e-04 ! Strain-hardening regularisation

m = 1.38e-01 ! Sensitivity to strain rate
p = 1.38e-01 ! Sensitivity to sliding velocity
! Usually p is equal to m
alpha = 3.00e-01 ! Friction coefficient alpha = 2.0e-01
glissant ! Sliding contact
K=1.625e+06 ! Constant term Ko

temp init = 1.05e+03 ! Initial temperature ( celsius )
.Fin Rheologie

.Increment
  Deformation=2.0D-2 ! Average strain increment - see time step
  PAS DE Calage ! pas de calage automatique
  t ouverture=0. DO ! Temps pour mettre outils pos. ouverte
.Fin Increment

.Toler Conv
  Rayon Conge= 0.1 ! Rayon rac. segment des outils
  Nb incr max= 5000
.Fin Toler Conv

.Execution
  dt contact+deform ! Time step function of contact and strain
  Incompress classique ! utilisation incompressibilite CLASSIQUE
  Contact suivi ! deplac noeuds sur outil avec suivi contraintes
  Imp Iteration NR
  Imp resume Contact
  sans visualisation
  ! visualisation ! Dynamic visualisation
  dhsto=7.0 ! Storage according to height
  ! (x unit of length)
  nbpusure = 200
  ! forcemax = X e+07 ! in MilliNewton; X = max force in tons
.Fin Execution

.Thermique
! Definition of the material
  Mvolumique = 7.5e-06 ! Density, ro
  Cmassique = 6.69e+08 ! Heat capacity, c
  Conduct Mat = 2.758e+04 ! Conductivity, k
  Effus Mat = 1.18e+04 !

! Definition of the limit conditions dies/part/free surface - Imposed flux
Appendix B. Forging simulation code

Outil1
alphat = 2.0e+03 ! Global transfert coefficient with dies
Tempout = 3.5e+02 ! Die temperature Deg Celsius
Effus Outil = 9.880e+03 ! effusivity (sqrt(k.ro.c))

Outil2
alphat = 2.0e+03
Tempout = 3.5e+02
Effus Outil = 9.880e+03

Face libre
Alphat = 8 ! Global transfert coefficient with air
Temp Ext = 50 ! external temperature Deg Celsius
Epsilon = 7.0e-01 ! material emissivity

.Fin Thermique

.mauto
taux de penetration global .05 ! global penetration mesh into dies
taux de penetration local .01 ! local penetration
taux de raffinement 1.3
precision .04 ! Accurate boundary-New mesh boundary
! (0=automatic=default)
courbure .25 ! Curvature - C = H/L
courbure max .30 ! Maximum curvature of the elements
homogeneite .35
Nombre de noeuds 200 ! Total number of nodes (triangles'summits)
maillage axi
taillemax 8. ! Maximum size - Length of the element
! (0=automatic=default)

Zone : taillemax 1,
ymax 15,
ymin 10,
xmax 25,
xmin 15
.fin mauto

.Histoire
Chaine: incr= 4 ,
        Fmay= simptv301.may-1
        Faux= simptv301.aux-1
        Finc= simptv301.inc-1

.Fin Histoire
!PLA
Appendix C

Sheet metal forming — experimental procedure

C.1 Experimental set-up

The "U" channel forming test is one of the standard springback tests that can be used to determine the amount of springback that may occur for a particular type of material. The “U” channel forming is a test where an rectangular blank is punched into a “U” shape while the edges of the rectangle blank (flanges) are held down by an applied force.

![Diagram of die set with punch shoe, die shoe and guide pillars.]

*Figure C-1: Die set with punch shoe, die shoe and guide pillars.*

The “U” channel tests are to be performed on a small die set. The die set to be used is a Heine & Sons (Model 80R642) small press which has been modified to allow variability in the sheet metal forming process parameters.

The base die set is made up of a die shoe upon which the die blocks are placed and a punch shoe to which the punch is connected. At the rear of the die shoe there are
two pillars ($\phi 1'' \times 10''$) that guide the punch shoe. The punch shoe complements the die shoe. The punch shoe has two guide bushes that restricts the shoe to only vertical movement. The set-up of the die can be seen in Figure C-1.

![Diagram of sheet metal forming setup](image)

**Figure C-2**: The "U" channel forming test.

The final experimental set-up uses several sensors, see Figure C-2. We use two force sensors to measure the blank holder forces and a linear potentiometer to measure the travel of the punch. The sensors’ signals are connected to a computer to record the data.

A data acquisition card (DAQ card) is used to interface between the sensor and the computer. Labview is the software that displays and manipulates the sensory data within the computer.

For each test sample the blank holder forces and the travel are recorded. Each sample is then scanned to obtain a two dimensional (2-D) cross section. The scanning process involves tracing one side of each test sample to a sheet of paper. The sheet of paper is then scanned into a computer using an electronic scanner at reasonable resolution. The scanned image is saved in the appropriate image file format that can easily be manipulated. The image file is imported into either Matlab and analysed with an image processing routines.

### C.1.1 Equipment

1. Die shoe with guide pins;
2. 2 × Die blocks;
C.1.2 Sensors

- Linear potentiometer
- Load cells

C.1.3 Test material

The test blanks are made out of G3N automotive steel material. The blanks will be 0.76mm in thickness and each blank has the initial dimensions of 150mm by 20mm. The properties of the G3N steel are given in the report by Herron and Lanzon (1997).

As an aside, a check was made to confirm that the springback of a deformed blank does not change over time. The check was performed by recording the shape of several deformed blanks directly after bending and recording the shapes again after a period of two weeks. This phenomenon is uncommon for steel products but is quite prevalent in Aluminium.

C.1.4 Blank holder force

The blank holder force was modified by creating a structure that applied a force on the blank. The force is applied by 4 bolts on each die block. The force is measured by sandwiching a force sensor between two blocks. The force is passed from the bolts through the upper block and force sensor through the lower block to the blank. The force sensor used is a Celtron STC-1500 S type tension load cell which can measure up to 1500kg weight. The set-up of the force sensor can be seen in Figure C-3.

Shims are used on either side of the blank to balance the force on the blank. The blank holder force used for previous experiments with this set-up by Crook (1995) were: 3.3kN, 8.3kN and 13.3kN. It was found that 13.3kN caused necking and tearing in the material and, thus, the blank holder force was decreased to: 3kN, 7.5kN, and 12kN.
Appendix C. Sheet metal forming — experimental procedure

Figure C-3: Diagram of blank holder force sensor set-up.

C.1.5 Die radii

The die blocks were designed such that they incorporated four radii, one on each main edge. The die blocks can be rotated to change the radius. The die radii available are: 3.175mm (1/8"), 4.7625mm (3/16"), 6.35mm (1/4") and 7.9375mm (5/16"). The radii values used were influenced by Kim and Thomson (1989) who also investigated varying the die radii between 2mm and 5mm.

Figure C-4: The four radii of a die block.

The die blocks are held to the die shoe by four bolts which screw into two sliding anchors that sit below the die shoe. The four radii for each die block can be seen in Figure C-4.
C.1.6 Punch radii

There are two punches available, each with two sets of radii. The punches have two opposing sides with a given radius along each axis. If the punch is rotated by 90 degrees around the long axis of the punch, then the other opposing sides have a different radius. The punch radii that can be used are: 3.175mm (1/8"), 4.7625mm (3/16"), 6.35mm (1/4") and 7.9375mm (5/16"). The radii values used are influenced by Kim and Thomson (1989) who also investigated varying the die radii between 2mm and 5mm. The radii of a single punch can be seen in Figure C-5.

![Diagram of the punch radii](image)

**Figure C-5:** Diagram of the radii on the punch.

The punches are aligned using two 6mm diameter guide pins drilled into the punch shoe.

C.1.7 Tool gap

Tool gap is the horizontal distance between the edge of the punch and edge of the die (see Figure C-6). The tool gap between the die blocks and the punch can be modified by sliding the die blocks towards and away from the punch. Each die block is secured to the die shoe by 4 bolts which screw into two sliding anchors on the bottom side of the die shoe. A set of shims is used to obtain the correct distance between the punch and die blocks. The tool gaps to be tested are: 1.05mm, 1.30mm and 1.62mm. These values were chosen to match those values used by Kim and Thomson (1989).

Clearance is the tool gap minus the initial thickness of the blank. The clearance-ratio
232 Appendix C. Sheet metal forming — experimental procedure

Figure C-6: Diagram showing the tool gap between the punch and the die.

is the ratio of the clearance to the initial blank thickness, refer to equation C.1.

\[
\text{clearance - ratio} = \frac{\text{clearance}}{\text{thickness}} = \frac{\text{toolgap - thickness}}{\text{thickness}} \quad (C.1)
\]

Clearance ratio is a useful measurement in sheet metal forming. Kim and Thomson (1989) investigated the effects of clearance-ratio for values between 0.2 and 1.2.

C.1.8 Lubrication

The lubrication of the blank is modified by either having no or some lubricant on the contact surfaces. The exact value of the coefficient of friction is hard to control because it is difficult to accurately change the interaction between surfaces. Thus, there will only be two levels of lubrication, no lubricant or some lubricant.

C.1.9 Calibration of linear potentiometer

The linear potentiometer (Novotechnik, model T-50) was calibrated by measuring its voltage output and comparing this to the length extended as measured by an electronic gauge. Linear regression was used to determine the relationship between the measured distance and the measured voltage.

C.2 Punching a channel

C.2.1 Placing the blank

The blank was first scribed with an arrow indicating which direction was the right hand end of the blank. If lubricant was to used the sheet metal deformation, then it was
applied to both sides of the blank by brush, see Figure C-7. The lubricant was the light oil Ferrocote 6/MAL K2 (Ref 98/026) as is used in automotive sheet metal forming. Lubricant was also applied to the lower blank holder whose face was in contact with the blank. In addition, lubricant was applied to the die block.

![Figure C-7: Lubricating the blank.](image)

The blank was placed with the direction arrow side of the blank facing downwards. The middle of the blank remains relatively unaffected by the channel forming, and when the arrow on the blank is facing down it does not come into contact with anything. This ensured that the arrow remained intact during the metal forming. The blank was placed on the die block between the front and the back screws beneath the lower die block. Care was taken to make sure that the blank was aligned on both the left and right sides with respect to each other. The blank was centred between the bolts/screws which applied the blank holder force. The blank was aligned such that it was half way between the front and back screws/bolts. The lower blank holder was placed carefully on the blank. The load cell and upper blank holder were then placed onto the lower blank holder. The blank holder force was applied to the blank by tightening the screws.

### C.2.2 Applying blank holder force

The blank holder force was applied to the blank via tightening the four screws which were secured to the die block, see Figure C-3. These screws applied a load to the upper blank and this load was transferred through the load cell and the lower die onto the blank. The load cell would then measure the amount of blank holder force.

The screws were tightened evenly so that blank holder force was consistent across the blank. The screws were tightened in a “X” fashion, that is, the lower left hand corner was tightened by an increment then the upper right hand corner and so on, see Figure C-8. The tightening of screws 1 through 4 was continued until the correct blank holder force was reached. The order of screw tightening was chosen so that no
two screws on the front or back sides of the blank were tightened consecutively. It was found experimentally that the flow of the blank past the under side of the blank holder was sensitive to uneven force distribution created by tightening the front or back sides higher than the other. Whereas, the flow was much more insensitive to uneven force distributions along the blank (left/right).

Also the tightening of the screws had to be performed in small increments so to obtain an even force distribution across the blank.

**BHF = 3KN:** For a final blank holder force of 3KN each screw was incrementally tightened by 0.2KN.

**BHF = 7.5KN:** For a final blank holder force of 7.5KN the screws were initially incrementally tightened by 0.5KN. Once a blank holder force of 5.5KN was reached, the incremental increase in force was lowered to 0.2KN.

**BHF = 12KN:** For a final blank holder force of 12KN the screws were initially incrementally tightened by 0.5KN on each screw. Once a blank holder force of 10KN force was reached, the blank holder force was incrementally increased by 0.2KN on each screw.

### C.2.3 Punching the Channel

The channel is punched or drawn using a Enerpac 30 ton manual hydraulic press. Before punching the channel, the punch was lowered carefully onto the blank. Once the punch
was touching the blank, the linear potentiometer, which measured the travel of the punch, was zeroed. This meant that the depth of the channel (pre-punch removal) could be measured by measuring the travel of the punch from the linear potentiometer.

![The travel measurement set-up.](image)

**Figure C-9:** The travel measurement set-up.

There was variability in the depth of the draw because the press was manually operated. A mechanical stop was not used because several punches were used and it would be difficult to set up a multi-height mechanical stop. The depth chosen for the channels was 32mm. This depth gave reasonable draw while leaving a reasonable flange to measure the springback. This decision was made after considering the results from several random heights. It was difficult to get the depth to exactly 32mm. It was easy to overrun the target particularly for the low BHF. Care was taken to not overrun more than 0.5mm, although not all samples fall into this range.

The press was manually operated to push the punch down to create the channel, which made it difficult to maintain a constant velocity of the punch. In addition, the velocity of the punch was very slow. For high BHF, the first period of the draw was difficult and required a lot of force. Nevertheless, when a certain point was reached, the pumping became easier and the velocity of the punch became constant. For medium BHF, the first period of the draw occurred for a longer time than for the high BHF. After this, the punch velocity became constant. For low BHF, the pump required a lot of force to push the punch downwards, however the punch traveled with a faster, but less constant, velocity towards the desired depth.
Once the channel has been formed the punch is removed. This operation was also a manual operation. The valve of the press column was reversed and the press column was removed from the vicinity of the punch shoe. The punch was then lifted and chocked above the deformed blank. The blank holder force was released (see Section C.2.4). The blank was removed and would springback varying amounts depending on the conditions of the set-up. The oil was removed and the shape and the relevant information about the press set-up was recorded and engraved on the bottom of the deformed blank.

C.2.4 Releasing the blank holder

The blank holder was released after the punch had been removed/unloaded. The releasing of the blank holder was performed in a similar fashion to the application of the blank holder. The screws were released in a diagonal fashion on each blank holder in order of screw “1” through to screw “4” as indicated by the diagram, see Figure C-10. The remaining screws were released in a random order so that no systemic release error could be introduced. The load cell was then removed as well as the front two screws. The lower die blocks could be removed to allow the deformed blank to be taken out of the set-up.

![Diagram of Blank Holder Release](image)

**Figure C-10**: Releasing the blank holder.

C.2.5 Problems that occurred in the punching

Because the press column was not driven by a motor, we may have caused backlash at the end of each pump stroke. The travel sensor, however, did not show the punch travel ever moving upwards after any downward stroke. There may possibly be some slight
release of punch force at the end of each stroke which may have affected the springback of the process. This may make the results from this study difficult to compare to other springback studies.

It was also noticed that the samples did not have any “wall curl”, while several of the simulated channels had “wall curl”. This may be due to the type of material that was being formed.

It was also noticed that a high blank holder force samples tended to tear after several samples were produced on the same punch setting. Possibly the punch needed to be tightened after a couple of runs to reset its alignment. The punch was very long and alignment was therefore a big problem. The tearing would always occur on the left side of the sample. Perhaps the punch is slightly tilted to the left edge or the die shoe is not absolutely flat.

This problem can be seen in the series of high BHF samples that were ran on the one setting, where the necking of the left became more noticeable as more samples were drawn.

C.3 Changing the set-up

C.3.1 Changing the punch radii

The punch radii was changed by first choosing the desired punch. There were two punches, one with 2/16th inches and 4/16th inches radii, and the second punch had 3/16th inches and 5/16th inches radii. The appropriate punch was then connected to the punch shoe in the desired alignment. The punch shoe had punch alignment pins to make sure the punch was square to the guide pins. The punch was held in place by a bolt which screwed into the top of the punch tool. A mechanical stop was used to stop the punch shoe from dropping while the punch was being attached to the punch shoe.

C.3.2 Changing the die radii

The die blocks were placed on the die shoe with the corners of the appropriate radii facing each other. The die blocks each were held in place by four bolts. The bolts were screwed into sliding locks below the die shoe.

C.3.3 Changing the tool gap

The tool gap was modified by loosening the die blocks from the die shoe. The punch was lowered such that it extended down its assigned gap in the die shoe. A number of shims were placed between the die block and the punch. The die block was pushed hard against the shims and the punch and the die block was locked into place, see Figure C-11.
Fixing the Die Block to the Die Shoe

Figure C-11: Changing the die radii and the tool gap.
Appendix D

Sheet metal forming - experimental design

D.1 Aim

This series of experiments has several aims. The major aim is to determine which process parameters have the most effect on the springback of a "U" channel. A factorial experiment is then designed to study the shape variation of the three most significant process parameters.

The second aim is to characterise the varying profile of a "U" channel stamped blank under the influence of springback with respect to varying process control variables. This involves developing a response surface of shape deformation measures against the process control parameters of the stamping process. The eventual aim of this characterisation is to attempt to link springback variation to process control variable variation. Finally, this experiment aims to measure some statistical information about the profile of a "U" channel stamped blank. This is because the noise or process variation in the "U" channel system affects the ability to elicit the relationship between the springback variation and the process control variables.

This appendix outlines the process parameters involved in "U" channel forming. The pilot study design is then presented. This is followed by the design of the actual factorial experiment. Finally, a design of an experiment to determine the noise in "U" channel forming is outlined.

D.2 Experimental variables

Within any experiment there are two types of variables. There are variables that can be controlled or set to a particular level. These are called the factors of an experiment. The other type of variables are uncontrollable variables and they assist in contributing noise to the measurements of the experiment's outputs. The uncontrollable variables
can also be split into observable and unobservable variables. Observable variables can be measured even if they cannot be explicitly controlled. In contrast, unobservable variables cannot be measured and contribute to the noise of the system. Table D.1 presents the controllable and uncontrollable variables for our “U” channel forming set-up.

Table D.1: Table that presents the controllable and uncontrollable variables involved in “U” channel forming.

<table>
<thead>
<tr>
<th>Controllable Variables</th>
<th>Uncontrollable Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blank holder force</td>
<td>Travel</td>
</tr>
<tr>
<td>Die radii</td>
<td>Punch force</td>
</tr>
<tr>
<td>Punch radii</td>
<td>Movement in the press</td>
</tr>
<tr>
<td>Tool gap</td>
<td>Movement in the stamping set-up</td>
</tr>
<tr>
<td>Lubrication</td>
<td>Fatigue and wear</td>
</tr>
<tr>
<td></td>
<td>Aging of the blank’s material</td>
</tr>
</tbody>
</table>

Table D.1 shows there are five variables which can be controlled in the current set-up of the die rig: blank holder force (BHF), die radii, tool gap, lubrication and punch radii. There are also random factors which will influence the shape of the deformed blank: material properties, die rig variation, operator variance and punch travel. Through proper experimental design these random factors can be confounded to reduce their influence on the deformed shape.

Table D.2: Table containing the levels for each of the process parameters in the “U” channel forming process.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blank Holder Force</td>
<td>3.3KN 8.3KN 13.3KN</td>
</tr>
<tr>
<td>Die Radii</td>
<td>3.175mm 4.7625mm 6.35mm 7.9375mm</td>
</tr>
<tr>
<td>Tool Gap</td>
<td>1.05mm 1.30mm 1.62mm</td>
</tr>
<tr>
<td>Lubrication</td>
<td>Lubricant No lubricant</td>
</tr>
<tr>
<td>Punch Radii</td>
<td>3.175mm 4.7625mm 6.35mm 7.9375mm</td>
</tr>
</tbody>
</table>

There are 288 combinations of variable (factor) levels given there are: 3 levels of blank holder force; 4 levels of die radii; 3 levels of tool gap; 2 levels of lubrication; and 4 levels of punch radii (see Table D.2). There is, however, only enough material to run 320 tests. If all the combinations were to be tested, then there would be no material for any repetition of experiments. This would lead to the problem of statistical significance when calculating an average combination given the amount of noise in the
process. Moreover, it would be too time consuming to run a full analysis on all the combinations. Therefore, the number of variables investigated needs to reduced.

D.3 Pilot study

D.3.1 Pilot study design

A pilot analysis is run, before any other experiments are performed, to reduce the number of variables to be examined. For this reason the pilot analysis determines which variables have greater influence on the forming process with respect to springback.

The pilot analysis is a $2^k$ no replicate factorial experiment where there are five factors, $k = 5$: blank holder force, die radii, tool gap, lubrication and punch radii. Each factor will have two levels, one high and one low, that are the highest level and lowest level for that factors range. The pilot analysis will have 32 combinations of the 5 factors. To confound operator variance and the other random factors the order the combinations are performed will be randomized. Table D.3 contains the list of tests to be performed in a randomized order.

Table D.3: This table contains the randomized order of the 32 process parameter combinations. The combinations are given by the following notation: b = small blank holder force, and B = large blank holder force; d = small die radius, and D = large die radius; p = small punch radius, and P = large punch radius; t = small tool gap, and T = large tool gap; f = lubrication, and F = no lubrication.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BDPtF</td>
<td>9</td>
<td>BdpTF</td>
<td>17</td>
</tr>
<tr>
<td>2</td>
<td>bDPtF</td>
<td>10</td>
<td>Bdptf</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>BDPtf</td>
<td>11</td>
<td>bdPTF</td>
<td>19</td>
</tr>
<tr>
<td>4</td>
<td>bdPtf</td>
<td>12</td>
<td>BdPtF</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>bdpTF</td>
<td>13</td>
<td>BDPTf</td>
<td>21</td>
</tr>
<tr>
<td>6</td>
<td>bDPTf</td>
<td>14</td>
<td>bDptF</td>
<td>22</td>
</tr>
<tr>
<td>7</td>
<td>bDPtF</td>
<td>15</td>
<td>BdpTF</td>
<td>23</td>
</tr>
<tr>
<td>8</td>
<td>BdPTF</td>
<td>16</td>
<td>bDptf</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Because the pilot analysis does not have any repetitions there are limitations on the inferences that can be made from the results. These restrictions are due to the lack of degrees of freedom (DoF) of the pilot study. A general rule of thumb suggests that an experiment must have between 12 and 24 degrees of freedom to be both valid and efficient. If all the higher order interactions between factors are to be investigated, then there would be only 1 degree of freedom left in the pilot study. For this reason we can only consider the influence of the direct factors and the second order interactions. Since we are reducing the number of variables in the final experiment to three, there is
little need for the higher order interactions.

<table>
<thead>
<tr>
<th>Factors and higher order interactions</th>
<th>Degrees of freedom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blank holder force</td>
<td>1</td>
</tr>
<tr>
<td>Die radii</td>
<td>1</td>
</tr>
<tr>
<td>Punch radii</td>
<td>1</td>
</tr>
<tr>
<td>Tool gap</td>
<td>1</td>
</tr>
<tr>
<td>Lubrication</td>
<td>1</td>
</tr>
<tr>
<td>Blank holder force.Die radii</td>
<td>1</td>
</tr>
<tr>
<td>Blank holder force.Punch radii</td>
<td>1</td>
</tr>
<tr>
<td>Blank holder force.Tool gap</td>
<td>1</td>
</tr>
<tr>
<td>Blank holder force.Lubrication</td>
<td>1</td>
</tr>
<tr>
<td>Die radii.Punch radii</td>
<td>1</td>
</tr>
<tr>
<td>Die radii.Tool gap</td>
<td>1</td>
</tr>
<tr>
<td>Die.radii.Lubrication</td>
<td>1</td>
</tr>
<tr>
<td>Punch radii.Tool gap</td>
<td>1</td>
</tr>
<tr>
<td>Punch radii.Lubrication</td>
<td>1</td>
</tr>
<tr>
<td>Tool gap.Lubrication</td>
<td>1</td>
</tr>
<tr>
<td>Total DoF available</td>
<td>31</td>
</tr>
<tr>
<td>Remaining DoF</td>
<td>16</td>
</tr>
</tbody>
</table>

The resulting shapes from the pilot study are initially compared manually to see if there are noticeable changes. Subsequently, the shapes are traced onto blank paper and scanned into an image file. The edge of each channel is detected using the method outlined in Section 6.4. The springback is manually calculated from the resulting scanned channel profiles.

D.3.2 Pilot study results

The results from the pilot study are presented in Tables D.4 and D.5. Two types of results have been calculated, see Figure D-1. The first set of results examines the variation of the upper and lower corner angles with respect to the vertical direction. It was found that the most significant variables for this set of results were blank holder force, die radii, and tool gap. In order of significance, the blank holder force, tool gap, punch radii, and the die radii had the most effect on the lower corner angle. The die radii and the punch radii created effects of similar magnitude on the lower corner angle. In order of significance, the combined effect of blank holder force and tool gap, the combined effect of blank holder force and die radii, the combined effect of blank holder force and punch radii had the most effect on the upper corner angle. In this case, the higher order interactions had more effect than the direct factors.
Figure D-1: This diagram shows the two desired corner angles used to create the two sets of results for the pilot study. Set 1 calculates the upper angle result $= (\theta - 90^\circ)$, and the lower angle result $= (\alpha - 90^\circ)$. Set 2 calculates the upper angle result $= (\theta - (90^\circ + \psi))$, and the lower angle result $= (\alpha - (90^\circ + \psi))$, where $\psi$ is dependent on the tool gap. In essence Set 1 measures the shape variation of the channel and Set 2 measures the springback of the channel.
The second set of results compares the corner angles to a tool gap modified channel. This type of measurement is directly proportional to the springback in the channel. It was found that the most significant variables for this set of results were blank holder force, die radii, punch radii, and tool gap. Although, punch radii and tool gap were very close in terms of effect on the corner angles. In order of significance, the blank holder force, punch radii, die radii had the most effect on the lower corner springback angle. In order of significance, the blank holder force, die radii, and tool gap had the most effect on the upper corner springback angle.

Table D.4: Table contains the results of analysing the lower corner of the “U” channels within the pilot data set. There are two sets of results. The first is based on comparing the angle between the side wall and floor (lower corner angle) with a right angle (90°) when varying the process parameters. The second set of results is based on comparing the angle between the side wall and the floor (lower corner angle) with a right angle plus the angle due to the tool gap, (90 + TG)°. This second set of data takes the effect of modifying the tool gap into account.

<table>
<thead>
<tr>
<th>Parameter Combination</th>
<th>Lower Corner, Right Angle</th>
<th>Effect</th>
<th>Sum of Squares</th>
<th>Lower Corner, Tool Gap Modified</th>
<th>Effect</th>
<th>Sum of Squares</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHF</td>
<td>-33.03</td>
<td>8700</td>
<td></td>
<td>-15.43</td>
<td>1900</td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>-10.00</td>
<td>800</td>
<td></td>
<td>-4.11</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>PR</td>
<td>11.91</td>
<td>1100</td>
<td></td>
<td>5.40</td>
<td>233</td>
<td></td>
</tr>
<tr>
<td>TG</td>
<td>21.31</td>
<td>3600</td>
<td></td>
<td>1.23</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>LB</td>
<td>-2.94</td>
<td>69</td>
<td></td>
<td>-2.30</td>
<td>42</td>
<td></td>
</tr>
<tr>
<td>BHF.DR</td>
<td>4.88</td>
<td>190</td>
<td></td>
<td>2.62</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>BHF.PR</td>
<td>-1.84</td>
<td>27</td>
<td></td>
<td>-2.42</td>
<td>47</td>
<td></td>
</tr>
<tr>
<td>BHF.TG</td>
<td>-2.38</td>
<td>45</td>
<td></td>
<td>-0.05</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>BHF.LB</td>
<td>5.19</td>
<td>220</td>
<td></td>
<td>3.22</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>DR.PR</td>
<td>1.75</td>
<td>25</td>
<td></td>
<td>0.51</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>DR.TG</td>
<td>0.59</td>
<td>3</td>
<td></td>
<td>0.14</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DR.LB</td>
<td>0.59</td>
<td>3</td>
<td></td>
<td>-0.19</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PR.TG</td>
<td>0.69</td>
<td>4</td>
<td></td>
<td>-0.41</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PR.LB</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td>-0.29</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TG.LB</td>
<td>2.03</td>
<td>33</td>
<td></td>
<td>1.03</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

In summary, the results from the pilot study indicate that the four variables that have the most effect on the shape of the channel are blank holder force, die radii, tool gap, and punch radii. The tool gap variable was chosen over the punch radii variable because the tool gap affected the shape of the channel more than the punch radii.
Table D.5: Table contains the results of analysing the upper corner of the “U” channels within the pilot data set. There are two sets of results. The first is based on the response equal to the difference between a right angle (90°) and the actual upper corner angle when varying the process parameters. The second set of results is based on the response equal to the difference between the desired angle (including the tool gap effects) and the actual upper corner. This second set of data takes the effect of modifying the tool gap into account.

<table>
<thead>
<tr>
<th>Parameter Combination</th>
<th>Upper Corner, Right Angle</th>
<th>Upper Corner, Tool Gap Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Effect</td>
<td>Sum of Squares</td>
</tr>
<tr>
<td>BHF</td>
<td>3.31</td>
<td>88</td>
</tr>
<tr>
<td>DR</td>
<td>2.75</td>
<td>61</td>
</tr>
<tr>
<td>PR</td>
<td>0.38</td>
<td>1</td>
</tr>
<tr>
<td>TG</td>
<td>-0.69</td>
<td>4</td>
</tr>
<tr>
<td>LB</td>
<td>-1.28</td>
<td>13</td>
</tr>
<tr>
<td>BHF.DR</td>
<td>6.5</td>
<td>340</td>
</tr>
<tr>
<td>BHF.PR</td>
<td>3.50</td>
<td>98</td>
</tr>
<tr>
<td>BHF.TG</td>
<td>7.88</td>
<td>500</td>
</tr>
<tr>
<td>BHF.LB</td>
<td>1.40</td>
<td>16</td>
</tr>
<tr>
<td>DR.PR</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>DR.TG</td>
<td>3.06</td>
<td>75</td>
</tr>
<tr>
<td>DR.LB</td>
<td>1.53</td>
<td>19</td>
</tr>
<tr>
<td>PR.TG</td>
<td>-0.69</td>
<td>4</td>
</tr>
<tr>
<td>PR.LB</td>
<td>0.78</td>
<td>5</td>
</tr>
<tr>
<td>TG.LB</td>
<td>0.09</td>
<td>0</td>
</tr>
</tbody>
</table>
Although, both variables did affect the springback angle quite strongly.

## D.4 Design of the main factorial experiment

Once the pilot analysis was complete and the three most significant variables were found (blank holder force, die radii, and tool gap), the main factorial experiment is designed to develop the response surface data.

A three-level, three-factor experiment was decided upon due to each factor having at least three levels. The die radii parameter has four levels and it was decided to remove the highest level. This decision was made because experimental data from Kim and Thomson (1989) corresponded to the lowest three die radii of our set-up. Thus, the final list of parameter levels is given in Table 6.1 of Chapter 6. Table D.6 contains all the combinations of a single three-level, three-factor experiment. In an ideal set-up, all these combinations would be randomised to confound systematic and stochastic errors. Unfortunately, the reset-up time between changes of die radii and tool gap was quite large. This forced the factorial experiment to be conducted in blocks of the three blank holder force levels for a given combination of die radii and tool gap. The combinations of die radii and tool gap were randomised, see Table D.7. Five sets of three-level, three-factor experiments (135 channels in total) were conducted to allow an average channel to be calculated for each of the 27 parameter combinations. This number of experiments allowed enough material for an analysis of the process noise within the channel forming set-up to be conducted.

## D.5 Noise in channel forming experimental design

This appendix also designs the set of experiments to find out more about the noise the in channel forming process. Two entirely separate sets of channels were created in addition to the previous section's data set. The two sets consist of 20 channels each for the combinations of [low blank holder force (B1), small die radii (D1), small tool gap (T1)] and [medium blank holder (B2), medium die radii (D2), medium tool gap (T2)] parameter levels respectively. Twenty channels was chosen because it gave an indication of the distribution of the noise within the channel forming process without using too much material.

## D.6 Additional experiments

Additional experiments were conducted which were not used in this thesis. A repetition of the pilot study was performed to give enough degrees of freedom to fully analyse the interactions between the five parameters. Another 20 channels were created for [high blank holder force (B3), very high die radii (D4), and high tool gap (T3)] to examine
Table D.6: Unrandomised block of tests.

<table>
<thead>
<tr>
<th>Test Number</th>
<th>Variable 1</th>
<th>Variable 2</th>
<th>Variable 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>level 1</td>
<td>level 1</td>
<td>level 1</td>
</tr>
<tr>
<td>2</td>
<td>level 2</td>
<td>level 1</td>
<td>level 1</td>
</tr>
<tr>
<td>3</td>
<td>level 3</td>
<td>level 1</td>
<td>level 1</td>
</tr>
<tr>
<td>4</td>
<td>level 1</td>
<td>level 2</td>
<td>level 1</td>
</tr>
<tr>
<td>5</td>
<td>level 2</td>
<td>level 2</td>
<td>level 1</td>
</tr>
<tr>
<td>6</td>
<td>level 3</td>
<td>level 2</td>
<td>level 1</td>
</tr>
<tr>
<td>7</td>
<td>level 1</td>
<td>level 3</td>
<td>level 1</td>
</tr>
<tr>
<td>8</td>
<td>level 2</td>
<td>level 3</td>
<td>level 1</td>
</tr>
<tr>
<td>9</td>
<td>level 3</td>
<td>level 3</td>
<td>level 1</td>
</tr>
<tr>
<td>10</td>
<td>level 1</td>
<td>level 1</td>
<td>level 2</td>
</tr>
<tr>
<td>11</td>
<td>level 2</td>
<td>level 1</td>
<td>level 2</td>
</tr>
<tr>
<td>12</td>
<td>level 3</td>
<td>level 1</td>
<td>level 2</td>
</tr>
<tr>
<td>13</td>
<td>level 1</td>
<td>level 2</td>
<td>level 2</td>
</tr>
<tr>
<td>14</td>
<td>level 2</td>
<td>level 2</td>
<td>level 2</td>
</tr>
<tr>
<td>15</td>
<td>level 3</td>
<td>level 2</td>
<td>level 2</td>
</tr>
<tr>
<td>16</td>
<td>level 1</td>
<td>level 3</td>
<td>level 2</td>
</tr>
<tr>
<td>17</td>
<td>level 2</td>
<td>level 3</td>
<td>level 2</td>
</tr>
<tr>
<td>18</td>
<td>level 3</td>
<td>level 3</td>
<td>level 2</td>
</tr>
<tr>
<td>19</td>
<td>level 1</td>
<td>level 1</td>
<td>level 3</td>
</tr>
<tr>
<td>20</td>
<td>level 2</td>
<td>level 1</td>
<td>level 3</td>
</tr>
<tr>
<td>21</td>
<td>level 3</td>
<td>level 1</td>
<td>level 3</td>
</tr>
<tr>
<td>22</td>
<td>level 1</td>
<td>level 2</td>
<td>level 3</td>
</tr>
<tr>
<td>23</td>
<td>level 2</td>
<td>level 2</td>
<td>level 3</td>
</tr>
<tr>
<td>24</td>
<td>level 3</td>
<td>level 2</td>
<td>level 3</td>
</tr>
<tr>
<td>25</td>
<td>level 1</td>
<td>level 3</td>
<td>level 3</td>
</tr>
<tr>
<td>26</td>
<td>level 2</td>
<td>level 3</td>
<td>level 3</td>
</tr>
<tr>
<td>27</td>
<td>level 3</td>
<td>level 3</td>
<td>level 3</td>
</tr>
</tbody>
</table>
**Table D.7:** This table contains the five randomised sets of the three-level, three-factor experiments. Only the die radii and tool gap parameters are randomised in blocks due to the time taken to set-up the channel forming equipment. The three blank holder force levels were tested for each die radii and tool gap combination.

<table>
<thead>
<tr>
<th>Set Number</th>
<th>Combinations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>D3T2</td>
</tr>
<tr>
<td>1</td>
<td>D1T2</td>
</tr>
<tr>
<td></td>
<td>D3T1</td>
</tr>
<tr>
<td></td>
<td>D3T3</td>
</tr>
<tr>
<td>2</td>
<td>D1T3</td>
</tr>
<tr>
<td></td>
<td>D2T1</td>
</tr>
<tr>
<td></td>
<td>D3T1</td>
</tr>
<tr>
<td>3</td>
<td>D1T1</td>
</tr>
<tr>
<td></td>
<td>D1T3</td>
</tr>
<tr>
<td>4</td>
<td>D2T3</td>
</tr>
<tr>
<td></td>
<td>D1T3</td>
</tr>
<tr>
<td></td>
<td>D3T2</td>
</tr>
<tr>
<td>5</td>
<td>D2T3</td>
</tr>
<tr>
<td></td>
<td>D3T3</td>
</tr>
<tr>
<td></td>
<td>D2T2</td>
</tr>
</tbody>
</table>
noise in the system for high blank holder force. These results were discarded because the die radii was out of the range used in the main factorial experiment.
Appendix E

Orthogonal regression

Rothwell's (1995) implementation of the orthogonal regression technique was used as a basis for the orthogonal regression technique used in this thesis.

E.1 Theory of orthogonal regression

The orthogonal regression method uses the minimized sum of squared perpendicular distances from the data points to the fitted line. Fitting is reduced to an eigenvector problem, and is done in closed form. The perpendicular distance from a point \((x_i, y_i)\) to the line \(ax + by + c = 0\) is:

\[
q_i = \frac{ax_i + by_i + c}{\sqrt{a^2 + b^2}}. \tag{E.1}
\]

If the line is normalised so that \(\sqrt{a^2 + b^2} = 1\), the fitting cost for all \(n\) data points reduces to:

\[
Q^2 = \frac{1}{n} \sum_i q_i^2 = \frac{1}{n} \sum_i (ax_i + by_i + c)^2. \tag{E.2}
\]

Writing \(\vec{v} = (a, b, c)^T\), and:

\[
S = \begin{bmatrix}
X^2 & XY & X \\
XY & Y^2 & Y \\
X & Y & n
\end{bmatrix},
\]

where \(X^2 = \sum_n x_i^2, Y^2 = \sum_n y_i^2, XY = \sum_n x_i y_i, X = \sum_n x_i\) and \(Y = \sum_n y_i\), gives:

\[
Q^2 = \frac{1}{n} \vec{v}^T S \vec{v}. \tag{E.3}
\]

The centre of mass of the point set, \((x_c, y_c)\), is at \((X/n, Y/n)\). Minimising the Lagrangian with respect to \(c\):

\[
L = \frac{1}{n} \vec{v}^T S \vec{v} - \lambda(a^2 + b^2 - 1)
\]

\[
\Rightarrow \frac{\partial L}{\partial c} = \frac{2}{n}(aX + bY + cn),
\]

\[
= 2(ax_c + by_c + c), \tag{E.4}
\]

\[
\min \left( \frac{\partial L}{\partial c} \right) = 0,
\]
and hence the centre of mass lies on the line. Translating the coordinate frame so that the centre of mass is at the origin, and fitting the line \( a_c x + b_c y + c_c = 0 \) to the data set \((x - x_c, y - y_c)\), where \( c_c = 0 \) because the line passes through the origin of the translated frame, gives the Lagrangian:

\[
L_c = \frac{1}{n} \bar{v}^T S_c \bar{v} - \lambda (a_c^2 + b_c^2 - 1),
\]

where:

\[
S_c = \begin{bmatrix}
X_c^2 & X_c Y_c & 0 \\
X_c Y_c & Y_c^2 & 0 \\
0 & 0 & n
\end{bmatrix},
\]

\( \bar{v} = (a_c, b_c, c_c)^T \). The elements of \( S_c \) are: \( X_c^2 = \sum_n (x_i - x_c)^2, Y_c^2 = \sum_n (y_i - y_c)^2, X_c Y_c = \sum_n (x_i - x_c)(y_i - y_c) \). Within the incremental algorithm the values of \( \{X, Y, X^2, XY, Y^2\} \) are stored from which the translated parameters can be computed:

\[
\begin{align*}
X_c^2 &= X^2 - X.X/n, \\
X_c Y_c &= XY - X.Y/n, \\
Y_c^2 &= Y^2 - Y.Y/n.
\end{align*}
\]

Minimising the Lagrangian with respect to \( a_c \) and \( b_c \) gives:

\[
\begin{align*}
\frac{\partial L_c}{\partial a_c} &= 2(a_c X_c^2 + b_c X_c Y_c - \lambda a_c), \\
\frac{\partial L_c}{\partial b_c} &= 2(a_c X_c Y_c + b_c Y_c^2 - \lambda b_c).
\end{align*}
\]

Setting:

\[
\frac{\partial L_c}{\partial a_c} = \frac{\partial L_c}{\partial b_c} = 0,
\]

gives:

\[
\begin{bmatrix}
X_c^2 & X_c Y_c \\
X_c Y_c & Y_c^2
\end{bmatrix}
\begin{bmatrix}
a_c \\
b_c
\end{bmatrix}
= \lambda \begin{bmatrix}
a_c \\
b_c
\end{bmatrix},
\]

which is an eigenvector problem. Taking \((a_c, b_c)^T\) corresponding to the smallest eigenvalue minimises \( Q^2 \): if the fitting cost in the translated frame is \( Q_c^2 \), and as the translation does not change the values of the distances from the line:

\[
Q^2 = Q_c^2,
\]

\[
= \frac{1}{n} \bar{v}_c^T S_c \bar{v}_c,
\]

\[
= \frac{1}{n} \bar{v}_c \begin{bmatrix}
X_c^2 & X_c Y_c \\
X_c Y_c & Y_c^2
\end{bmatrix}
\begin{bmatrix}
a_c \\
b_c
\end{bmatrix},
\]

as \( c_c = 0 \),

\[
= \frac{\lambda}{n} \bar{v}_c \begin{bmatrix}
a_c \\
b_c
\end{bmatrix},
\]

\[
= \frac{\lambda}{n} \bar{v}_c \begin{bmatrix}
a_c \\
b_c
\end{bmatrix} = 1.
\]
As the move of coordinate frame is a translation, \((a_c, b_c)^T = (a, b)^T\), and so \(\bar{v}\) has been solved for in closed form.

This outlines the orthogonal regression algorithm for \(n\) points. Fitting can be performed incrementally in the following manner (incrementally means that a line can be refit to \(n+1\) points using the result of the fit to \(n\) points without having to repeat the entire operation):

1. from the fit to \(n\) points store the values of \(\{X, Y, X^2, XY, Y^2\}\);
2. increment each of these terms by \(x_{n+1}, \ldots, y_{n+1}^2\);
3. compute the position of the centre of mass from \(X\) and \(Y\), and determine \(\{X_c^2, X_cY_c, Y_c^2\}\) from \(\{X, Y, X^2, XY, Y^2\}\);
4. perform the eigenvalue computation.

This provides an efficient algorithm for updating the fit after each extra point is extracted from an edge chain; alternatively point can be deleted if required.
Appendix F

Chain coding

The *chain code* provides a storage-efficient representation for the boundary of an object in a Boolean image. The chain code representation incorporates such pertinent information as length of the boundary of the encoded object, its area, and moments. Additionally, chain codes are invertible in that an object can be reconstructed from its chain code representation.

The basic idea behind the chain code is that each boundary pixel of an object has an adjacent pixel neighbour whose direction from the pixel boundary pixel can be specified by a unique number between 0 and 7. Given a pixel, consider its eight neighbouring pixels. Each 8-neighbour can be assigned a number from 0 to 7 representing one of eight possible directions from the given pixel (see Figure F-1). This is done with the same orientation throughout the entire image.

The extraction of boundary features requires tracing the contours of objects present in the image. There are a number of contour coding methods, but the simplest appears to be the Freeman chain code. Chain codes are used to represent a boundary by a connected sequence of straight line segments of specified length and direction. This representation is based on 8-connectivity of the segment, and the direction of each segment is coded by using a numbering scheme such as the one shown in Figure F-1. Only a starting pixel is represented by its coordinates; each other pixel on the contour is represented by a number from the set 0, 1, 2, 3, 4, 5, 6, 7. This number describes the direction to the next pixel for every connected pixel on the boundary.

The first step in chain coding is to locate a pixel on the boundary of the thresholded binary image. A search is then conducted to find the next pixel on the boundary. It is important to trace the boundary always in the same direction, either clockwise (CW) or counter-clockwise (CCW). The search direction for the next boundary pixel depends on the tracing direction, and on the previous chain code vector. Figure F-2 shows the search directions for the next pixel on the boundary. It is known from the previous chain vector that the pixels marked in white in Figure F-2 do not belong to the boundary. Thus, only the black marked pixels of the neighbouring pixels of the current boundary.
need to be checked to locate the next boundary pixel. At every boundary pixel, the neighbouring pixels (black only) are checked in the same order as the tracing direction. For example, if we are tracing a boundary in the CCW direction, and the current chain vector is 0, then searching the neighbouring pixels proceeds in the sequence 7,0,1,2,3 until a pixel with a desired grey level is found. This pixel is then declared a boundary pixel. Assuming that the boundary is closed, this method of tracing ensures that the starting pixel is always re-encountered.

Figure F-1: Diagram showing the directions from a pixel on the boundary.
Figure F-2: Abdallah’s (2000) diagram showing the search directions for the next pixel on a boundary: (a) clockwise tracking; (b) counter-clockwise tracking.
Appendix G

“U” channel forming simulation code

** File: dltl.inp
** Author: Abaqus manual
** Modified by: Bernard Rolfe
** Date: 19/4/99
** Description:
** Part of abaqus file, contains Die and Tool gap information
** Die - 3.175; Tool Gap - 1.05
** Model of standard springback U bend channel forming test.
** X - horizontal position of blank
** Y - opposite direction of punch
** Z - in the plane of the blank
** Model is axisymmetric

*HEADING
U Bend channel forming springback test
G3N steel, 2.45kN blankholder force
Shell elements ; Explicit analysis

**

**------------------------------------------------------ blank

**
** Blank is 150mm by 20mm thickness is 0.76
** -change 12/5/99 - try with a thinner width otherwise would
** have to try having several rows of nodes (about 5) across
** the width of the blank
** That is, changing z value from +- 0.010 to +- 0.002

*NODE
260 Appendix G. "U" channel forming simulation code

1, 0.000, 0, 0.010
301, 0.075, 0, 0.010
*NGEN, NSET=BLANK1
  1,301,1
*NCOPY, OLD=BLANK1,NEW=BLANK2,CHANGE=1000,SHIFT
  0.0, 0.0, -0.020

*NSET, NSET=BLANK
  BLANK1, BLANK2
*ELEMENT, TYPE=S4R
  1,1,3,1,003,1,001
*ELGEN, ELSET=BLANK
  1,150,2,2
*SHELL SECTION, ELSET=BLANK, MATERIAL=STEEL, SECTION INTEGRA=GAUSS
  0.00076,5
*MATERIAL, NAME=STEEL
*DENSITY
  7800.0
*ELASTIC
  206.E9,0.3
*PLASTIC
  .15403E+09,.00000E+00
  .19410E+09,.10000E-01
  .21913E+09,.20000E-01
  .23803E+09,.30000E-01
  .25348E+09,.40000E-01
  .26668E+09,.50000E-01
  .27826E+09,.60000E-01
  .28864E+09,.70000E-01
  .29807E+09,.80000E-01
  .30674E+09,.90000E-01
  .31477E+09,1.00000E+00
  .32226E+09,1.10000E+00
  .32930E+09,1.20000E+00
  .33594E+09,1.30000E+00
  .34224E+09,1.40000E+00
  .34822E+09,1.50000E+00
  .35393E+09,1.60000E+00
  .35940E+09,1.70000E+00
  .36464E+09,1.80000E+00
.36968E+09, .19000E+00
.37454E+09, .20000E+00
.37922E+09, .21000E+00
.38375E+09, .22000E+00
.38814E+09, .23000E+00
.39239E+09, .24000E+00
.39652E+09, .25000E+00
.40053E+09, .26000E+00
.40443E+09, .27000E+00
.40822E+09, .28000E+00
.41193E+09, .29000E+00
.41554E+09, .30000E+00
.41906E+09, .31000E+00
.42251E+09, .32000E+00
.42588E+09, .33000E+00
.42917E+09, .34000E+00
.43240E+09, .35000E+00
.43556E+09, .36000E+00
.43866E+09, .37000E+00
.44169E+09, .38000E+00
.44468E+09, .39000E+00
.44760E+09, .40000E+00
.45047E+09, .41000E+00
.45330E+09, .42000E+00
.45607E+09, .43000E+00
.45880E+09, .44000E+00
.46148E+09, .45000E+00
.46413E+09, .46000E+00
.46672E+09, .47000E+00
.46928E+09, .48000E+00
.47181E+09, .49000E+00
.47429E+09, .50000E+00

*POTENTIAL
1.0, 1.24897, 1.0402, 1.0, 1.07895, 1.0

**

**-------------------------------- right die
**

*NODE
2000, 0.02105, -0.0320, 0.020
*NODE, NSET=DIE
Appendix G. "U" channel forming simulation code

2001, 0.02105, -0.0320, 0.020
2002, 0.02105, -0.003555, 0.020
2008, 0.024225, -0.00038, 0.020
2009, 0.080, -0.00038, 0.020
*NGEN, NSET=DIE, LINE=C
2002, 2008, 1,, 0.024225, -0.003555, 0.020
*NCOPY, CHANGE=100, OLD=DIE, NEW=DIE, SHIFT
0.0, 0.0, -0.040

*ELEMENT, TYPE=R3D4
*ELGEN, ELSET=DIE
2001, 8, 1, 1
*RIGID BODY, ELSET=DIE, REF NODE=2000**
**----------------------------------------------------------------- punch**
**
*NODE
4000, 0.000, 0.00038, 0.020
*NODE, NSET=PUNCH
4001, 0.000, 0.00038, 0.020
4002, 0.016825, 0.00038, 0.020
4008, 0.020, 0.003555, 0.020
4009, 0.020, 0.040, 0.020
*NGEN, NSET=PUNCH, LINE=C
4002, 4008, 1,, 0.016825, 0.003555, 0.020
*NCOPY, CHANGE=100, OLD=PUNCH, NEW=PUNCH, SHIFT
0.0, 0.0, -0.040

*ELEMENT, TYPE=R3D4
4001, 4001, 4002, 4102, 4101
*ELGEN, ELSET=PUNCH
4001, 8, 1
*RIGID BODY, ELSET=PUNCH, REF NODE=4000
**
**----------------------------------------------------------------- right holder**
**
*NODE, NSET=HOLDER
5001, 0.030, 0.010, 0.020
5002, 0.030, 0.00038, 0.020
5003, 0.075, 0.00038, 0.020
Appendix G. "U" channel forming simulation code

5004, 0.075, 0.010, 0.020
*NODE
5005, 0.0555, 0.005, 0.000
*NCOPY, OLD=HOLDER, NEW=HOLDER, CHANGE=100, SHIFT
0.0, 0.0, -0.040

*ELEMENT, TYPE=R3D4
5001, 5001, 5002, 5102, 5101
*ELGEN, ELSET=HOLDER
5001, 3, 1,
*RIGID BODY, ELSET=HOLDER, REF NODE=5005
*ELEMENT, TYPE=MASS, ELSET=EM1
5100, 5005
*MASS, ELSET=EM1
2.5
**
**----------------------------------------- boundary conditions
**
*BOUNDARY
BLANK, 3,5
1, 1,1
1, 6,6
1001, 1,1
1001, 6,6
2000, 1,6
4000, 1,1
4000, 3,6
5005, 1,1
5005, 3,6
**
*RESTART, WRITE, NUMBER INT=1
**
*AMPLITUDE,NAME=APUNCH
0.,0., 0.00199999999, 1., 0.00399999999, 0.
*AMPLITUDE,NAME=RAMP
0.0, 0.0, 0.00001, 1.0
**
**----------------------------------------- analysis
**
**
apply blankholder force

*STEP
*DYNAMIC, EXPLICIT
,0.00001
*CLOAD, AMP=RAMP
 5005, 2, -3300.0

**
** Define surfaces
**
*SURFACE DEF, NAME=PUNCH
  PUNCH,S2
*SURFACE DEF, NAME=HOLDER
  HOLDER,S2
*SURFACE DEF, NAME=DIE
  DIE,S1
*SURFACE DEF, NAME=TOP
  BLANK,S1
*SURFACE DEF, NAME=BOTTOM
  BLANK,S2

**
** Define surface interaction
**
*SURFACE INTERACTION, NAME=ALLCONT
*FRICION
  0.5
*CONTACT PAIR, INTERACTION=ALLCONT
  PUNCH, TOP
*CONTACT PAIR, INTERACTION=ALLCONT
  HOLDER, TOP
*CONTACT PAIR, INTERACTION=ALLCONT
  DIE, BOTTOM

**
*FILE OUTPUT, TIMEMARKS=YES, NUM=1
*NSET, NSET=BH
  BLANK, HOLDER
*ELSET, ELSET=BH
  BLANK, HOLDER
*NODE FILE, NSET=BH
  U, V
 Appendix G. "U" channel forming simulation code

*EL FILE, ELSET=BH
  S, PEEQ, MISES
  STH,
  *HISTORY OUTPUT, TIME=0.0
  *ENERGY HISTORY
  ALLKE, ALLIE, ALLSE, ALLAE, ALLVD, ALLCD, ALLPD, ALLFD, ALLWK, ETOTAL, DT
  *END STEP
  **
  **------------------------------------------ move punch down
  **
  *STEP
  *DYNAMIC, EXPLICIT
  , 0.00399999999
  *BOUNDARY, AMP=APUNCH, TYPE=VELOCITY
  4000, 2, 2, -15.0
  **
  *RESTART, WRITE, NUMBER INT=1
  *HISTORY OUTPUT, TIME=0.0
  *ENERGY HISTORY
  ALLKE, ALLSE, ALLPD, ALLIE, ALLWK, ETOTAL, ALLFD, ALLAE, ALLVD, ALLCD, DT
  *END STEP