Selective laser sintering of hydroxyapatite-polyamide composites

This item was submitted to Loughborough University’s Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/7907

Publisher: © Monica M. Savalani

Please cite the published version.
This item is held in Loughborough University’s Institutional Repository (https://dspace.lboro.ac.uk/) and was harvested from the British Library’s EThOS service (http://www.ethos.bl.uk/). It is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/
A CASE-BASED REASONING METHODOLOGY TO FORMULATING POLYURETHANES

A Doctoral Thesis Submitted in Partial Fulfilment of the Requirements for the Award of Doctor of Philosophy of Loughborough University

by
Diana Marcela Segura Velandia

Wolfson School of Mechanical and Manufacturing Engineering
Loughborough University
United Kingdom
July, 2006
Abstract

Formulation of polyurethanes is a complex problem poorly understood as it has developed more as an art rather than a science. Only a few experts have mastered polyurethane (PU) formulation after years of experience and the major raw material manufacturers largely hold such expertise. Understanding of PU formulation is at present insufficient to be developed from first principles. The first principle approach requires time and a detailed understanding of the underlying principles that govern the formulation process (e.g. PU chemistry, kinetics) and a number of measurements of process conditions. Even in the simplest formulations, there are more than 20 variables often interacting with each other in very intricate ways. In this doctoral thesis the use of the Case-Based Reasoning and Artificial Neural Network paradigm is proposed to enable support for PUs formulation tasks by providing a framework for the collection, structure, and representation of real formulating knowledge. The framework is also aimed at facilitating the sharing and deployment of solutions in a consistent and referable way, when appropriate, for future problem solving.

Two basic problems in the development of a Case-Based Reasoning tool that uses past flexible PU foam formulation recipes or cases to solve new problems were studied. A PU case was divided into a problem description (i.e. PU measured mechanical properties) and a solution description (i.e. the ingredients and their quantities to produce a PU). The problems investigated are related to the retrieval of former PU cases that are similar to a new problem description, and the adaptation of the retrieved case to meet the problem constraints.

For retrieval, an alternative similarity measure based on the moment’s description of a case when it is represented as a two dimensional image was studied. The retrieval using geometric, central and Legendre moments was also studied and compared with a standard nearest neighbour algorithm using nine different distance functions (e.g. Euclidean, Canberra, City Block, among others). It was concluded that when cases were represented as 2D images and matching is performed by using moment functions in a similar fashion to the approaches studied in image analysis in pattern recognition, low order geometric and Legendre moments and central moments of any order retrieve the same case as the Euclidean distance does when used in a nearest neighbour algorithm. This means that the Euclidean distance acts a low moment function that represents gross level case features. Higher order (moment’s order>3) geometric and Legendre moments while enabling finer details about an image to be represented had no standard distance function counterpart.

For the adaptation of retrieved cases, a feed-forward back-propagation artificial neural network was proposed to reduce the adaptation knowledge acquisition effort that has prevented building complete CBR systems and to generate a mapping between change in mechanical properties and formulation ingredients. The proposed network was trained with the differences between problem descriptions (i.e. mechanical properties of a pair of foams) as input patterns and the differences between solution descriptions (i.e. formulation ingredients) as the output patterns. A complete data set was used based on 34 initial formulations and a 16950 epochs trained network with 1102 training exemplars, produced from the case differences, gave only 4% error. However, further work with a data set consisting of a training set and a small validation set failed to generalise returning a high percentage of errors. Further tests on different training/test splits of the data also failed to generalise. The conclusion reached is that the data as such has insufficient common structure to form any general conclusions. Other evidence to suggest that the data does not contain generalisable structure includes the large number of hidden nodes necessary to achieve convergence on the complete data set.

Keywords: Polyurethane Foam, Product Formulation, Case-Based Reasoning, Retrieval Algorithm, Moment Functions, Artificial Neural Network, Knowledge-Based Systems
To my beloved mum, Hilda

For her example, love and prayers
Acknowledgements

"Nobody can carry anyone else on his shoulders to the final goal. At most, with love and compassion one can say, ‘Well, this is the path, and this is how I have walked on it. You also work, you also walk, and you will reach the final goal’. But each person has to walk himself, has to take every step on the path himself. He who has taken one step on the path is one step nearer the goal. He who has taken a hundred steps is a hundred steps nearer the goal. He who has taken all the steps on the path has reached the final goal. You have to walk on the path yourself”.

Early Buddhist

Marie desJardins, in her guide to graduate students and advisors [17] wrote:

“A good advisor will serve as a mentor as well as a source of technical assistance. A mentor should provide, or help you to find, the resources you need (financial, equipment, and psychological support); introduce you and promote your work to important people in your field; encourage your own interests, rather than promoting their own; be available to give you advice on the direction of your thesis and your career; and help you to find a job when you finish”.

I was lucky to get all of the above as these desiderata were fulfilled not by a single advisor, but by four advisors, Andrew Nurse, Dick Heath, Chris Hinde and Andy West.

I would like to thank my first supervisor, Andrew Nurse. Andrew’s interests were amazingly broad, he was a mind opener. He gave me the freedom to pursue my own interests and when he could, provided me with the resources I needed. He always encouraged me to talk to and visit other people on conferences and during visits abroad. For this, I am very grateful.

Dick Heath and Chris Hinde were my technical advisors. Dick introduced me to the scientific rather than artistic polyurethane formulation and chemistry and helped me to find collaborators in industry. We spent a lot of time together trying to figure out what the best way of tackling polyurethane formulation problems and Dick’s meticulous approach has been an example for me.

Chris’ thought provoking personality made me realise that knowledge is best acquired by trying. He painstakingly and repeatedly explained me how the basics of neural networks work
and how I could make the best of using them for the project I had in mind. I have exceeded his friendliness enormously and I cannot thank him enough for the wonderful time I had, both scientifically as well as personally.

Without doubt, the person who has had the most significant impact on my life at Loughborough is Andy West. He has helped in many ways and for this I am very grateful. He introduced me to Chris and supported and encouraged me when I most needed it. It has been an honour to have him as mentor and friend and a constant source of inspiration and perspective, thank you.

I owe also much to those that have been with me from the beginning. My colleagues at the PUR centre of excellence, Becky Phelps, Alastair McCourt, and Dave Britton. They were very helpful and supportive during the initial stages of my work and I hope our paths will not part entirely! Furthermore, I must mention that the ‘tea time’ moments with most friends and colleagues (too many to name you all guys) in the common room were quite inspiring.

A rather special “tanka” to Alejandro. Ale is the perfect example of an altruistic person. He has always helped me many times and in different ways. Most importantly I am grateful for his love that encouraged me to walk on the path myself. To you I will always be grateful.

In the final phase of my work, the members of the examination committee, Dr Neil Hopkinson and Professor Kei Cheng, have given me good advice and constructive criticism. This surely improved the thesis in many ways.

Most importantly however, I would like to say to my beloved Star how much he means to me. When I was writing up my thesis, he was a steadfast rock to me even though he had his own worries. He always reminded me to remain calm and keep my chin up because everything will be fine. This work shows, once more, that he is entirely right. Gracias amorsote!

Por ultimo, la visión, inteligencia y el amor incondicional de mi mamá siempre han estado allí cuando nada más ha estado. Gracias mamí por consentirme tanto; yo no sería lo que soy sino fuera por ti.
List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>MOCA</td>
<td>4,4’ Methylene Bis (2-Chloroaniline)</td>
</tr>
<tr>
<td>MDI</td>
<td>4,4’-Diphenylmethane Diisocyanate</td>
</tr>
<tr>
<td>ASTM</td>
<td>American Society For Testing And Materials</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial Neural Networks</td>
</tr>
<tr>
<td>TODI</td>
<td>Bi-tolylene Diisocyanate</td>
</tr>
<tr>
<td>BIFMA</td>
<td>Business And Institutional Furniture Manufacture Association</td>
</tr>
<tr>
<td>CB</td>
<td>Case Base</td>
</tr>
<tr>
<td>CBR</td>
<td>Case-Based Reasoning</td>
</tr>
<tr>
<td>CFCs</td>
<td>Chlorofluorocarbons</td>
</tr>
<tr>
<td>DM</td>
<td>Data Mining</td>
</tr>
<tr>
<td>DNA</td>
<td>Deoxyribonucleic Acid</td>
</tr>
<tr>
<td>DMDI</td>
<td>Dimethylidiphenylmethane Diisocyanate</td>
</tr>
<tr>
<td>PDI</td>
<td>Diphenylisopropylidene</td>
</tr>
<tr>
<td>DMDI</td>
<td>Dymethyl-Dyphenilmethane</td>
</tr>
<tr>
<td>ES</td>
<td>Expert Systems</td>
</tr>
<tr>
<td>FPF</td>
<td>Flexible Polyurethane Foams</td>
</tr>
<tr>
<td>FL</td>
<td>Fuzzy Logic</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>Tg</td>
<td>Glass Transition Temperature</td>
</tr>
<tr>
<td>HVDM</td>
<td>Heterogeneous Value Difference Metric</td>
</tr>
<tr>
<td>HCFC</td>
<td>Hydrochlorofluorocarbons</td>
</tr>
<tr>
<td>IFD</td>
<td>Indentation Force Deflection</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standards Organisation</td>
</tr>
<tr>
<td>IVDM</td>
<td>Interpolated Value Difference Metric</td>
</tr>
<tr>
<td>KE</td>
<td>Knowledge Engineer</td>
</tr>
<tr>
<td>KM</td>
<td>Knowledge Management</td>
</tr>
<tr>
<td>KBS</td>
<td>Knowledge-Based System</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>TRAINLM</td>
<td>Levenberg-Marquardt Back-Propagation Algorithm</td>
</tr>
<tr>
<td>ML</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>MPa</td>
<td>Mega Pascals</td>
</tr>
<tr>
<td>NDI</td>
<td>Naphthalene Diisocyanate</td>
</tr>
<tr>
<td>ODS</td>
<td>Ozone Depleting Substances</td>
</tr>
<tr>
<td>PF</td>
<td>Phenol-Formaldehyde</td>
</tr>
<tr>
<td>PDCPD</td>
<td>Polydicyclopentadiene</td>
</tr>
<tr>
<td>PAPI</td>
<td>Polymethylene Polyphenyl Isocyanate</td>
</tr>
<tr>
<td>PTMG</td>
<td>Polytetramethylene Glycol (Polyether Polyol)</td>
</tr>
<tr>
<td>PU, PUR</td>
<td>Polyurethane</td>
</tr>
<tr>
<td>PPDI</td>
<td>p-Phenylene Diisocyanate</td>
</tr>
<tr>
<td>PUDs</td>
<td>PU Dispersions</td>
</tr>
<tr>
<td>RIM</td>
<td>Reaction Injection Moulding</td>
</tr>
<tr>
<td>RTM</td>
<td>Resin Transfer Moulding</td>
</tr>
<tr>
<td>RBS</td>
<td>Rule-Based Systems</td>
</tr>
<tr>
<td>SPR</td>
<td>Structure-Property Relationship</td>
</tr>
<tr>
<td>TANSIG</td>
<td>Tangent Sigmoid Transfer Function</td>
</tr>
<tr>
<td>TDI</td>
<td>Toluene Diisocyanate</td>
</tr>
<tr>
<td>2,4-TDI</td>
<td>Toluene-2,4-Diisocyanate</td>
</tr>
<tr>
<td>2,6-TDI</td>
<td>Toluene-2,6-Diisocyanate</td>
</tr>
<tr>
<td>UNEP</td>
<td>United Nations Environment Programme</td>
</tr>
<tr>
<td>UPRs</td>
<td>Unsaturated Polyester Resins</td>
</tr>
<tr>
<td>WVDM</td>
<td>Windowed Value Difference Metric</td>
</tr>
</tbody>
</table>
## Contents

Acknowledgements ................................. v
Publications originated from this work ........ xviii

1. Chapter 1. Problem Definition and Research Objectives .......................... 1
   1.1 PROBLEM STATEMENT ................................. 1
   1.2 OBJECTIVES AND RESEARCH QUESTIONS .......... 2
   1.3 EXPERIMENTAL SETUP ................................. 4
   1.3.1 Research Question 1 ..................................... 4
   1.3.2 Research Question 2 ..................................... 4
   1.3.3 Research Question 3 ..................................... 5
   1.4 A ROAD MAP TO THIS THESIS .......................... 5

2. Chapter 2. Polyurethane Formulation .............................................. 12
   2.1 OBJECTIVE OF THIS CHAPTER .......................... 12
   2.2 WHAT IS A POLYURETHANE FORMULATION? .......... 13
   2.3 FORMULATING PUS ........................................... 17
   2.3.1 PU Materials’ Specifications .......................... 17
   2.3.2 The Formulation Process ................................. 23
   2.4 APPROACHES TO FORMULATION ......................... 27
   2.4.1 PU Processing Variations ............................... 27
   2.4.2 Reactant and Additive Variation ....................... 28
   2.4.2.1 Polyisocyanates ................................. 30
   2.4.2.2 Polylols ........................................... 32
   2.4.2.3 Chain Extenders ................................... 34
   2.5 SUMMARY AND CONCLUSIONS ........................... 36
   2.6 MOTIVATION FOR A COMPUTATIONALLY-BASED APPROACH 36

   3.1 PROBLEM STATEMENT AND OBJECTIVE OF THIS CHAPTER 39
   3.1.1 Research Question 1 ..................................... 40
   3.2 METHODOLOGY FOR RESEARCH QUESTION 1 .......... 40
   3.3 WHAT IS ARTIFICIAL INTELLIGENCE? .................... 40
   3.3.1 AI History ........................................... 41
   3.3.2 Expert Reasoning and Expert Systems ................ 41
   3.3.2.1 Rule-Based Systems .................................. 42
   3.3.2.2 Case-Based Reasoning (CBR) ....................... 43
   3.3.3 Choice of the Computational Methodology ............ 47
   3.4 CASE-BASED REASONING .................................. 49
   3.4.1 What is CBR? ........................................... 49
   3.4.2 How to Represent Knowledge in CBR? ................. 50
   3.4.2.1 Cases ........................................... 51
   3.4.2.2 Other Domain Knowledge ............................ 52
### CBR’s Knowledge Representation

- **3.4.2.3** How to Retrieve Similar Cases? .......................................................... 55
  - **3.4.3.1** The retrieval algorithm ................................. 57
  - **3.4.3.2** Selected measures of similarity ................. 61
  - **3.4.3.3** General Considerations for Retrieval ............ 63
  - **3.4.3.4** Other Retrieval Approaches .......................... 65

### Summary

- **3.4.4** Summary ...................................................................................................... 66

### CASE-BASED REASONERS FOR PRODUCT FORMULATION

- **3.5** Case-Based Reasoners for Product Formulation 67
  - **3.5.1** Dish Planning......................................................................................... 68
  - **3.5.2** Drug Tablet Formulation ....................................................................... 70
  - **3.5.3** Rubber Compounding .......................................................................... 73
  - **3.5.4** Others CBR Systems in Manufacturing .............................................. 75
    - **3.5.4.1** Steel Industry ..................................................................................... 75
    - **3.5.4.2** P-Truck Project .................................................................................... 77
    - **3.5.4.3** Cast Refractory Manufacture ............................................................. 77
    - **3.5.4.4** Polymer Injection Moulding ............................................................... 78
    - **3.5.4.5** Process Control .................................................................................. 79
    - **3.5.4.6** Aluminium Alloys ............................................................................. 80

### SUMMARY AND CONCLUSIONS

- **3.6** Summary and Conclusions .......................................................................... 81

### Chapter 4. Moment Feature Vector Matching

- **4.** Chapter 4. Moment Feature Vector Matching 85
  - **4.1** Introduction ................................................................................................. 85
  - **4.2** Problem Statement .................................................................................... 86
  - **4.3** Objectives and Research Questions ......................................................... 87
    - **4.3.1** Research Question 2 .............................................................................. 87
  - **4.4** Methodology and Experimental Setup ...................................................... 88
    - **4.4.1** Introduction ............................................................................................. 88
    - **4.4.2** Case Study 1. Water-Blown Integral Self-Skinning Flexible Polyurethane Foam Formulation .......................................................... 90
      - **4.4.2.1** Experimental Statistical Design ...................................................... 90
      - **4.4.2.2** Production of Test Foams in the Laboratory ..................................... 91
      - **4.4.2.3** Measurement of the Foams’ Physical and Mechanical Properties ... 93
    - **4.4.3** Case Study 2. Combustion Modified High Resilient Foam ................. 95
      - **4.4.3.1** Experimental Statistical Design ...................................................... 95
      - **4.4.3.2** Experimental Procedure ................................................................. 96
      - **4.4.3.3** Measurement of the Foams’ Physical and Mechanical Properties ... 97
    - **4.4.4** Methodology for Sub-question A: Local-Based Similarity Retrieval .... 98
      - **4.4.4.1** Sample: Attribute-Valued Case-Bases ........................................... 98
      - **4.4.4.2** Preparing the Input Data ................................................................. 100
      - **4.4.4.3** The Retrieval Algorithm ................................................................. 101
    - **4.4.5** Methodology for Sub question B: Global-Based Similarity .......... 103
      - **4.4.5.1** Representation of Cases using Moment Functions ....................... 104
      - **4.4.5.2** Moment Functions ........................................................................... 106
      - **4.4.5.3** Case Retrieval based on Moment Functions .................................... 110
    - **4.4.6** Analysis of the Results ......................................................................... 113
    - **4.4.7** Summary ............................................................................................... 115
  - **4.5** Results and Discussion ............................................................................. 118
    - **4.5.1** Retrieval Using Equal Feature Weights .......................................... 123
      - **4.5.1.1** Local retrieval .................................................................................. 123
      - **4.5.1.2** Global retrieval ............................................................................... 128
    - **4.5.2** Retrieval Using Expert’s Selected Feature Weights ......................... 136
      - **4.5.2.1** Local retrieval .................................................................................. 136
      - **4.5.2.2** Global retrieval ............................................................................... 142
    - **4.5.3** Comparison of Local and Global Retrieval ...................................... 149
  - **4.6** Summary and Conclusions ....................................................................... 153
4.6.1 Local Retrieval ................................................................................................ 154
4.6.2 Global Retrieval .............................................................................................. 155

5. Chapter 5. CBR Adaptation .................................................................................. 158
   5.1 INTRODUCTION .................................................................................................. 158
   5.2 PROBLEM STATEMENT.................................................................................... 159
   5.3 OBJECTIVES AND RESEARCH QUESTIONS .................................................. 160
      5.3.1 Research Question 3 .................................................................................... 160
   5.4 METHODOLOGY ................................................................................................ 161
   5.5 ADAPTATION USING THE CASE-BASE: RELATED WORK .............................. 162
   5.6 ARTIFICIAL NEURAL NETWORK APPLICATIONS WITHIN THE FORMULATION
       DOMAIN .............................................................................................................. 164
      5.6.1 Multi-Layer Networks.................................................................................. 168
         5.6.1.1 Learning and Generalisation .................................................................. 168
      5.7 IMPLEMENTATION OF AN ARTIFICIAL NEURAL NETWORK TO GUIDE CBR
       ADAPTATION ............................................................................................................ 170
         5.7.1 Methodology and Experimental Setup....................................................... 170
            5.7.1.1 Pre-treatment Task .............................................................................. 172
            5.7.1.2 Algorithm in the MATLAB Artificial Neural Network Toolbox .......................... 180
            5.7.1.3 Algorithm in GHOST ........................................................................... 183
      5.8 USING AN ARTIFICIAL NEURAL NETWORK TO ADAPT CASES ...................... 184
      5.9 RESULTS AND DISCUSSION ......................................................................... 187
         5.9.1 Artificial Neural Network Implementation ............................................... 188
            5.9.1.1 Real-valued outputs ............................................................................ 188
            5.9.1.2 Binary-valued output results ............................................................... 189
            5.9.1.3 Training with CBR retrieved differences .............................................. 191
         5.9.2 Adaptation using the Artificial Neural Network ....................................... 191
      5.10 SUMMARY ...................................................................................................... 196
      5.11 CONCLUSION .................................................................................................. 197
         5.11.1 Adaptation Using the Case-Base: Sub-question A .................................... 197
         5.11.2 Using Artificial Neural Networks as Inductive Technique: Sub-question B 198
         5.11.3 An Artificial Neural Network Approach for Adaptation of Cases: Sub-
             question C ......................................................................................................... 198

6. Chapter 6. Conclusions and Future Work .................................................................. 202
   6.1 CONCLUSIONS ABOUT RESEARCH QUESTIONS ........................................ 203
      6.1.1 Research Question 1 .................................................................................... 203
          6.1.2 Research Question 2 .................................................................................. 203
             6.1.2.1 Sub-question A .................................................................................... 203
             6.1.2.2 Sub-question B .................................................................................... 204
      6.1.3 Research Question 3 .................................................................................... 206
          6.1.3.1 Sub-question A .................................................................................... 206
          6.1.3.2 Sub-question B .................................................................................... 206
          6.1.3.3 Sub-question C .................................................................................... 207
      6.2 MAIN CONTRIBUTIONS OF THIS THESIS ...................................................... 208
      6.3 FUTURE WORK ................................................................................................. 209
          6.3.1 Application Domain ...................................................................................... 209
             6.3.1.1 Industrial implementation and validation ................................................. 209
             6.3.1.2 Greater scope ....................................................................................... 210
          6.3.2 Case-Based Reasoning-Artificial Neural Network System ......................... 210
             6.3.2.1 Experimental Validation of the Retrieval Algorithm .................................. 210
             6.3.2.2 The Use of Inductive techniques ............................................................ 211
             6.3.2.3 Producing Evidence for the Hypotheses of the Artificial Neural Networks .... 211
             6.3.2.4 Other Network Architectures ................................................................. 211
6.3.2.5 Extended Capabilities

References 213

Appendix I. Manufacturing of Foams in the laboratory 222
List of Figures

1-1. DEFINITION OF THE RESEARCH PROBLEM STUDIED IN THIS THESIS ......................... 3
1-2. MAIN AIM IN THIS THESIS. ....................................................................................... 11
2-1. FORMULATION EXAMPLE FOR FLEXIBLE PU FOAM ........................................ 14
2-2. PU FORMULATION PROCESS ................................................................................ 23
2-3. INTERRELATION BETWEEN PU’S LAYERS: COMPOSITION STRUCTURE AND THE MANUFACTURING PROCESS THAT INFLUENCE THE DESIGN OF NEW FORMULATIONS. LAYOUT INSPIRED BY THE WORK OF [72]. ................................................................. 26
3-1. CASE-BASED REASONING. SOURCE: [94] ............................................................... 42
3-2. THE BASIC FEATURES OF A FORWARD CHAINING RBS. SOURCE: [95] ............ 43
3-3. CASE-BASED REASONING AS A PROBLEM-SOLVING METHODOLOGY ................. 45
3-4. A TASK METHOD DECOMPOSITION OF CBR. SOURCE: [101] ............................... 46
3-5. THE KINDS OF DOMAIN KNOWLEDGE THAT CAN GO INTO A CBR SYSTEM’S KNOWLEDGE BASE. ADAPTED FROM [111] ............................................. 51
3-6. ONE OF CHEF’S CASES: BROCCOLI WITH TOFU. TAKEN FROM [28 P. 172] .......... 52
3-7. FEATURE VECTOR REPRESENTATION OF CASES. .................................................... 54
3-8. THE PROCESS OF RETRIEVING CASES FROM THE CASE BASE ......................... 56
3-9. CASE’S SIMILARITY ASSESSMENT BASED ON NUMERICAL COMPUTATION .......... 57
3-10. DIFFERENT WAYS TO REPRESENT SIMILARITY. SOURCE: [129] ....................... 61
3-11. EFFECT OF VARIABLE’S MAGNITUDE ON THE EUCLIDEAN DISTANCE ............... 64
3-12. GLOBAL AND LOCAL WEIGHTING. ADAPTED FROM [135] ................................. 65
3-13. CREATING “BEEF AND BROCCOLI” USING CHEF. SOURCE: [112] .................... 69
3-14. FACILITIES IN RECALL USED IN CBRTFS. SOURCE: [118] ............................... 71
3-15. THE P-RACE SYSTEM. SOURCE: [152] ................................................................. 74
3-17. ADAPTATION ALGORITHM USED IN THE DIAGNOSTICS CBR SYSTEM KBPCS. SOURCE: [149] ................................................................. 80
4-1. RESEARCH QUESTION FOR THE CHAPTER 4 ......................................................... 88
4-2. CASE STUDIES. ....................................................................................................... 90
4-3. EXPERIMENTAL PROCEDURE DESCRIPTION TO PRODUCE MOULDED PU FOAM 93
4-4. SELF-SKINNING PU FOAM DENSITY MEASUREMENT ........................................... 94
4-5. PU FOAM TESTING ................................................................................................. 95
4-6. A PU CASE HAS TWO MAIN PARTS, THE PROBLEM AND THE SOLUTION DESCRIPTIONS 100
4-6. RETRIEVAL ALGORITHM USING STANDARD DISTANCE FUNCTIONS .................................................................................................................. 101
4-7. EXAMPLE OF RETRIEVAL USING FEATURE WEIGHTS ................................................................................................................................. 103
4-8. DNA REPRESENTATION OF PU CASES ...................................................................................................................................................... 105
4-9. CODIFICATION OF CASES AS GENES ....................................................................................................................................................... 106
4-10. CENTRAL MOMENTS APPROXIMATION OF IMAGE FUNCTIONS .............................................................................................................. 109
4-11. FEATURE VECTOR REPRESENTATION OF IMAGES (GENES) .................................................................................................................. 110
4-12. PSEUDO-CODE FOR THE RETRIEVAL ALGORITHM FOR MATCHING CASES REPRESENTED AS TWO-DIMENSIONAL IMAGES OR GENES ................................................................................................................................................. 113
4-13. METHODOLOGY TO RETRIEVE CASES USING THE EUCLIDEAN DISTANCE .................................................................................................. 116
4-14. METHODOLOGY TO RETRIEVE CASES USING GEOMETRIC MOMENTS .............................................................................................. 117
4-15. RETRIEVAL USING STANDARD DISTANCE FUNCTIONS FOR CB1 ........................................................................................................ 123
4-16. K-MEANS CLUSTER ANALYSIS WITH K=2 (THAT IS SEEKING TWO CLUSTERS) FOR THE DISTANCE FUNCTIONS USED IN RETRIEVAL OF CASES FROM CB1 ............................................................................................................. 124
4-17. K-MEANS CLUSTER ANALYSIS WITH THREE CLUSTERS FOR THE DISTANCE FUNCTIONS USED IN RETRIEVAL OF CASES FROM CB1 WHEN USING EQUAL FEATURE WEIGHTS .............................................................................................................................. 125
4-18. RETRIEVAL USING STANDARD DISTANCE FUNCTIONS FOR CB2 ........................................................................................................ 126
4-19. K-MEANS CLUSTER ANALYSIS WITH K=2, CB2 ......................................................................................................................................... 127
4-20. K-MEANS CLUSTER ANALYSIS WITH K=3 FOR THE DISTANCE FUNCTIONS USED IN RETRIEVAL OF CASES FROM CB2 WHEN USING EQUAL FEATURE WEIGHTS .............................................................................................................................. 128
4-21. RETRIEVED CASE FOR 18 QUERIES BY USING LOW AND HIGH ORDER MOMENTS ........................................................................................................ 130
4-22. RETRIEVED CASE FOR 34 QUERIES BY USING LOW AND HIGH ORDER MOMENTS ........................................................................................................ 130
4-23. COMPARISON OF LOW AND HIGH ORDER MOMENTS FOR RETRIEVAL IN CB1 ........................................................................................................ 131
4-24. COMPARISON OF LOW AND HIGH ORDER MOMENTS FOR RETRIEVAL IN CB2 ........................................................................................................ 131
4-25. K-MEANS CLUSTER ANALYSIS WITH K=2 FOR THE MOMENT-BASED RETRIEVAL FUNCTIONS FOR CB1 WHEN USING EQUAL FEATURE WEIGHTS .............................................................................................................................. 132
4-26. K-MEANS CLUSTER ANALYSIS WITH K=2 FOR THE MOMENT-BASED RETRIEVAL FUNCTIONS USING EQUAL FEATURE WEIGHTS FOR CB2 .............................................................................................................................................. 133
4-27. GLOBAL RETRIEVAL VISUALISATION ...................................................................................................................................................... 135
4-28. LOCAL RETRIEVAL VISUALISATION. BASED ON [162]. ....................................................................................................................................... 136
4-29. K-MEANS CLUSTER ANALYSIS WITH K=2 FOR THE DISTANCE FUNCTIONS USED IN RETRIEVAL OF CASES FROM CB1 WHEN USING EXPERT FEATURE WEIGHTS ......................................................................................................................... 138
4-30. MEMBERS OF EACH CLUSTER AND DISTANCES FROM RESPECTIVE CLUSTER CENTRE .................................................................................................................................................................................................................. 138
4-31. K-MEANS CLUSTER ANALYSIS WITH K=3 FOR THE DISTANCE FUNCTIONS USED IN RETRIEVAL OF CASES FROM CB1 WHEN USING EXPERT FEATURE WEIGHTS .................................................................................................................................................................................................................. 139
4-32. COMPARISON BETWEEN EUCLIDEAN, D7 AND D10 DISTANCE FUNCTIONS ..................................................................................................... 140
4-33. K-MEANS CLUSTER ANALYSIS WITH K=2, CB2 ......................................................................................................................................... 141
4-34. MEMBERS OF EACH CLUSTER AND DISTANCES FROM RESPECTIVE CLUSTER CENTRE .................................................................................................................................................................................................................. 141
4-35. K-MEANS CLUSTER ANALYSIS WITH K=3 FOR THE DISTANCE FUNCTIONS USED IN RETRIEVAL OF CASES FROM CB2 WHEN USING EXPERT FEATURE WEIGHTS .................................................................................................................................................................................................................. 142
# List of Tables

2-1. Reaction of isocyanates with hydroxyl containing compounds ............................................. 16
2-2. Customer's needs for a high performance urethane foam. *Adapted from [64 Ch. 5] .... 18
2-3. List of metrics for the high performance urethane given in 2-2. ........................................... 19
2-4. Common properties used as metrics in the final specifications for formulating foamed PU elastomers. *Source: PU Foam Association [57, 66]................................................................. 21
2-5. Reactant and additive factors that influence PU formulation design ...................................... 29
2-6. Effect of reduced symmetry combined with methyl substituents in aromatic isocyanates ................................................................. 31
2-7. Effect of reduced flexibility on properties using aromatic isocyanates ................................. 32
2-8. Effect of isocyanate structure on the mechanical properties of PU elastomers [70]. .... 32
2-9. Effect of structure of polyesters derived from adipic and succinic acids on the mechanical properties of PU elastomers. Reference [70] .............................................................. 34
2-10. Hydroxylated chain extenders and crosslinkers ................................................................... 35
3-1. A dataset with seven instances from a heart-disease domain. *Source: [127] ................... 59
3-2. A case-problem set with seven cases from the PU-formulation domain. ............................. 59
3-3. Selected quantitative distance measures. *Adapted from [122], [132] ............................. 63
4-1. Water-blown integral self-skinning flexible PU foam formulation ........................................... 91
4-2. Combustion modified high resilient (CMHR) foam formulation ......................................... 96
4-3. Foam expert's selected variables levels to yield a stable foam ............................................. 96
4-4. Selected quantitative distance functions applied to CB1 and CB2 for analysis ............... 102
4-5. Retrieved cases using standard distance functions using equal feature weights for the CB1 .......................................................................................................................... 119
4-6. Retrieved cases using equal feature weights from the CB1 for different queries when using low and high order geometric, central and Legendre moment functions ...... 119
4-7. Retrieved cases using standard distance functions using expert feature weights for the case-base 1 .............................................................................................................. 120
4-8. Retrieved cases using expert feature weights from the CB1 for different queries when using low and high order geometric, central and Legendre moment functions ........... 120
4-9. Retrieved cases using standard distance functions with equal feature weights for the CB2 .......................................................................................................................... 121
4-10. Retrieved cases from the CB2 using equal feature weights for different queries when using low and high order geometric, central and Legendre moment functions. 121

4-11. Retrieved cases using standard distance functions using expert feature weights for the CB2. 122

4-12. Retrieved cases from the CB2 using expert feature weights for different queries when using low and high order geometric, central and Legendre moment functions. 122

4-13. Weights rated by a PU expert scaled between [0, 1] for each feature of CB1 and CB2. 136

4-14. Summary of the results of the cluster analysis performed on the standard distance functions for CB1 and CB2. 148

4-15. Summary of the results of the cluster analysis performed on moment-based retrieval functions for CB1 and CB2. 149

5-1. Dataset for neural network feasibility study 176

5-2. Data sets used for network model 178

5-3. Network functions and training parameters used in MATLAB 183

5-4. Network functions and training parameters used in GHOST 184

5-5. Results on the 20th thousand iterations 189

5-6. Results for trained network using merged training and test set 190

5-7. Testing results of best performance network 190

5-8. Results for the network NET 8 191

6-1. Factor description and levels 222

6-2. Experimental design with 18 observations 223

6-3. Measured foam physical properties 224

6-4. (8^2) fractional factorial design for the CMHR foam 225

6-5. Foams' reactivity profile. Average of two values reported 226

6-6. Measured foam physical properties. Average of three values reported 227

6-7. Measured foam physical properties. Average of three values reported 228
Publications originated from this work

The following publications have resulted from carrying out this work

**Journal Papers** (submitted)


(To be submitted) "CBR adaptation Using an Artificial Neural Network that learns from the Case-Base". Segura D.M., West A. D, Hinde C. J. (2006), Expert Systems With Applications

**Book Chapters**


**Conference Papers**


**Conference Posters**


**Technical Reports**


Chapter 1

PROBLEM DEFINITION AND RESEARCH OBJECTIVES

1. CHAPTER 1. PROBLEM DEFINITION AND RESEARCH OBJECTIVES

1.1 Problem Statement

Polyurethanes (PUs) are complex materials to formulate and there has been little structure and scientific method behind their development. They are one of the most versatile polymers as they can be tailor-made into foams, adhesives, sealants, and elastomers for many industrial and everyday applications. These applications have been developed mostly by trial and error methods. Nevertheless, this unsystematic approach to formulation has resulted in many innovative and useful materials. However, further PU development requires products produced faster (reduced lead times), cheaper (lower development costs) and with greater ease of processing and design flexibility in order to penetrate new markets and to comply with environmental legislations (e.g. waste management, sustainability). Without a true understanding of the formulation of PUs it is difficult to strive for innovative PU materials that meet predetermined specifications. This understanding can only be achieved through a systematic approach (i.e. does not rely entirely upon trial and error) to formulation.

New product development in an industrial setting involves study about a variety and sometimes contradictory considerations [23]. These considerations include formulation design, product performance requirements, materials and processing costs, market trends, and governmental regulations [23]. Formulation design requires a solid scientific understanding of PUs. Approaches to formulation design could be divided into two methods, systematic or rational, and unsystematic or heuristic.

Study into the structure-property relationships (SPR) of PUs, aiming for the development of a model or detailed understanding of how changes in PU formulation affect product properties, has been the scientific approach followed in academia as evidenced by academic literature [24-26]. This approach contrasts the heuristic approach followed by some in industry in which previous performance experiences are used to enable PUs to be formulated, synthesised, and tested. Eventually, novel products are obtained, but success depends on skill, intuition, and expert experience. In addition, while academic studies on SPR are openly
published giving some insight into a rational formulation design, industrial PU experts are reticent to reveal their knowledge, even within their workplace.

In the SPR approach, one attempts to visualise, directly or indirectly, the actual PU morphology at different levels of structure (e.g. micromolecular interactions and macromolecular phase separation) in order to relate it to observable parameters, commonly its bulk properties (e.g. stiffness and tensile strength). By this means, several studies that partially address the deficiencies of the traditional empirical route have been done to date [24, 27]. These studies have provided crucial knowledge on how formulation and processing parameters affect PU properties. However, the ability to predict accurately and effectively PU properties has not been achieved. This is due to the highly complex interactions between components further affected by processing conditions in PU systems. Clearly, there is a need to investigate alternative approaches that enable effective PU formulation while at the same time providing a structured framework for PU formulation problem solving.

1.2 Objectives and Research Questions

The main objectives of this thesis can be stated as:
1. to investigate how to assist the PU formulation design using a methodology that enables the systematic and consistent development of new formulations,
2. to provide information of how desired properties can be obtained by changes in formulation (predictive), and how changes in formulation affects product properties (deductive),
3. to provide a PU formulation problem solving framework that allows reuse of accumulated knowledge and expertise for future problem solving while reaching solutions to formulations faster (reducing testing time) and cheaper.

These objectives can be met if the following research questions are answered (refer to Figure 1-1):
1. What computational methodologies are available for dealing with complicated problems arising in PU formulation? What are the basic assumptions for using these techniques? What technique is the most appropriate to support the formulation of PUs? Why?
2. How could these techniques be improved so that they are able to be used to determine optimal solutions?
3. How can experimentally developed formulations be incorporated into a fast and accurate validated computational method that can be used to support users for the effective formulation of PUs?
PU Formulation Problem

How to?
Improve current approaches to formulation?
Reuse accumulated knowledge and expertise?
Understand formulation ingredients-properties relationships?
Preserve knowledge and expertise for future problem solving?
Reach solutions to formulations faster (reduce testing time) and cheaper?

Chapter 1, 2

Research Question 1
What methodologies are available for dealing with complicated problems arising in PU formulation?

Research Question 2
How to improve a chosen methodology or technique to generate optimum solutions?

Research Question 3
How can the problem of formulation of PUs be supported with computational tools?

Research Question 4
How can experimental developed formulations be incorporated into a validated computation methodology that can be used to guide/advise users to effectively formulate PUs?

State of the art to assist PU formulation

Case-Based Reasoning (CBR)

Problem Description
Retrieve Similar
Adapt
Review
Retain

Evaluation of Methodology using two case studies: PU flexible PU foam formulation

Figure 1-1. Definition of the research problem studied in this thesis
1.3 Experimental Setup

1.3.1 Research Question 1

The first research question is addressed via a literature review of the methodologies and techniques available for dealing with complicated problems arising in formulation design. Evaluation of Rule-Based Systems (RBS) and Case-Based Reasoning (CBR) paradigms has indicated that the CBR technique [28] provides the most appropriate method for PU formulation problem solving. The focus of the present work has been on (i) the codification of available experimental formulation data within the CBR framework and (ii) the assessment of a CBR framework to support the retrieval of PU formulation by using (i). It should also be stated that CBR provides an effective framework for integrating many different Artificial Intelligence (AI) techniques and paradigms. This feature is essential for retrieving a past case and adapting it to the needs of a new problem. By reflecting the way experts revise their knowledge when dealing with new circumstances, different types of AI techniques may be combined within the CBR framework to match the characteristics of the problem domain and comply with the principles and assumptions of the PU formulation process.

1.3.2 Research Question 2

The retrieval phase is one of the crucial steps in CBR systems because by definition the most similar cases retrieved need the least adaptation. Then, if matching is poorer, the system will propose distant and less useful solutions. Therefore, if the retrieval algorithm is optimum, the system is capable of producing an optimum match and optimum solutions are more likely to be generated. For this reason, the design and implementation of an effective case retrieval algorithm is an important issue. Several algorithms have been developed to date; however, their use is in some cases domain dependent and study of the research does not reveal what distance function should be used in the retrieval algorithm and why. This research question addresses the identification of an alternative similarity measure i.e. a global similarity measure and its application to the retrieval of PU formulations. A novel similarity measure based on the “moments” description of a case when this is represented as a two-dimensional image has been developed. The use of moments in shape-based object recognition problems [29] indicates that matching of objects can be performed using moments descriptors of objects that represent the global (instead of local) features of the object. In CBR, a moment of a case that has been represented as a two-dimensional image can be interpreted as a descriptor that synthesises the information contained in a case and hence it describes it globally. One of our primary interests is in determining how well a case may be characterised by a small finite set
of its moments. A comparison between the standard measures of similarity and the image's moments based approach is discussed for the retrieval of PU foam formulations. The aim is to evaluate how the alternative case matching approach contrasts the classical similarity measures when only numerical attributes are used to describe a case.

### 1.3.3 Research Question 3

Adaptation is one of the challenges in the development of complete performing CBR systems. In general, adaptation within a CBR framework stands for the modification of a retrieved case that matches partially a query to solve a new PU formulation problem [28]. In particular, for the PU formulation domain, adaptation is expressed in terms of how to modify a retrieved formulation i.e. what ingredients and their quantities meet the given constraints. The optimisation of this formulation in economic terms, i.e. what is the cheapest formulation that would give the desired properties is an important issue but it is out of the scope of this thesis due to the limitations of cost data available.

The use of the artificial neural network (ANN) approximation technique is evaluated for the adaptation stage in the CBR cycle. In this thesis ANNs are proposed as a technique that could be able to reduce the adaptation knowledge acquisition effort that has hindered the building of complete CBR systems. Artificial Neural Networks have proven useful in domains where theoretical descriptions are difficult to obtain, but partial knowledge about the process is known and input–output patterns are available. Quantitative relationships between the composition and the properties of the product can be modelled by the ANN approach. This research question addresses the development, use, and implications, of an ANN acting as an interpolating function to estimate changes in a formulation when given specific formulations with specific mechanical properties.

### 1.4 A Road Map to this Thesis

This thesis consists of three different parts.

Part I consists of two chapters (Chapter 2 and Chapter 3) and answers research question 1. Chapter 2 commences with a definition of PU formulation in the framework of flexible PU materials. This is followed by a description of the basic chemistry involved in PU formulation and a description of the strategies currently used to formulate PUs. The methodologies that can be used in dealing with complex problems arising in formulation design are reviewed in Chapter 3. Part I closes with a summary and conclusions drawn.

Part II is composed of two chapters. Chapter 4 reports on an alternative similarity measure based on the moments description of a case when it is represented as a two dimensional
image. In this chapter, retrieval using geometric, central and Legendre moments is compared with a standard nearest neighbour algorithm using nine different distance functions (e.g. Euclidean, Canberra, City Block, Divergence, Chebychev, D7, D8, D9, D10). The problems of CBR adaptation are studied in Chapter 5. For the adaptation of retrieved cases, an ANN approach was proposed to enable build complete CBR systems by using a knowledge light approach i.e. by using the case-base to reduce the adaptation knowledge acquisition effort. This chapter studies how an ANN could acquire knowledge from the case-base and use it to guide CBR adaptation. A description of the basic assumptions for using this technique and the limitations of using this approach are also reviewed. The methodological issues to build learning systems and how an ANN can be implemented to adapt retrieved PU cases when these are represented as feature vectors are explained.

Part III presents the conclusions of this thesis and outlines future work that is required if this system is to be implemented in a real PU manufacturing process.
PART I. LITERATURE REVIEW

Polyurethanes (PUs) are a well established class of polymeric materials that have found their way in many diverse industries and applications (e.g. automotive [30, 31], construction [32], electronics [33], environmental and medical [34, 35]). PUs were discovered in 1937 [36] and since then, by varying the nature of the constituent materials (e.g. monomer precursors such as polyols, isocyanates, and chain extenders), many applications have been developed, ranging from thermal insulation foams to shoe soles, and from abrasion resistant coatings to biomedical implants, and Spandex fibres (see [37] for a review of PU applications).

Nowadays, it is difficult to think of a world without PUs. In 2000, the global consumption of PU was 8 billion kilograms [38] and it is estimated that new markets are increasing the demand for PU materials, mainly in the construction and automotive industries. Asia, and South and Central America are having the strongest expansion, growing on average 3% per year [39]. Germany, the USA, and Italy are the leading exporters, accounting for over 54% of PU exports in 1999 [39].

The large volume of PU usage is understandable, because of the wide range of properties PUs can offer and the many economical, environmental and sustainable benefits these materials provide. For instance, in terms of efficiency, PUs can be tailor-made to meet exact shape and size requirements thereby reducing the generation of waste and reducing time and costs in manual finishing operations. Furthermore, diverse PU products can be recycled or energy recovered by a range of approved technologies [40-43]. PUs are also considered sustainable materials [44, 45] as they can be also used to reduce energy consumption, for example PU foams' low density characteristics provides a means of weight reduction in cars (i.e. car seat cushioning), thus giving rise to improved fuel efficiency [45]. Rigid PU foams' excellent insulation properties allow for the exclusive use of PU in buildings and diverse appliances to reduce heating costs [46]. Other benefits PUs offer rely on their versatility, which makes them ideal suited to a broad range of needs. For these reasons, the development of PUs is very important for their applications as commodities and as low volume niche specialities.

Even though many PU applications exist, their formulation chemistry can be very similar. This similarity across applications allows formulators to apply their expertise to various applications. This similarity arises because in the formulation of PUs, only a few chemical families are used (hydroxyl-, isocyanate- or amine-containing compounds) and the molecular structure-property relationships found in one application can usually be transferred to another. What distinguishes one application from another is the kinetics that takes place depending on
the chemical nature of the compound used (e.g. linear polyols used in thermoplastic PU elastomers, highly branched polyols used in rigid foams), the additives used, and the conditions for the reaction. For example, structural adhesives are formulated similarly to hard elastomers or high density rigid foams [47]. As a consequence of the chemical similarity across formulations for different PU applications, formulators can theoretically design new formulations for almost any PU application if they have a comprehensive knowledge of the nature of the system they are formulating. Conversely, the skills of the operating personnel vary between manufacturing technologies, which seems reasonable as each PU's manufacturing process has fundamental differences. For this reason, only formulating experts or those who have a comprehensive and authoritative knowledge or skill in the area can deliver products in time within budget and at a quality that meets customer needs.

However, only a few experts in the field have attained this capability. This has led to an increasing monopolisation of knowledge by the raw material manufacturers who “own” the formulation expertise. For example, a recent visit by the author to Superlon Oy¹, one of the largest foam manufactures in the Scandinavian region, revealed that foam manufacturers had little intimate control over their PU systems. If a PU foam is out-of-specification, they could only modify the processing parameters up to a certain extent. However, this is not always the case, as PUs are reactive systems which are very sensitive to processing variations, for instance, temperature, mixing speed or contamination along the manufacturing line, or any change in the processing is likely to be reflected in the properties of the final product. Companies, such as Superlon Oy, do not know how to adjust their PU formulations to solve the problems and challenges they are facing now. Instead, they have been relying entirely upon their suppliers for PU formulation adjustment. Mostly, this is because they do not know what exactly is in their PU systems, so we may assume they work “blind-folded”. Currently if Superlon Oy would like to produce a different product, they would have to buy a new system at higher cost from a raw material supplier or a formulation house company. In doing so, they would not have ownership of the knowledge behind the development of the formulation they have bought, other than the general process and service properties that roughly meet their requests.

This monopolisation of knowledge by the raw material manufacturers is of concern for the future PU development and innovation. PU manufacturers will have to face the demand placed by new market and governmental regulations that this century brings. The major challenges for the PU industry relate to the growing regulatory focus on issues such as sustainability and climate change [48-54]. As it is shown in Figure 1-2 some of these

---
¹ Superlon Oy's website: http://www.superlon.fi/
Problem Definition and Research Objectives

challenges have already begun to be tackled at different levels by research efforts in both academia and industry. For example, the increasing health and environmental concerns in recent decades have increasingly forced the solvent-based PUs to be replaced by aqueous PU dispersions (PUDs) [55]. Another example relates to the phase-out of chlorofluorocarbons (CFCs) in the PU foams due to restriction imposed by the Montreal Protocol in 1987 [56]. Most research and development in this area has been focused to find non-ozone depleting substances with equivalent or better performance than CFC 11 in PU foams; this is, to improve insulation efficiency or to reduce density levels. Another example is in the appliance industry, where the focus on global warming has led to intensive research to reduce the energy consumption of appliances [46]. This is currently being achieved by a combination of improvements in the various components such as the compressor efficiency, layout of the cooling circuit, appliance design, but mostly to efforts in the PU foam insulation formulation optimisation [46].

However, in order to comply successfully with these regulations while taking advantage of the challenges they offer at the same time (e.g. new market opportunities see Figure 1-2 part b), PU manufacturers desperately need to understand their products’ development so that they can face effectively any innovative challenges.

Knowledge is the most competitive resource in this century for any industry [57]. For the PU manufacturers this is the greatest challenge of all, as they need to understand the technology intimately. They need to possess the knowledge, readily assess, and deploy it when required. This will bring a strategic advantage to their business as knowledge will allow them to experiment and implement their ideas before others do and profiting by the challenges imposed on them in an independent fashion, avoiding reliance on their material suppliers. Therefore, in order to enable PU organisations the ability to capture, share, and apply collective experience and knowledge fundamental for progress, the ideas and technology that have been developed by the Artificial Intelligence (AI) community are required. What has been learned from the AI community about knowledge management e.g. knowledge acquisition, representation, inference, search, and learning could be used for assisting formulators as they solve problems. The exploration of these topics to assist the PU industry and in particular the PU formulation task is one of the aims in this thesis.

A road map for Part I

---

2 The Montreal Protocol is an international agreement designed to protect the stratospheric ozone layer. The treaty was originally signed in 1987 and substantially amended in 1990 and 1992.
The area covered in discussing these issues is necessarily broad, and involves a synthesis of ideas and concepts from various disciplines e.g. PU chemistry, PU formulation, and AI. The features that make a PU formulation problem a natural AI application and the complexity of the PU formulation task are fully described in Chapter 2. This description is based on a series of interviews held with a PU expert, Dr Richard Heath. Analogies with other PUs or formulated products can be made, so the description may apply to similar disciplines, (e.g. agrochemicals, pharmaceuticals, rubbers, alloys, among others). Specifically, a description of formulated PUs is given first followed by a review of the current and long established approaches to PU formulation. Emphasis has been given to the knowledge a formulator needs to draw upon when formulating (i.e. knowledge about chemical variables that affect the formulation of PUs, knowledge at a micro and macro-structure level, knowledge about the processing variables that affect the formulation PUs and their relationships, and understanding of end-use environment application).

Finally, the problem of formulating PUs is described along with the motivations for a computational approach that could solve the problems arising in the formulation of PUs. The methodologies that can be used in dealing with complex problems arising in formulation design are reviewed in Chapter 3.

---

3 Dr Richard Heath has worked in PUs for nearly 40 years both as a PU technologist at both Marley Polyurethanes and Armstrong Cork's Middlesbrough plant and as a PU scientist in the IPTME department at Loughborough University.
Figure 1.2. Main aim in this thesis.
The main aim in this thesis is to study a computationally methodology that can assist the formulation of PUs while at the same time enable the representation, capture and utilisation of PU formulation current production data.
Chapter 2

POLYURETHANE FORMULATION

*How to solve complex problems arising in the formulation of PUs?*

Abstract: The formulation of polyurethanes is a difficult task. This is due to complex interactions between components further affected by processing conditions. This process often involves reasoning about microstructure, abstract concepts, interpretations and heuristic "rules of thumb" based on experience. The reasoning is highly symbolic rather than numeric. The aim of this chapter is to understand the need for and motivations behind an investigation into computational tools to assist the formulation of PUs. The PU formulation process and the PU chemistry were reviewed by showing examples of common PU formulations. Although the literature available offered a source of diverse knowledge on the field, this knowledge is not comprehensive enough to provide a source of expertise to support non-expert formulators when developing new formulations.

Key words: PU Formulation, Formulation Process, Process Parameters

2. CHAPTER 2. POLYURETHANE FORMULATION

2.1 Objective of this Chapter

The objective of this chapter is to understand the need for and motivations behind an investigation into alternative computational tools that enable formulation, problem solving, and the development of a framework to represent, structure, and deploy current PU knowledge. This is in order to enhance processing capabilities, so that the PU formulation task can be eased. This objective can be satisfied if the following topics are reviewed:

- **PU chemistry.** This covers the description of what is understood for a PU formulation (section 2.2);

- **PU formulation process.** Firstly, a description of how customers normally specify a PU foam is presented. Secondly, the formulator's reasoning when facing a new customer demand and producing a specified PU is described (section 2.3);

- Traditional approaches to PU formulation and the knowledge a formulator draws on when formulating are presented (section 2.4).
2.2 What is a Polyurethane Formulation?

PUs are produced according to a chemical formula in which a number of ingredients are combined under specific processing conditions. The formula is a list of ingredients in various proportions that are mixed to provide a product with properties tailored to meet functional requirements. The specification normally varies from one application to another. In some cases, it may be extremely precise, expressed in terms of a performance level, or conditional on a defined test. In other cases, it may be very general. Furthermore, it may contain many, often conflicting; constraints that formulators might need to redefine from experience in order to balance customers' needs. Typically, when formulating PUs the mixed ingredients interact chemically and consequently, product properties are governed by exact ratios of ingredients. Additionally, the manufacturing process also affects the product's final properties. Therefore, the processing method and the processing parameters used are often considered an integral part of a product's formulation.

A PU formulation includes, besides the polyols and isocyanates, catalysts and other additives specific to the desired application. Usually for the process stage, PU formulations consist of two reactive components, first an isocyanate and second a masterbatch, which is a pre-mixed system with the remaining reactant or additives. For PU foams, the masterbatch may include catalysts, blowing agents, fire retardants, pigments, and surfactants premixed with a polyol or polyols according to a predetermined formula. In the USA, this is known as the B-Side, "Cross-linker"- or "Catalyst-Side", and in Europe it is referred to as the "A-Side". The isocyanate part is arbitrarily called "A-Side" in the USA or "B-side" in Europe.

A model formulation for a flexible moulded PU foam in which auxiliary blowing agents and surfactants are used is shown in Figure 2-1. These components are only present in most foamed formulations. Several additives distinguish one formulation from another resulting in different chemical and mechanical properties.
A typical PU foam formulation consists apart from polyols and isocyanates, of additives such as catalysts, surfactants, fire retardants, pigments and blowing agents, (e.g. water, hydrocarbons or hydrochlorofluorocarbons). Typically, in the formulating of a PU in the laboratory, the polyols are first premixed with the additives to form the B-Side or Polyol-side and they are subsequently mixed at specific ratios with isocyanates to produce the foam.

Some of the PU formula terms that appear in Figure 2-1 are defined as follows [58]:

1. **Index.** It is a measure of the stoichiometric balance between the equivalents used to the total equivalents of water, polyols and other reactants, e.g. the relative amount of isocyanate used as compared with the theoretical requirement. An index of 100 means just enough isocyanate is provided to react with all compounds containing active hydrogen atoms (e.g. polyols, water, primary and secondary amines). However, the PU industry has tended to be imprecise in its usage of indexes. For instance, when isocyanate index >100 is employed, this may be to compensate for batch to batch variation of reactant but also to encourage secondary reactions to occur (e.g. biuret and allophanate reactions (Table 2-1)). This limits the usefulness of the index since it is not always used as a scientifically predetermined and hence useful calculation of reactant quantities. It is argued by the author that if formulation is to be controlled then secondary reactants must also be part of the rigorous formulation and not left to the experience of formulators.

2. **Equivalent Weight.** The equivalent weight (Eq Wt) of a polyol is the mass of active hydrogen components per reactive hydroxyl group. It is calculated by dividing the
molecular weight of each hydroxyl-containing compound by the compound’s functionality (i.e. number of reactive sites).

\[
\text{Equivalent weight} = \frac{\text{Molecular Weight}}{\text{Number of reactive sites}}
\]  

(2-1)

3. **Iso/Polyol Ratio (NCO/OH).** It is the relationship between the isocyanate quantities to polyl.

4. **Iso/Amine Ratio (NCO/NH).** It is the relationship between the isocyanate quantities to amine.

5. **OH Number.** This number indicates the number of reactive hydroxyl groups available for reaction. It is expressed as the number of milligrams of potassium hydroxide equivalent to the hydroxyl content of one gram of the sample.

6. **Molecular Weight per Crosslink (Mc).** It is a widely used term that expresses the molecular weight of the repeating branched unit in the crosslink polymer derived from a tri-functional monomer. It is equal to the unit weight of polymer divided by the number of crosslink junctions or branch points in the unit weight of polymer.

7. **Average Equivalent Weight of Polyol.** Mass of polyol per reactive hydroxyl. One equivalent weight of polyol will completely react with one equivalent weight of isocyanate.

The main reactions that characterise PU chemistry are described in Table 2-1. They are the reactions of isocyanates with compounds containing hydrogen atoms attached to a nitrogen atom, and of great importance polyols (e.g. high molecular multifunctional esters or ethers), chain extenders (low molecular hexols e.g. glycols), water, and amines. Furthermore, the reactions of isocyanates with urethanes, ureas, and amides are also of significant importance in PU chemistry as they will lead to increased branching and product variation [59, 60]. Exhaustive studies of the chemistry of organic isocyanates, their reactions, properties and applications can be found elsewhere [59, 61], more recent reviews are found in [62].
Table 2-1. Reaction of isocyanates with hydroxyl containing compounds

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{R}_1\text{N} = \text{C}=\text{O} + \text{R}_2\text{OH} \rightarrow \text{R}_1\text{N} - \text{C}-\text{O}-\text{R}_2$</td>
<td>The reaction proceeds at ambient temperatures without the use of catalysts. Reactivity is higher for primary alcohols, decreasing for secondary, tertiary and aromatic alcohols. Isocyanates react with amines usually at 0-25°C yielding urea. Primary aliphatic amines react most quickly followed by secondary aliphatic amines and aromatic amines.</td>
</tr>
<tr>
<td>$\text{R}_1\text{N} = \text{C}=\text{O} + \text{R}_2\text{NH}_2 \rightarrow \text{R}_1\text{N} - \text{C}-\text{N}-\text{R}_2$</td>
<td>Isocyanate reacts with water giving carbamic acid, an unstable compound that spontaneously decomposes into a primary amine and carbon dioxide, together with a subsequent urea formation. This reaction is important in the formulation of foamed PUs and PU adhesives and sealants that cure with humidity. The control of water in the PU formulation is very important. If it is not controlled, a considerable amount of heat is generated, which in the case of PU can be detrimental [63]. Isocyanates can react, in so-called excess conditions i.e. high index, with urea and a urethane group to produce biuret, and allophanate compounds respectively. The first reaction is faster than the latter [59 p.3]. These reactions will occur more readily at higher temperatures or specific catalyst conditions. It can be seen in these reactions that the products, allophanates and biurets are in equilibrium (i.e. reversible) with the starting materials, isocyanates and active hydrogen compounds.</td>
</tr>
<tr>
<td>$\text{R}_3\text{N} = \text{C}=\text{O} + \text{R}_1\text{N} - \text{C}-\text{N}-\text{R}_2 \rightarrow \text{R}_3\text{N} - \text{C}=\text{O}$</td>
<td>Isocyanate reacts with water giving carbamic acid, an unstable compound that spontaneously decomposes into a primary amine and carbon dioxide, together with a subsequent urea formation. This reaction is important in the formulation of foamed PUs and PU adhesives and sealants that cure with humidity. The control of water in the PU formulation is very important. If it is not controlled, a considerable amount of heat is generated, which in the case of PU can be detrimental [63]. Isocyanates can react, in so-called excess conditions i.e. high index, with urea and a urethane group to produce biuret, and allophanate compounds respectively. The first reaction is faster than the latter [59 p.3]. These reactions will occur more readily at higher temperatures or specific catalyst conditions. It can be seen in these reactions that the products, allophanates and biurets are in equilibrium (i.e. reversible) with the starting materials, isocyanates and active hydrogen compounds.</td>
</tr>
<tr>
<td>$\text{R}_3\text{N} = \text{C}=\text{O} + \text{R}_1\text{N} - \text{C}-\text{O}-\text{R}_2 \rightarrow \text{R}_3\text{N} - \text{C}=\text{O}$</td>
<td>Isocyanate reacts with water giving carbamic acid, an unstable compound that spontaneously decomposes into a primary amine and carbon dioxide, together with a subsequent urea formation. This reaction is important in the formulation of foamed PUs and PU adhesives and sealants that cure with humidity. The control of water in the PU formulation is very important. If it is not controlled, a considerable amount of heat is generated, which in the case of PU can be detrimental [63]. Isocyanates can react, in so-called excess conditions i.e. high index, with urea and a urethane group to produce biuret, and allophanate compounds respectively. The first reaction is faster than the latter [59 p.3]. These reactions will occur more readily at higher temperatures or specific catalyst conditions. It can be seen in these reactions that the products, allophanates and biurets are in equilibrium (i.e. reversible) with the starting materials, isocyanates and active hydrogen compounds.</td>
</tr>
</tbody>
</table>

Reaction 1. Reaction of isocyanates with alcohols

Reaction 2. Reaction of isocyanates with primary amines

Reaction 3. Reaction of isocyanates with secondary amines

Reaction 4. Reaction of isocyanates with water

Reaction 5. Biuret formation

Reaction 6. Allophanate formation
As can be seen from Table 2-1, $R$ stands for an aromatic or aryl group in either organic component. By choosing the nature of only this group in either the isocyanate, polyol or chain extender, formulators can introduce changes in the structure of the resultant polymer and hence give rise to tailor-made PUs with a variety of properties [47]. However, in order to obtain successful end products, a chemical background and many years of experience are required.

2.3 Formulating PUs

When designing a PU formulation, the formulator normally begins to establish the problem with a set of constraints on the mechanical properties of the polymer product to manufacture. Some of the common PU material specifications in the PU domain are explained below in section 2.3.1. Once property constraints have been specified and the problem is fully described, formulators usually select some baseline formulation to begin their search for a solution. PU formulators use several different strategies for selecting a baseline, and from there they attempt to find a suitable formulation that can meet the specifications given by the customer. The most common strategies to PU formulation are the statistical design of experiments or by designing from first principles. These strategies and the general PU formulation process are explained in section 2.4.

2.3.1 PU Materials' Specifications

Depending on the intended application of the PU product, several features are used by (i) customers to specify the material they require or (ii) manufacturers to market the products they offer. There are similarities and differences between these descriptions. However, end users probably do not fully know the kind of product they want and usually describe it imprecisely. Furthermore, customer needs are generally expressed in the "language of the customer", typically subjectively hence providing little specific guidance about how to design a formulation to produce a particular PU. Essentially, specifications are the translation of customer needs into technical terms. They provide a precise description of what the PU has to do. In general, this description includes technical requirements for materials together with the procedures to be used to determine whether the requirement has been met.

A typical specification consists of a metric and a value. The value can take on several forms, including a particular number, a range or an inequality. For example, "average time to cure" is a metric, while "less than 5 minutes" is the value of this metric. Values are always labelled with the appropriate units. The units of measurement are most commonly conventional engineering units such as kilograms and seconds. However, some metrics will
Polyurethane Formulation

not lend themselves to numerical values. In some cases, the metric involves the success or failure of a test, which can be indicated by “binary” in the units column (see Table 2-3). An example is when a PU fails a UV degradation test, the report is usually reported as either “fail or no fail”. A set of individual specifications makes the overall available material specifications. Other terms used by the PU industry include “product requirements”, “engineering characteristics”, “specifications” or “technical specifications”.

PU raw materials’ manufacturers regularly publish comprehensive material specifications and performance data in product catalogues and other technical literature (see [64] for example Bayer’s product specifications). Nevertheless, some specifications can appear shallow even if they provide important insights into the potential use or application of a product. Although many of these figures, including chemical reactivity and end-properties, are self-explanatory, many are far from straight forward. Unsurprisingly people without a specific background in polymer chemistry do not always fully comprehend how some product’s specifications are calculated, or how to interpret important statistics in real-world applications.

Like in any product process development, immediately after identifying the customer’s needs, target specifications are set. These specifications are constrained by the product technology and will be refined after a product concept has been selected. A method for the specification process that can be adapted to the formulating design process is given by Ulrich and Eppinger [65 Ch. 5].

The primary customer needs for a high performance urethane foam are listed in Table 2-2.

Table 2-2. Customer’s needs for a high performance urethane foam. Adapted from [65 Ch. 5]
This is an illustrative example in which the values have been assumed.

<table>
<thead>
<tr>
<th>No.</th>
<th>Need</th>
<th>Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The foam</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>has excellent hydrolytic stability</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>preserves colour with time</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>can be easily cleaned</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>is lightweight</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>lasts a long time</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>can be used in various environments</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>can be easily removed from mould</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>instils luxury</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>reduces vibration</td>
<td>3</td>
</tr>
</tbody>
</table>

As it can be seen in Table 2-2 the description of the desired material is given in terms of a subjective quality of expressions. When translating these needs into specifications, a precise and measurable characteristic of the material for each need will reflect the degree to which the material satisfies the need (see Table 2-3). In the ideal case, there is only one metric for each need. However, this is frequently not possible in practice.
In addition, the importance weightings of a metric are derived from the importance rating of the needs it reflects. This of course depends upon the product’s application. For cases in which a metric maps directly to a single need, the importance rating of the need becomes the importance rating of the metric. Conversely, for cases in which a metric is related to more than one need, the importance of the metric is determined by considering the importance of the needs to which it relates, and the nature of this relationship [65].

Some specifications to consider for a formulation team in the development of PU materials include mechanical properties, chemical and physical properties, aesthetics, rheological properties, and chemical and environmental resistance.

Table 2-3. List of metrics for the high performance urethanes given in Table 2-2.
A metric is a measurable characteristic of the material that reflects the degree to which the material satisfies a need.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Need Nos.</th>
<th>Metric</th>
<th>Importance</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 5</td>
<td>Change in tensile strength after 50 hours after attack by steam hydrolysis</td>
<td>3</td>
<td>MPa</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Gardner colour change after oven aging at 130°C</td>
<td>1</td>
<td>Hours</td>
</tr>
<tr>
<td>3</td>
<td>3, 5</td>
<td>UV test duration to degrade foam</td>
<td>5</td>
<td>Hours</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>Total weight</td>
<td>4</td>
<td>kg</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>Tear strength</td>
<td>3</td>
<td>kg/cm</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>Instills luxury</td>
<td>2</td>
<td>Subjective</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>Spring rate in load per unit deflection</td>
<td>3</td>
<td>Nm/rad</td>
</tr>
</tbody>
</table>

Table 2-3 shows that there is more than one metric for some needs, for instance, need No. 5 (lasts a long time) two metrics are measured (i.e. metric No. 1 and 3) to reflect the degree to which the material satisfies the need. This is because designers may conclude that several metrics are required to capture specific needs. Metrics specify the overall performance of a product and should therefore be a dependent variable in the design problem [65 Ch. 5]. However, some needs cannot be translated into quantifiable metrics. In these cases the metric is subjective and is normally evaluated by a panel of customers.

In the realm of solid PU materials, product specifications usually relate to two types of technology i.e. cellular (or foamed PUs) and non-foamed PUs. The basic difference is the use of water in the formulation to allow for the production of carbon dioxide, for instance, during the foaming reaction. Generally, cellular PUs exhibit improved thermal and acoustical insulation, and load bearing capability. Various techniques are available to specify these materials commonly by using several different protocols, including methods contained in ASTM, ISO, and BIFMA testing standards. Similarly, non-foamed PU materials including cast, reaction injection moulding (RIM), and thermoplastic PUs are specified in terms of
mechanical performance (e.g. stress-strain properties) depending on their intended final application [66].

This thesis is focused on flexible PU foams (FPFs). The metrics commonly used for FPFs are presented in Table 2-4. As explained before, some of these metrics measure one or more different characteristics of a PU. They reflect the degree to which the material satisfies the customer’s needs. For example, the load bearing capacity of a foam is revealed to some extent by measuring the density of the foam, which in turn is based on experience of the influence of density variation within a grade of PU foam. However, a more comprehensive test such as the indentation force deflection (IFD) gives more insight into the load bearing properties of a foam [67]. It is usual to find in the specification for FPFs these two metrics together i.e. density and 25% IFD (firmness). Specification of either property alone is usually not sufficient. Firmness is almost never specified without density [67]. Other metrics such as compression modulus or support factor are also used when specifying FPF. Commonly, higher densities can provide increased support which in turn varies within ranges with different densities, types and grades of foam.
<table>
<thead>
<tr>
<th>Property</th>
<th>Definition</th>
<th>A measure of</th>
<th>Tests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density</td>
<td>A measurement of the mass per unit volume. It is measured and expressed in pounds per cubic foot (pcf) or kilograms per cubic metre (kg/m³).</td>
<td>Cost and load bearing. High densities generally result in higher costs and improved load-bearing properties.</td>
<td>ISO 845, ASTM D3574 ISO 1183, ASTM D792 (Compact materials)</td>
</tr>
<tr>
<td>Indentation force</td>
<td>IFD is generally measured as the force (in pounds) required to compress a 50 square inch circular indenter foot into a four inch thick sample, typically 15 inches square or larger, to a stated percentage of the sample's initial height. Common IFD values are generated at 25 and 65 % of initial height. Firmness measured by 25% IFD, is an indicator of the surface feel of the foam. Two other terms related to IFD are the Support Factor, and the Guide Factor.</td>
<td>A measure of the load bearing capacity of a foam. Firmness is independent of density. High density foam can be produced to be very soft or very firm, or any firmness in between.</td>
<td>ASTM D3574</td>
</tr>
<tr>
<td>Indentation force</td>
<td>(IFD) or firmness</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Support factor</td>
<td>It is determined by taking the ratio of the foams' IFD at 35% indentation and 65% indentation (65% IFD / 25% IFD). Others terms used to describe foams' ability to provide support include compression modulus, modulus, compression factor, modulus, and sag factor. All these may be used to refer the same properties. Compression modulus measurements for foam range from about 1.8 to 3.0.</td>
<td>Support factor is an indicator of the foams' ultimate ability to support a load placed upon it. It gives an indication of cushioning quality. Seating foams with low support factor are more likely to bottom out under load. Higher support factor indicate better cushioning quality</td>
<td>Calculated from the IFD test</td>
</tr>
<tr>
<td>(compression modulus)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The Guide Factor</td>
<td>Guide factor is useful in determining the relative firmness of foams with different densities. It is calculated as the ratio of 25% or 65% IFD to density.</td>
<td>It is used to compare the economy of foams. Economical foams i.e. the higher the guide factor, can be obtained with a lower density but not necessarily with performance advantages.</td>
<td>Calculated from the IFD test</td>
</tr>
<tr>
<td>Flex fatigue</td>
<td>Fatigue is measured by repeatedly compressing a foam sample and measuring the change in IFD.</td>
<td>It is a measure of softening or loss of firmness.</td>
<td>Measured by different tests:</td>
</tr>
<tr>
<td>(dynamic fatigue)</td>
<td></td>
<td></td>
<td>• Constant-Load Pounding ISO 3385-1975 (E)  • Roller Shear ASTM 3574-86</td>
</tr>
<tr>
<td>Resilience</td>
<td>It is measured by dropping a steel ball onto the foam cushion and measuring how high the ball rebounds. Measurement of a foams' ability to maintain original support characteristics after flexing. Normally, the firmness (IFD) is tested at 25% indentation, 65% indentation and again at 25% on the way up. Hysteresis is the percent of 25% IFD loss measure as a compression tester returns to the normal (25% IFD) position after measuring 65% compression.</td>
<td>It is an indicator of the surface elasticity or &quot;springiness&quot; of a foam. Lower hysteresis values, or less IFD loss are desirable. Hysteresis values may provide a good indication of overall flexible foam durability. Low hysteresis in conventional foam is equal to less IFD loss in durability and fatigue tests.</td>
<td>Calculated from the IFD test</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
<td>Standards</td>
<td></td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
</tbody>
</table>
| Air flow                       | Amount of air expressed in cubic feet per minute, that can be drawn through a 2" x 2" x 1" foam sample at 0.5-inch water pressure differential. Airflow can be a critical factor in FPF performance depending on the application. For instance, in normal furniture cushioning applications, a pneumatic foam with low airflow will have poor durability and high compression sets, regardless of density. However, in a packaging foam, a foam with minimal airflow may be desirable as a shock absorber, or in medical applications, slow recovery foams are used to improve patient care. Low airflow foams are also used in gasketing applications in air conditioning and heating. | • ASTM D3574 (Flexible foams)  
• ASTM D1667 (closed cell foams)  
• ISO 815 (Ambient & High Temp.)  
• BS 903-A6:1992 replaces ISO 1635 (High & Low Temp.)  
• ASTM D395 (Ambient & High Temp.)  
• D1229 (Low Temp.) |
| Compression Set (Constant Deflection Compression Set) | A permanent partial loss of initial height of a flexible PU foam sample after compression due to a bending or collapse of the cell framework within the foam sample. Compression set is measured by compressing a foam sample 90% of its thickness (or down to 10% of its original thickness) and holding it at 70°C (or 158°F) for 22 hours. Compression set is most commonly expressed as a percentage of original compression. Other deflections, times, and temperatures can be used. | It is a measurement of the ratio of elastic to viscous components of a material’s response to a given deformation. A high value of compression set will cause a foam cushion to quickly lose its original appearance with use, leaving its surface depressed or "hollowed out". |
| Flammability and smoke propagation | Numerous tests exist for the evaluation of flammability performance. These tests are carried out on a variety of scales, ranging from laboratory scale to full-scale tests. However, the former cannot be used to predict the performance of these materials under actual fire conditions, rather indicate fire initiation or start of the propagation effects. | |
2.3.2 The Formulation Process

Once a PU has been specified, the process of formulation continues by choosing one or more formulations that can meet the design requirements. After the setting of an initial formulation, testing and reformulation follows. Irrespective of the formulating approach employed (e.g. following a planned statistical design, or formulating from basic principles), at some point in the process, adjustments to a formulation are carried out by trial and error (see Figure 2-2 a). This iterative process can be both time consuming and expensive, and provides a need for the PU industry to consider computational automated approaches in the understanding, development and reformulation of PUs. Figure 2-2 b) shows the tasks and subtasks involved in a PU formulation process.

Problem description

In particular, the problem of formulating a PU is usually described by a set of constraints, which in turn are generally expressed in terms of mechanical properties of the material e.g. “tensile strength must be higher than x, and hardness must be lower than y”. Occasionally, PU foam constraints might involve properties that are not expressed numerically such as
surface appearance or 'touch'. Furthermore, the problem may be over-constrained. This occurs when no formulation exists which gives a material that can meet the specifications given. This can happen when it is difficult to know in advance which problems can be solved.

**Selection of known/similar properties to design target**

Once property constraints are specified and the problem is fully described, formulators draw upon different types of knowledge, which vary amongst individuals, to select some baseline formulation to begin their search for a solution. PU formulators use several different strategies for doing this, but in general they start with a similar formulation to the design target.

Once a base formulation(s) is selected, statistical designs are used in order to recognise the variables that greatly influence a PU system. Statistical design requires an expert to decide the levels at which the variables will lead to stable products. In situations where a product is to be introduced, screening designs are best suited (see for example [69, 70]). However, the subsequent optimisation and tuning of the statistical design is knowledge-intensive. An advantage of experimental designs is that they allow the formulator to understand, by constructing predictive models, the effect of the variables on the PU properties. Even though the design might not result in any commercially viable PU formulations, statistical analysis of unsuccessful products can also shed some light on the understanding of the formulation at hand.

Formulators are also inclined to begin with commercially pure materials and design from basic principles but in real practice this process can take many years to result in a commercial application. However, this approach (and even the statistical design approach) is only achieved by experienced and knowledgeable PU manufacturers.

Depending on the product's application some formulators, when searching for alternatives to meet target properties and constraints, construct a comprehensive model of the microstructure that will meet all the properties and then identify compositions and processing options. However, some formulators prefer to avoid reasoning about microstructure as much as possible by using direct relationships between decision variables and design targets. Commonly, all formulators probe their partial schemes by evaluating the effects on processing decisions, structure and properties. Some features that cannot be controlled are mainly related to environmental conditions such as relative humidity, temperature, and features intrinsic to the equipment operation that restrict process variables variation (e.g. maintenance of pipes and reading of valves in storage tanks).

**Alteration of properties by changes in composition and processing**
The formulation process involves an iterative approach to change the characteristics of the materials hopefully in the direction of the target properties. Formulators look for changes that can be made to the materials or processing conditions in order to improve the properties. However, this complex task is only efficiently done by experts in the area [47]. Features that can be changed include the composition i.e. the components added, their type, their amounts and ratios, and the process, i.e. the fabrication steps used, their sequence, temperature and time.

There are rules that indicate the effects of these design choices on the product's final properties. Some examples are:

- **R1.** IF ester group separation (chain length) in linear adipate polyesters increases THEN tear strength decreases AND elongation (% and set) decreases AND density decreases;
- **R2.** IF side chains increases (linear to branched with methyl side chains) THEN tensile modulus decreases AND tear strength decreases AND density decreases.

Some rules can be more complicated than above as the interaction between components and end-properties can be more complex. The factors influencing this complexity are outlined in the sections 2.4.2 and 2.4.1 (p. 28), where the various interactions, at different levels of understanding, of the several chemical, and processing variables that interact and characterise the PU formulation process are illustrated.

The interaction of the types of knowledge that characterises the formulation process is shown in Figure 2-3. The components of the scheme represent the four classes of knowledge i.e. the composition (C), structure (S), manufacturing or processing (M), and final properties (P). Their relationships are used heavily in the design of new formulations. The rules just mentioned are examples of these relationships. For instance, the first rule R1 given formerly corresponds to relationships composition-property.

---

4 These rules are inferred from the findings of Pigott et al. 71. Pigott, K.A., et al., Development of Cast Urethane Elastomers for Ultimate Properties. Journal of Chemical Engineering Data, 1960, 5: p. 391- when studying the influence of polyesters' chain length based on adipic and succinic acid on the mechanical properties of PU elastomers prepared from MDI extended with 1,4-butanediol.
These types of rules are deduced through experimentation and testing i.e. by gaining experience. The rules are applied opportunistically to make some progress in solving the problems arising in PU formulation. However, rules that involve reasoning about structure and microstructure are given a preference over rules that do not. This is because the relationship is “more understood” i.e. it can be explained systematically.

In addition, when developing materials with properties outside of the range of existing materials, formulators must reason about designs which have never been tested and whose behaviour is not known with certainty. This gap is “filled” with general models, speculation, extrapolation of known trends, and analogy with existing materials. However, even after trying these strategies, it is generally impossible to determine exactly what composition and processing specification will meet the targets [47].

**Testing and estimation of properties**

After a group of possible formulations are identified likely to meet the specifications, tests and analysis of the data give rules that are more accurate and provide the basis for a finer set of experiments in the next iteration.
2.4 Approaches to Formulation

In this section, the knowledge a formulator draws upon when formulating PUs is described. In Figure 1-2 (p.11) a PU's layered structure schematically represents the three classes of knowledge that characterise a PU formulation and which directly affects final properties i.e. the composition, structure, manufacturing or processing. The core layer corresponds to the compositional layer and as its name suggests, it corresponds to chemical components (i.e. chemical ingredients) used in a formulation, nature of reactants (e.g. aromatic, aliphatic) and functionality of components. The effects of these and their influence on the structure and the final properties are described in section 2.4.2. The processing layer relates to the knowledge and operating conditions such as mixing ratio, temperature of the mould, pressure and method of mixing that influence the final properties of the product manufactured with the inherent complexity of variations in PU formulation due to environmental changes. Sandham [73] has done some work on this area. He studied the effect of climatic conditions e.g. altitude (i.e. atmospheric pressure), humidity on foam physical properties. These variations, however, are considered outside the scope of this thesis, so they will only be briefly examined in section 2.4.1. This aspect will have to be integrated with future development.

The aims of this section are (i) to revise the complexity and variety of knowledge, information and data involved in the development of formulations by studying specific examples in the literature and (ii) to recognise how the PU formulation design options become complex and hence the competences to formulate (e.g. knowledge) evolve into expertise which becomes difficult to structure and deploy for future problem solving.

2.4.1 PU Processing Variations

There is an extensive range of manufacturing process for the production of PU materials i.e. for different PU applications various manufacturing processes can deliver a suitable product. A decision as to whether one process is chosen over another usually depends upon economic factors, production cycle time, as well as what equipment is available locally, and more recently, upon sustainability and recycling issues. All of these factors are not always manipulated by formulators but instead are usually constraints within which the formulator has to work. This is because several industries adhere to only one production process. However, there is great potential to change formulations by varying processing parameters in a comprehensive way.

With the broad spectrum of PU types available, any conventional polymer-shaping process can be employed, from thermoplastic injection moulding to thermo-vacuum forming (e.g. with thermoplastic PUs–TPUs), conventional vulcanised rubber shaping techniques, to
thermoset composite methods [74, 75]. However, specific processes have been developed for PU materials, which only in a few cases are also employed for other types of polymer (e.g. in respect to reaction injection moulding (RIM), amide 6 copolymers, polydicyclopentadiene (PDCPD), acrylics, unsaturated polyester resins (UPRs), Phenol-formaldehyde (PF) resols. It must also be noted that these shaping methods are not only used for urethane-polymers but also for polyureas, (urethane/urea) copolymers, (urethane-isocyanurate) copolymers, along with a number of lesser types of urethane modified by a low percentage of a copolymer derived from secondary isocyanate reaction, e.g. biuret, amide, allophanate. There are also a range of processes where isocyanate (non-prepolymer type) is used to react with hydroxyl rich materials, e.g. as the cellulose molecular structure of wood chip binder [76]. The processing options for PUs are considerable and its detailed study is out of the scope of this thesis.

2.4.2 Reactant and Additive Variation

Compositional variables are known to affect the morphology of a PU e.g. degree of phase separation, phase mixing, and hard segment domain organisation, and hence the polymer properties [71, 77]. The large list of chemicals that influences the final product's properties for any type of formulation is outlined in Table 2-5. This is by no means complete and comprehensive; however, the objective is to illustrate the exponential growth of complexity when introducing a new compound in a formulation. Each one of these components is known to have an influence on the kinetics and on the reactivity of the system and hence on the final properties.

Each of the chemical variables and their influence on the properties of PUs has been the subject of numerous studies. See for example, studies on the structure-property relationships for PU elastomers in the Ph.D. thesis of Sykes [24], and the work carried out in the last decade on PU flexible foams by Wilkes and co-workers [25, 26, 78, 79]. The influence of structure on the properties has also been explained in terms of the soft-hard segment theory especially for elastomeric PUs [80-82]. Examples of specific changes introduced in the chemical structure of the main components in a PU formulation i.e. the isocyanate, the polyols and chain extenders are given below.
### Reactant and additive factors that influence PU formulation design

<table>
<thead>
<tr>
<th>Isocyanates</th>
<th>Polyols</th>
<th>Chain Extenders</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Different chemical types</td>
<td>1. Different chemical types</td>
<td>1. Different chemical types</td>
</tr>
<tr>
<td>1.1. Aromatics</td>
<td>i.e. variation of repeat unit in oligomer with broad choices from:</td>
<td>1.1.1 Linear aliphatic saturated diols</td>
</tr>
<tr>
<td>1.2.1. Linear aliphatic saturated diols - different aromatics</td>
<td>- polyethylene glycols</td>
<td>- 1,4-butanediol</td>
</tr>
<tr>
<td>1.2. Aliphatics and alicyclics - different types</td>
<td>- polypropylene glycols</td>
<td>- 1,2-propanediol</td>
</tr>
<tr>
<td>2. Purity</td>
<td>Ethylene glycol tipped-polypropylene polyols</td>
<td>- 1,3-butanediol</td>
</tr>
<tr>
<td>- 100% purity</td>
<td>- PTMG polyols</td>
<td>- 1.1.3. Linear aliphatic unsaturated diols (which contain carbon-carbon double and triple bonds)</td>
</tr>
<tr>
<td>- active secondaries (e.g. the eutectics, or polymeric/PAPI types)</td>
<td>- Many others, either derived from synthetic or natural occurring monomers</td>
<td>1.1.4 Aromatic diols</td>
</tr>
<tr>
<td>- inactive secondaries</td>
<td>- 1,2-propanediol</td>
<td></td>
</tr>
<tr>
<td>- synthesis contaminants, e.g. chloride</td>
<td>- 1,4-butanediol</td>
<td></td>
</tr>
<tr>
<td>- isocyanate isomers</td>
<td>- 1,4-butanediol</td>
<td></td>
</tr>
<tr>
<td>- stand alone</td>
<td>1.2 Polyester polyols</td>
<td></td>
</tr>
<tr>
<td>- blends</td>
<td>- caprolactone polyester polyols</td>
<td></td>
</tr>
<tr>
<td>- isocyanate blends</td>
<td>Many others, either derived from synthetic or natural occurring monomers</td>
<td></td>
</tr>
<tr>
<td>- isocyanate prepolymers</td>
<td>1.3. Polycarbonate polyols</td>
<td></td>
</tr>
<tr>
<td>3. Eutectics</td>
<td>1.4 polyester polyamines</td>
<td></td>
</tr>
<tr>
<td>Note: then the molecular/morphology influence of the repeat unit, e.g. in-chain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Blends</td>
<td>- 1,4-cyclobutanediol and - 1,4-di(butyloxy)methylene</td>
<td></td>
</tr>
<tr>
<td>4.1 Co-isocyanates</td>
<td>- 1.1.5 Mesogenic diols chain extenders</td>
<td></td>
</tr>
<tr>
<td>4.2 Co-isomers</td>
<td>- 4,4-bis(3-hydroxyalkoxy) biphenyls</td>
<td></td>
</tr>
<tr>
<td>5. Functionalities</td>
<td>- Benzene- 1,4-di (iminophenoxy-n-hexanol)</td>
<td></td>
</tr>
<tr>
<td>2 to 3 typically: dependence on purity or eutectic types</td>
<td>1.1.6 Heterocyclic diol chain extenders</td>
<td></td>
</tr>
<tr>
<td>2. Purity</td>
<td>1.2 Diamines</td>
<td></td>
</tr>
<tr>
<td>- polydispersivity</td>
<td>1.2.1 Primary amines</td>
<td></td>
</tr>
<tr>
<td>- blended to give nominal mol wt</td>
<td>- Aliphatic</td>
<td></td>
</tr>
<tr>
<td>- secondary chemicals</td>
<td>- Aromatic</td>
<td></td>
</tr>
<tr>
<td>- residual catalysts</td>
<td>1.2.1.1. Reduction of their reactivity by</td>
<td></td>
</tr>
<tr>
<td>- water content</td>
<td>- Incorporation of electron withdrawing substituents e.g. (chlorine, fluoride, ester, sulphur and nitro) e.g. MOCA</td>
<td></td>
</tr>
<tr>
<td>- monol content</td>
<td>- Incorporation of steric hindering substituents e.g. (methyl, ethyl, isopropyl and tertiary butyl)</td>
<td></td>
</tr>
<tr>
<td>- unsaturation</td>
<td>- Formation of complexes</td>
<td></td>
</tr>
<tr>
<td>3. Molecular weight</td>
<td>- Use of Secondary amines (also see catalysts)</td>
<td></td>
</tr>
<tr>
<td>- 300 to 10,000</td>
<td>1.3 Ionisable chain extenders</td>
<td></td>
</tr>
<tr>
<td>- polydispersivity</td>
<td>- Ionising agents</td>
<td></td>
</tr>
<tr>
<td>- blends to give nominal mol wt</td>
<td>- Quaternisation of tertiary amines</td>
<td></td>
</tr>
<tr>
<td>4. Functionalities 2 to 4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**2 Morphological influences**
- repeat unit
- differences introduced whether number of repeat units in aliphatic chain is odd or even
- mol wt
- aromaticity
- pendant groups
- ionizing agents
Blowing Agent
1. Mechanism
1.1. Chemical (exothermic - irreversible)
   e.g. chemical reaction between isocyanate and water.
1.2. Physical (endothermic - reversible) - Surfactants
   - CFCs (Chlorofluorocarbons)
     They are banned since 1995 because they are Ozone Depleting Substances (ODS)
   - HCFCs (Hydrochlorofluorocarbons)
     (phase out)
   - Chlorinated solvents e.g. Methylene Chlorides (phase out)
     - n-pentane
     - cyclopentane
     - Liquid CO₂
   1.3 Mechanical/frothing
   1.4 Mineral salt dispersions
   1.5 Hollow spheres

Catalyst
1. Types
1.1. metal soaps
1.2. Tertiary and reactive secondary amines

Additive
- Nucleation Agents
- Liquids
1.1. Silicone based polyols
1.2. Gas e.g. Air, Nitrogen
- Surfactants
  (for cellular development)
- Fire Retardants
- Pigments
- Reinforcements

2. Glass fibre
   - chopped, hammer, milled fibre
   - cloth, mat e.g. RTM

2.4.2.1 Polyisocyanates

Common isocyanates used as building blocks for PU include aromatics (containing a planar unsaturated ring of six carbon atoms), aliphatics (open chain with no aromatic rings) and alicyclics (cyclic structure with aliphatic properties). They can be used pure, or as a mixture of different isomers and their reactivities are dependent on their chemical structures (e.g. different rates of reaction with hydrogen active species). In general, aromatic isocyanates are more reactive than aliphatic ones. This reactivity can be explained by the resonance structure of the isocyanate group [60]. The principal commercial aromatic isocyanates include toluene diisocyanate (TDI), 4,4'-diphenylmethane diisocyanate (MDI) and polymeric MDI. Other types are employed for specialist elastomers. Non-aromatics are less reactive, more expensive and tend to be used for solid elastomers (e.g. service UV resistance).

Morphological differences of isocyanates have a marked effect on the PUs produced from them. For instance, 4-4 MDI is preferred for the preparation of segmented PU elastomers; 2,6-TDI produce segmented PU elastomers with better properties than MDI but economical commercial separation of the isomers has not yet been accomplished. TDI is mainly used in flexible foams, because of its high reactivity. In general, the diisocyanate structure governs the tendency of chains to orient within the polymer. Thus, it exhibits an important influence on the potential of van der Waals forces and hydrogen bonding within the polymer. Examples of the effects of diisocyanate symmetry and chain flexibility on the properties of PUs are given as follows.

Effect of Symmetry
In general, PUs of higher tensile strength, tear strength, and hardness are associated with diisocyanate molecules of greatest symmetry and rigidity. Experiments carried out by Pigott et al. [71] were designed to use equivalent stoichiometric amounts of the various isocyanates in order to draw conclusions on the effect of molecular structure of aromatic isocyanates on properties. They suggested that bulky, rigid molecules influence properties, and demonstrated the effect of reduced symmetry combined with methyl substituents. Where methyl substituents were present a general decrease in tensile modulus, tear, and hardness was found (see Table 2-6). They attributed the effect of methyl groups to the induced spatial separation of adjacent polymer chains, resulting in an increased mobility of local segments.

<table>
<thead>
<tr>
<th>Comparison between NDI and PPDI, and TDI</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>NDI: Naphthalene diisocyanate</td>
<td>Bulky, Rigid molecules</td>
</tr>
<tr>
<td>PPDI: p-Phenylene diisocyanate</td>
<td>Symmetric, No methyl substituents are present</td>
</tr>
<tr>
<td>TDI: Toluene diisocyanate</td>
<td>Reduced symmetry, Methyl substituents present</td>
</tr>
</tbody>
</table>

Table 2-6. Effect of reduced symmetry combined with methyl substituents in aromatic isocyanates

<table>
<thead>
<tr>
<th>Isocyanate</th>
<th>Tensile Strength (MPa)</th>
<th>Modulus at 300% elongation (MPa)</th>
<th>Tear Strength (N/m)</th>
<th>Hardness (Shore B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NDI</td>
<td>30</td>
<td>21</td>
<td>35025</td>
<td>80</td>
</tr>
<tr>
<td>PPDI</td>
<td>44</td>
<td>16</td>
<td>52538</td>
<td>72</td>
</tr>
<tr>
<td>2,4-TDI</td>
<td>32</td>
<td>2.4</td>
<td>26279</td>
<td>40</td>
</tr>
</tbody>
</table>

In general, an isocyanate's symmetry is affected by the introduction of methyl substituents. A reduction in symmetry leads to a decrease in tensile strength, modulus, tear and hardness. This effect can also be seen when comparing MDI, dymethyl-dyphenilmethane (DMDI) and diphenylisopropylidene (PDI). The same impact on properties is seen when increasing bulkiness and rigidity in isocyanate molecules. The bulky and rigid diisocyanates give products of highest modulus and hardness.

Effect of Flexibility

5 For polyester-based PU elastomers chain extended with 1,4-Butanediol.
Decrease in flexibility within isocyanate molecules results in increased hardness, tear strength, and modulus, and at the same time, a reduction in ultimate tensile strength, and elongation. This was indicated by Pigott et al. [71] when dimethyldiphenylmethane diisocyanate (DMDI) elastomer was compared with bitolylene diisocyanate (TODI) (refer to Table 2-7). Additionally, the influence of the structure of these isocyanates on the glass transition temperature of the urethane elastomers was affected only slightly when compared to other changes.

Table 2-7. Effect of reduced flexibility on properties using aromatic isocyanates

<table>
<thead>
<tr>
<th>Comparison between DMDI and TODI</th>
<th>DMDI: dimethyldiphenylmethane diisocyanate</th>
<th>TODI: bitolylene diisocyanate</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Flexible due to the methylene bridge</td>
<td>Less flexible</td>
<td></td>
</tr>
<tr>
<td>DMDI</td>
<td>36 4</td>
<td>28 16</td>
<td></td>
</tr>
<tr>
<td>TODI</td>
<td>15748 47</td>
<td>31496 70</td>
<td></td>
</tr>
</tbody>
</table>

Table 2-8. Effect of Isocyanate structure on the mechanical properties of PU elastomers [71]

<table>
<thead>
<tr>
<th>Aromatic Isocyanates Structure</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Tensile Strength</td>
</tr>
<tr>
<td>Bulkiness &amp; rigidity ↑</td>
<td>Decrease</td>
</tr>
<tr>
<td>Symmetry ↓ e.g. introduction of methyl groups</td>
<td>Decrease</td>
</tr>
<tr>
<td>Flexibility ↓</td>
<td>Decrease</td>
</tr>
</tbody>
</table>

2.4.2.2 Polyols

Polyols are oligomeric materials i.e. molecules having intermediate relative molecular weight from about 200 to 12,000 depending on functionality i.e. number of reactive hydroxyl (OH) groups. Many different polyol structures are available, esters and ethers are the two major
Polyurethane Formulation

types, although other polyols (e.g. acrylics, phenolic, carbonate) are used to a lesser extent in special PU applications such as coatings. The polymeric structure of the polyol introduces gross changes in the properties of PUs, as is detailed below.

**Effect of Flexibility by Introduction of Side Chains**

Flexibility of PU chains is affected by the introduction of methyl side chains and ester group spacing (repeating unit chain length). In general, tensile strength and modulus are more functions of the presence of side chains than of ester group separation while tear strength appears to be a function of both of them. For example in the manufacture of PU elastomers, commercial hydroxyl-terminated polyethers, such as poly(oxytetramethylene) glycol, give elastomers having the best mechanical properties, a fact which is due in part to the regularity of chain structure and therefore on the ability to crystallise upon extension. The side chain methyl group in poly(oxyethylene) glycol prevents crystallisation of the flexible segments and increases interchain separation contributing to the lower level of mechanical properties of elastomers obtained from the poly(oxyethylene) glycol. Pigott et al. [71] prepared elastomers from MDI extended with 1,4-butanediol for stoichiometrically equivalent formulations using a series of polyesters based on adipic and succinic acid. They found that tensile strength and modulus were significantly lower for the elastomers prepared from adipate polyesters with side chains, than for the linear ones. Similar observations were found for the polyesters made from succinic acid.

Tear strength was apparently dependent on both, the presence of methyl side chains and ester group separation. It decreased significantly in the elastomers made with poly(1,5-pentaethylene adipate) and with poly(1,3-butylene adipate) as compared to those made from poly(ethylene adipate). However, the data were too variable to enable conclusive information on the relationship to be determined. Closer ester group spacing reduced flexibility and at room temperature favoured higher hardness values, higher modulus and increase in permanent elongation. These results may be attributed to increased van der Waals attractive forces [71].

Additionally, variations on the structure of polyesters, especially ester group separation variation influences low temperature flexibility [71]. Closer spacing in ester groups reduces flexibility at low temperatures, and in polyesters of adipic and succinic acids results in the glass transition occurring at higher temperature.

In conclusion, if the polyester is prepared from intermediates that are symmetrical and contain rigid ring structures, then harder PU elastomers are produced due to reduced polyester chain flexibility.
Table 2-9. Effect of structure of polyesters derived from adipic and succinic acids on the mechanical properties of PU elastomers. Reference [71].

<table>
<thead>
<tr>
<th>Change in structure of Polyesters derived from adipic and succinic acids</th>
<th>Properties</th>
<th>Tensile Strength</th>
<th>Elongation</th>
<th>Modulus at 300% elongation</th>
<th>Tear Strength</th>
<th>Hardness (Shore B)</th>
<th>Tg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Side chains †</td>
<td>Decrease (54%)</td>
<td>Small tendency to decrease</td>
<td>Decrease (29%)</td>
<td>Decrease (58%)</td>
<td>Little effect</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ester group separation †</td>
<td>Little effect</td>
<td>Small tendency to decrease</td>
<td>Little effect</td>
<td>Decrease (75%)</td>
<td>No effect</td>
<td>Decrease</td>
<td></td>
</tr>
</tbody>
</table>

2.4.2.3 Chain Extenders

Chain extenders are usually di-functional hydroxylated compounds, such as glycols and diamines or hydroxylamines (Table 2-10) used in polyureas and (urethane/urea) copolymers.
The properties of PUs are known to be greatly influenced by the choice of chain extender. For instance, PU elastomers prepared from poly(ethylene adipate) and MDI, and extended with a homologous series of aliphatic glycols were studied by Pigott et al. [71]. It was found that the optimum balance in properties was obtained with 1,4-butanediol, a chain extender also widely used in many PU foam flexible formulations. Additionally, they found lower tensile strength along side higher tear strength when change in structure of polyesters derived from adipic and succinic acids was introduced. The authors attributed these results to greater rigidity of aromatic diols.

Similarly, properties of PU elastomers synthesised using primary diamines possess enhanced mechanical properties compared to those from diols with equivalent structures as reported by Sung [81, 82]. This was due to an improvement in the extent of phase segregation.
suggested by a lower transition temperature (or Tg) of the soft segment phase and a much higher Tg of the hard segment domains in (urethane/urea) copolymer extended with the Diamine (ethylenediamine) rather than with the diol (1,4-butanediol).

2.5 Summary and Conclusions

PUs have versatile chemistry. By changing the nature of the main reactants a wide range of structural changes, which in turn affect the final properties of the polymer are found. PUs consisting, in particular, of alternating soft segment (aliphatic polyether or polyester) and hard segment (aromatic urethane), offer unique possibilities for tailor-making their properties by varying their composition. The effects on the end-properties of the PUs (e.g. thermoplastics and PU elastomers) belonging to this group that affect the macrostructure of the polymer have been analysed by means of various techniques. These characterisation techniques (e.g. thermal transition analysis, both wide-angle and small-angle X-ray studies, and infrared studies) are used to account for changes in morphologic features (i.e. degree of phase segregation, phase mixing, and hard segment domain organisation among others) to understand the correlation between composition-structure, manufacturing-composition, and manufacture-structure relationships effects on the materials’ end-properties.

Reviewing the systematic studies described in the open literature has highlighted the complexity of the formulation task. Although the literature reviewed offers a source of diverse knowledge on the field, this knowledge is not comprehensive enough to provide a source of expertise to support non-expert formulators. This is in part due to the intrinsic complexity of the formulation task due to the overwhelming number of variables, their unknown interrelationships and their effects on the end-properties. These problems are further compounded by the lack of vital information in the reported results. For example, most studies lack a rigorous approach as they report on brand names when referring to the raw materials used instead of chemical names

2.6 Motivation for a Computationally-Based Approach

It is clear that the PU formulation process is a highly complex problem. The degree of success of solving the problems arising in the development or modification of PU formulations depends on the choice of the raw materials, the processing, and to a large extent on the skills of the formulator. The traditional human-based approach to formulation often lacks a

7 This is an adverse disadvantage that has been accompanied most fundamental PU studies, ranging from reports on books and journal papers.
Polyurethane Formulation

consistent and systematic procedure. As a consequence, formulations for a variety of applications have been generated in an ad hoc manner leading to a lack of understanding of the mechanisms that could explain how and why the formulations work. This in turn has impeded the development of improved formulations and also it has made the formulation task even more complex and difficult to unravel.

In summary, the PU formulating process is characterised by the following features:

1. The PU formulation process involves reasoning about experiential knowledge involving micro and macro structure, abstract concepts and heuristic “rules of thumb” based on experience. This reasoning is highly symbolic rather than numeric.

2. No predetermined algorithm is used in formulation. By contrast, formulators are guided through heuristic rules and partial models (there are no complete models due to the vast space of design variables). They normally start to reformulate by using a similar formulation to the design target.

3. The reasoning depends on the availability and reliability of knowledge about the problem, at the time this is specified.

4. Quantitative models for calculating the mechanical properties of a proposed formulation are often not available.

For these reasons, and to provide PU manufacturers with the ability to represent, share and apply collective experience for future progress, an artificial intelligence (AI) approach seems appropriate for the PU formulation problem. The use of knowledge-based systems technology could benefit the PU industry in many ways, mainly by:

1. Increasing the quality of the formulations, this could be achieved by enforcing a consistent approach, free of the biases of individual formulators.

2. Reducing time on formulating process. The system would improve the formulation with an interpretation of the process resulting in a more consistent, efficient and less costly formulation performed by better trained employees.

3. Accomplishing training objectives. This could be accomplished by supporting the formulation task by use of knowledge-based systems that can provide a training vehicle to less experienced persons operating in the field.

4. Other advantages include improving the industries’ public image that will directly benefit by increasing the quality of the formulations and training of the formulators.

Evidently, practical experience and knowledge of PU formulators and PU experts cannot be readily converted into an automated system. Therefore, the focus of the present work has been on the codification of available formulation data following the best practices and
solutions explored in AI that can assist formulators in developing their own solutions. These are reviewed in the next chapter.
Chapter 3

KNOWLEDGE-BASED SYSTEMS

How can the lessons learnt in artificial intelligence be used to support the formulation of PUs?

Abstract: Several features make polyurethane formulation design a natural artificial intelligence application. In particular, the case-based reasoning technique provides an appropriate methodology in dealing with complicated problems arising in PU formulation. The application of this methodology to support the PU formulation process is described here. When using CBR problem solving experiences are collected in the form of cases that are then reused to solve new problems. It was found that PU formulations could be represented as feature vectors that contain a problem description and its solution. This case representation also facilitates retrieval, which is commonly done in the CBR field by using a nearest neighbour classifier that uses a distance function to assess the similarity between two cases. The use of a particular distance function depends greatly on the type of attributes.

Key words: Artificial Intelligence, Knowledge-Based Systems, Expert Systems, Case-Based Reasoning, Knowledge Representation, Retrieval, Similarity Assessment, Adaptation, Artificial Neural Networks.

3. CHAPTER 3. KNOWLEDGE-BASED SYSTEMS

3.1 Problem Statement and Objective of this Chapter

In the previous chapter, the complexity of the PU formulation task and also the motivations for a computational approach to solve the issues arising in PU formulation were presented. The available computational approaches to assist the PU formulation process are reviewed in this chapter. The problem to solve in this chapter can be stated as:

*Given a set of experimentally obtained PU formulations find a computational methodology that enables the reuse of this accumulated knowledge and expertise to develop new PU formulations.*

The objective of this chapter can be met if the following research question is answered:
3.1.1 Research Question 1

What computational methodologies are available for dealing with complicated problems arising in PU formulation? What are the basic assumptions for using these techniques? What technique is the most appropriate to support the formulation of PUs?

3.2 Methodology for Research Question 1

For successful knowledge management, the ideas and techniques that have been developed by the Artificial Intelligence (AI) community are increasingly required in complex domains such as engineering, business management, medicine, and design and wherever an organisation wants to enhance its ability and capacity to perform, compete, and cope with change [83]. This is mainly because information technology enables the practical capture, sharing and control of information and knowledge throughout organisations that is seen fundamental for progress.

The history of the AI field is reviewed first in section 3.3. This is followed by a description in section 3.3.2 of how knowledge in the form of human expertise has been "computed" (i.e. represented, structured and utilised) in expert systems that intend to solve problems or give advice. The selection of a methodology that best suits the problem statement defined previously (section 3.1) can then be identified. The selected methodology is thoroughly described in section 3.4 along with pertinent issues to enable its best implementation across design domains. A summary in section 3.6 and the answer to the research question that motivated this chapter closes this chapter.

3.3 What is Artificial Intelligence?

It is difficult to suggest a single definition for AI, as the field has many facets. Its philosophical aspect, devoted to the study of ancient problems about the relationship between the mind and the body; its scientific facet, in where researchers are trying to provide tools for understanding cognition in humans and animals; its mathematical facet, to formulating and analysing effective algorithms proving computers with some of the functions of intelligent activity; and its technological aspect, that provides tools for solving practical problems in various domains (e.g. medicine, law, engineering*).

AI is a form of advanced computing [84] decomposable into various sub-disciplines including: Artificial Neural Networks, Natural Language Processing, Robotics, Game

---

*The tools that have been used for solving problems in the engineering domain are explored further in this chapter.
Playing, and Expert Systems. In the latter, software tools are programmed in order to support the decision-making in various domains such as chemistry, engineering, medicine, law, and business among others. The computational methodologies that can be used to assist the formulation of PUs are reviewed in the next section after a brief AI history review.

### 3.3.1 AI History

The term “Artificial Intelligence” was coined for the first time in 1956 at the Dartmouth Conference [85 p.9]. This conference gathered researchers interested in how to describe and simulate aspects of human learning and any other feature of intelligence using a machine. From 1956 until about 1960 was termed the Prehistoric Time [86]. This was because computers were not available to enable the testing of experimental ideas. Around 1960, the Dawn Age [86] started, a period in which extensive predictions about AI were made. In the next period (1965-1970), the Dark Period [86] little progress in AI occurred. Then, from the period between 1970 and 1975 (the Renaissance [86]), MYCIN [87] and other seminal AI systems were developed. Artificial Intelligence began to specialise into areas such as expert systems, language analysis, knowledge representation and computer vision. However, this specialisation was regarded as a detrimental fragmentation of the AI field. This period was followed by the Age of Partnerships (1975-1980) [86] as researchers from multiple disciplines became increasingly involved in the field of AI. Then from 1980 until about 1990, the Age of the Entrepreneur [86], several companies started to commercialise intelligent systems. The subsequent decade was named the Age of Adjustment in [88] as for the need of the AI field to readjust its position to re-define expectations of AI at both long and short term. However, this period has become known as the AI winter of the 1990s. This has been due to little progress in AI field, although advances in many areas such as computer vision, speech recognition, text analysis, robot control, and automatic planning were witnessed [89]. In the author’s point of view, the AI field should start an Integration era. Advancements in many AI sub disciplines, computing and electronics should all be brought together to support the challenges faced by humanity in the 21st century.

### 3.3.2 Expert Reasoning and Expert Systems

The terms expert systems (ES) or knowledge-based systems (KBS) emphasise the role of knowledge in the form of human expertise in such systems that intend to solve problems or give advice [90]. Boundaries between KBS and other problem solving programs are not sharply defined. Therefore, some elaborated programs might wrongly be classified as ES, and
the opposite can also be true, for instance when authors might not choose to characterise their programs as expert systems.

In order to distinguish a KBS from a complex software program, certain characteristics based on the type of problem a system is solving, and the form in which knowledge is encoded can help in identifying expert systems. In general, KBS do not need to function as experts in order to be useful. This is because most KBS can also be built to function as both assistants and peers. An abstract view of a KBS allows it to be categorised into a problem solving methodology composed of two elements, domain knowledge and problem-solving methods (i.e. specific ways to deal with a problem) [90]. Therefore, a KBS can be distinguished from a complex program in which is a computer system that uses knowledge and reasoning techniques to solve problems that would require human expertise [90].

For the computational implementation of expert systems two classical models of human problem solving exist. One model proposes that a human expert reasons concatenating a series of rules and makes inferences based on those rules to understand and solve a problem. Therefore, most of an expert system’s domain knowledge built under this model would consist of a set of rules. This premise was derived historically from Newell and Simon’s pioneering work on the general problem solver (GPS) [91] one of the first AI programs that led to the first Rule-Based System (RBS) [92, 93].

The other well known model of human reasoning is one where human experts solve new problems by adapting solutions that were used to solve old problems. Therefore, an expert system’s domain knowledge built under this model would be composed of a set of experiences that can be retrieved and adapted to solve new problems (Figure 3-1). This paradigm of problem solving is referred to as case-based reasoning (CBR).

![Figure 3-1. Case-based reasoning. Source: [94]](image)

Expertise comprises experience. When solving a new problem, experts rely on past episodes. Case-based reasoning is a general paradigm for reasoning from experience.

3.3.2.1 Rule-Based Systems

Historically, the first technical approach to building KBS used a vast set of if-then rules as the system’s knowledge. The system, DENDRAL, [92, 93] was the first successful rule-based system (RBS). The program intended to simulate the work that an organic chemist expert
would do in analysing mass spectrometry data. Afterwards, numerous KBS came about to handle several functions ranging from diagnosis of diseases (e.g. MYCIN [87]) to analysing geological data (e.g. PROSPECTOR [95]).

In a broad sense, a RBS consist of a knowledge base and an inference engine (see Figure 3-2). The knowledge base contains rules and facts than have been elicited from domain experts. The inference engine is the reasoning structure that uses and controls the application of knowledge to meet the purposes of the RBS. For instance, when all the initial facts are known but the conclusions are not, forward or data-driven rule systems are used. In forward chaining systems, rules represent possible actions to take when specified conditions hold on.

If the application requires a goal to be probed or a hypothesis to be tested instead of drawing conclusions from data, backward chaining RBS are used. In a backward chaining system, given a goal state to try and prove, the system will first check to see if the goal matches the initial facts given, and keep looking for rules until it finds evidence to conclude that hypothesis.

![Figure 3-2. The basic features of a forward chaining RBS. Source: [95]](image)

The problem solving with RBSs involves an iterative cycle of (i) identifying from the rules and fact (i.e. experience) and (ii) applying those rules to solve or simplify the problem. In forward chaining RBS, the system first checks to find all the rules whose conditions hold, given the current state of working memory. It then selects one and performs any actions in the action part of the rule. The actions update the working memory, and the cycle begins again. This cycle stops when either no rules are activated, or some specified goal state is satisfied.

### 3.3.2.2 Case-Based Reasoning (CBR)

Case-Based Reasoning emerged as an alternative to RBS. Arguments presented in [94], suggest that the ability of CBR systems to extract information from its experience, and
acquire knowledge on its own, was crucial for the long-range success of the expert system concept in AI. This allowed for a rapid growth in CBR applications in the 1980s [96 Ch. 2]. Current CBR systems take advantage of the strengths of different paradigms and technologies that act synergistically rather than competitively to enhance CBR systems.

Roger Schank and his group at Yale University conducted early research on CBR in the 1980s. They generated both a cognitive model for CBR and the first CBR systems based on this model. Janet Kolodner developed the first CBR system called CYRUS (Computerized Yale Retrieval and Updating System) [28, 97]. This CBR later served as the basis for the implementation of several other CBR systems in diverse domains including MEDIATOR (mediation), CHEF (meal planning), CASEY (medicine), JULIA (meal planning), among others. Compiled research carried out to bring about these systems is presented elsewhere [96 Ch. 2]. In addition, CBR has expanded to other application domains where the analysis or diagnosis and the recommendation or prescription of how to solve problems is a crucial task e.g. (help desk support, e-commerce [98, 99])

The basic idea of a CB reasoner is to solve problems by adapting past solutions i.e.

"A case-based reasoner solves new problems by adapting solutions that were used to solve old problems" [96]

Conceptually, CBR is commonly described as a cyclic process [100] (see Figure 3-3). First, a problem situation or query is analysed and indices (i.e. labels) can be assigned to key problem features. If indices are assigned, matching of those suitable cases is performed and then retrieved from the case-base. Otherwise, a similarity measure between the query and the cases in the case-base is computed so that the most similar cases to the query are retrieved. The information in the retrieved case is then used to provide an initial solution to the problem. If this does not fully satisfy the problem specifications, the retrieved case’s solution is adapted. This adaptation can be done by using “adaptation knowledge” which includes domain rules, heuristics or human expertise [28]. Once the retrieved case has been adapted, this candidate solution is tested. If the test is not successful, sources of failure are investigated and the partial solution is repaired and tested again until eventually a suitable solution is found.
Each stage in the CBR cycle is associated with tasks. For instance, the retrieval stage needs to identify relevant descriptors, search the case library for past cases, match and select the most similar case. Aamodt and Plaza [101] described a task-oriented view of a CBR system. They distinguished between CBR “top-level” tasks that are partitioned into “subtasks” and “methods” (see Figure 3-4). This task-oriented view allows identification of the detailed mechanisms or methods that are likely to be used in the CBR system.
Figure 3-4. A task method decomposition of CBR. *Source:* [101].

Each CBR top-level task is partitioned into subtasks (in bold letters) and methods (in italic), which allow identifying the detailed mechanisms or methods that could be used to achieve the system's goals. The relation between tasks and methods (stippled lines) identify alternative methods applicable for solving a task.
3.3.3 Choice of the Computational Methodology

That CBR is a more appropriate methodology than RBS to support the problems arising in the formulation of PUs is argued in this section. Before presenting the arguments, (based on a former discussion given in chapter 2), the problems arising in PU formulation are described.

The problems arising in the formulation of PUs are concerned mainly with the development of formulations that meet a set of constraints. This relates fundamentally to the synthesis of a formulation (a basic example in which processing parameters are not taken into account, the formulation problem would be to propose ingredients and their quantities) that achieves the desired properties in an economic way.

These problems and their different solutions were analysed in chapter 2. In essence, it was reviewed that PU experts develop new formulations by drawing upon various kind of PU knowledge (formal e.g. partial mathematical models and informal e.g. rules of thumb) and guide their search for a solution through heuristic reasoning. This is, experts tackle new problems based on their past experience and intuition (knowledge that cannot be formalised).

Although experts use various kind of knowledge (such as causal rules) that they develop through experience, the PU field lacks a comprehensive knowledge base. This has been mainly due to commercial secrecy that impedes knowledge sharing and publication of information in a tractable manner. In addition, this has led to a perception and practice of the process of formulation as an art, more rather than as a science, such that after fifty years of heritage the domain still remains poorly understood.

One of the main reasons of using CBR to support PU formulation is that the model of reasoning that CBR uses is analogous to the way formulators solve problems. A formulator when solving a problem naturally starts by reasoning about past experiences and tries to bring the closest similar experience back as a base to solve the problem at hand (i.e. essentially retrieval in CBR). After this selection of similar formulations to the target, a process of trial and error is initiated guided primarily by their experiences in the field (for example experts memorise errors made in the past and their corrections and they recall them when needed [47]). They alter the product properties by normally making changes in the formulation composition and/or processing parameters (this is essentially adaptation in CBR) until they eventually find a formulation that meets the requirements.

Although experts also use causal reasoning to guide their problem solving approach, a RBS does not cope well with the complexities intrinsic to PU formulation. For example, PU development is an open, dynamic and weak theory domain. It is open in the sense that involves a vast base of knowledge, dynamic because this knowledge base changes over time and the term weak is normally used to refer to theories where there is a high degree of
uncertainty between the dependent and independent variables that are used to model the domain [102]. Therefore in open, dynamic and weak domains, knowledge elicitation in the form of causal rules is a difficult task that can be achieved to a certain extent to support, for instance low-knowledge demanding tasks in a KBS for PUs, but cannot be proposed as a unique framework for PU problem solving.

Other advantages of CBR over RBS (see for example [94 p.48-50], [103], [104 ch.1 p2-4], [105 p234]) are listed as follows:

1. **Reduction of knowledge acquisition task when compared to RBS.** The knowledge acquisition task is eased in CBR by eliminating the need to extract a set of rules as required in RBS. However, as it will be explained in chapter 5, the adaptation stage in CBR is knowledge intensive and hence requires that this premise is totally valid only for CBR systems that do not perform adaptation.

2. **Avoiding repeating mistakes made in the past.** By definition CBR systems must learn. By recording both successful and unsuccessful experiences and for some diagnostic CBR the reasons of those outcomes, a CBR can be used to predict potential failures in the future based on past experience. However, it can be argued that a CBR system can deceive the user into believing that the retrieved case is the best that could have been retrieved. However, in principle, CBR makes use of experiences to remind of a likely similar solution to solve a problem instead of solving it from first principles.

3. **Learning over time.** If the solution cases proposed by CBR systems are tested in the real world, knowledge about the correctness of the proposed solution is obtained. Therefore, new cases can be added to the case base and be used in future problem solving while also anticipating and preventing errors.

4. **Reasoning in a domain with a small body of knowledge.** For domain problems in which a few cases are available, a CBR system can build its knowledge incrementally as the cases are added and hence increase its efficiency over time.

5. **Reasoning with imprecise data.** Soft computing tools are being used to support CBR tasks to cope with uncertain and imprecise data [105, 106]. Although CBR retrieves cases that might not be identical to the current problem the technique offers an alternative solution with various degrees of correctness (depending on the similarity measure used). Rule-Based Systems are not as flexible as CBR systems in this respect as they reason by concatenating a series of rules, and in the event that one is not valid, the system simply cannot obtain a solution.

6. **Providing a means for explanation.** A CBR user finds it easier to explain how a solution was arrived at by using the similarities between the cases and the reasoning involved in adaptation.
Although CBR presents many advantages over RBS, the integration of these two techniques has illustrated many advantages to the problem solving task [107-109]. The resultant systems are usually referred to as hybrids (i.e. systems that combine synergistically two methodologies or techniques). Marling et al. [109] details some of the reasons for integrating CBR with RBS and other reasoning modalities and the techniques to enable the integration. The options are considerable and their implementation mainly depends on the complexity of the problem, the tools available for knowledge representation and the system's purpose (or what is required of the system).

More recently, the use of soft computing techniques has had great impact in the development of hybrid CBR systems [105, 106]. Soft computing has been defined by Zadeh [110] as a group of methods and techniques that cope with uncertainty, imprecision and ambiguity i.e. they deal with reasoning that is approximate rather than exact. Pal and Shiu [106] present the foundations of Soft CBR. These and other prevailing research topics investigated by the CBR community to develop CBR systems to assist problem solving processes in similar domains to PUs are discussed in the following section. The essential components and processes involved in CBR and the topics currently investigated by the CBR community to develop CBR systems to assist solving processes in similar domains to PUs are also reviewed.

### 3.4 Case-Based Reasoning

#### 3.4.1 What is CBR?

As a methodology of problem solving, CBR implies what a case-based reasoner does (i.e. the method) but not how it does what it does (i.e. the technique). CBR uses former experiences to solve new problems [96]. The answer to the “how” question (i.e. methods and techniques to accomplish its tasks) is still a research topic very much in flux. This has partly been due to the increasingly new software engineering capabilities (e.g. use of graphical interfaces, powerful single-user power stations). Special focus has been given to the use of these capabilities to implement soft computing techniques (e.g. neural networks, genetic algorithms) to effectively accomplish retrieval and adaptation CBR tasks [106].

In particular, integration of CBR with soft computing techniques have been classified as soft CBR in [106]. Thus, soft CBR include the use of soft computing techniques (i.e. fuzzy logic (FL), evolutionary computing (EC), artificial neural networks (ANN), probabilistic reasoning (PR), and rough set theory) that can potentially be used in the following CBR tasks [105]:
1. Retrieval: case-based inference using fuzzy rules, connectionist indexing, fuzzy clustering and classification of cases, fuzzy feature weights learning, fuzzy indexing, fuzzy retrieval of cases, genetic algorithms for learning cases similarity, neural fuzzy techniques for similarity assessment, probability and/or Bayesian models for case selection, rough set based methods for case retrieval.


The use of one technique or other usually depends on the application domain. For example, several issues need consideration in order to choose whether one technique or another or a combination of techniques meets the purpose of a CBR for a particular application. For the application of CBR for the formulation of PUs, these issues basically encompass:

1. Case and knowledge representation
2. Retrieval and assessment of similarity between a problem and the cases in the case base
3. Adaptation of the retrieved solution to fit a problem

The approaches that are being studied to address the issues of case representation are reviewed in section 3.4.2. The retrieval approaches used in product formulation applications are studied in section 3.4.3. The issue of adaptation is reviewed in Chapter 5. Other issues that are beyond the scope of this work and therefore not addressed here relate to CBR learning (i.e. the revision and retain task).

3.4.2 How to Represent Knowledge in CBR?

Case-Based Reasoning can be viewed as a problem solving methodology that emphasises the combination of what is similar from the past, and how to use it to solve problems. In this view, a CBR system's knowledge consists of both past experiences or cases and "other" domain knowledge (Figure 3-5).
An expert's experience includes a wide range of information and knowledge that typically has many forms. A CBR's knowledge base consists of cases and other domain knowledge. Cases refer to experiences acquired over a period of time in a particular activity. Other domain knowledge refers to a collection of many different kinds of knowledge about the activity that can effectively support the CBR system operation. This integrates, for example, descriptive definitions of specific terms, description of objects and their relationships to each other, criteria of making decisions (e.g. rules, heuristic knowledge), etc.

3.4.2.1 Cases

There has been much debate about what constitutes a case. There is a growing body of research, which suggests that a case is the relevant information required to tell an experience by itself. In considering what information a case needs to include, the first point to consider is what is meant by the term experience.

An experience refers to the knowledge acquired over a period of time in a particular activity [96 p11]. According to Riesbeck and Schank [96 p11], the issue is whether something is best called a case, or a rule, or a story i.e. a case stands alone as an exception to a rule until numerous cases just like it are encountered. When this happens, very specific rules can be inferred about the data.

A case in CBR systems has been defined as "a contextualised piece of knowledge representing a previous experience or problem" [28]. Information recorded about such an experience depends both on its context and on the case-based reasoner's purpose. For case-based reasoners that assist problem-solving, a case has two major parts: a problem description and its solution. However, other features such as the methodology followed to reach a solution, and a measure of success of the applied solution in real life (outcome) can also be
included. In general, a case needs to include information relevant to the purpose of the system.

For instance in CHEF, one of the first CBR programs in the area of planning [112], a case description is comprised of goals and constraints. CHEF solution descriptions are plans. More specifically, a plan in CHEF was a recipe for preparing a complex dish (Figure 3-6). CHEF's case representation is in predicate form. The problem is described as a set of dish constraints. The solution is the recipe (or formulation) including the ingredients' type and quantity, and the actions (process steps) to plan a dish.

Problem:

(include tofu)
(taste hot )
(style stir-fry)

Solution:

(ingredients
  ingr1 (tofu lb .5)
  ingr2 (soy-sauce tablespoon 2)
  ingr3 (rice-wine spoon 1)
  ingr4 (cornstarch tablespoon .5)
  ingr5 (sugar spoon 1)
  ingr6 (broccoli lb 1)
  ingr7 (r-pepper piece 6)

(actions
  act1 (chop object (ingr1) size (chunk))
  act2 (marinate object (results act1)
        in (& (ingr2) (ingr3) (ingr4) (ingr5)
        time (20))
  act3 (chop object (ingr6) size (chunk))
  act4 (stir-fry object (& (result act2) (ingr7))
        time (1))
  act5 (add object (result act3 to result act4))
  act6 (stir-fry object (result act5) time (2)))

(style stir-fry)

Figure 3-6. One of CHEF's cases: broccoli with tofu. Taken from [28 p.172].

3.4.2.2 Other Domain Knowledge

In several applications, the knowledge base of a system also uses other domain knowledge besides cases to support effectively the system (refer to Figure 3-5). Other domain knowledge refers to a collection of many different kinds of knowledge that can be very specific (e.g. rules, facts) and general (e.g. heuristics or rules of thumb, models). Cases and domain knowledge can interact at different levels or stages in the CBR cycle. For instance, when similar cases are collected to a certain number, rules can be inferred and thus employed to
guide adaptation of cases, this inductive process has been referred to as adaptation guided retrieval [113]. However, for sparse datasets, these inductive learning algorithms would only provide very specific domain rules that cannot support the knowledge intensive adaptation tasks effectively.

3.4.2.3 CBR's Knowledge Representation

There are different kinds of knowledge a system usually draws upon about a particular domain with various representation formats. Historically, particular views of the nature of intelligent reasoning in such fields as, mathematical logic, psychology, biology, statistics and economics led, respectively, to five different representations technologies [114], namely logic, semantic nets, connectionism, causal networks, and rational agents. However, knowledge representation has become by itself an AI sub-discipline. Several articles have reviewed the multiple options available to represent knowledge in expert systems [111, 115]. For instance, in [111], the authors describe frame-based representations indicating integration with other formats (e.g. production rules) and the shortcomings of combining representations. A more recent review [115], gives examples on the major representations used in AI applications.

However, experience with RBS development made it clear that none of the major knowledge representation languages is by itself able to satisfy a system's and user's needs [114, 116]. Therefore, a combination of representations, that integrates effectively different representations to benefit synergistically from them, normally results in a coherent knowledge base. Particularly in CBR systems several representation formats have been used to represent both cases and other domain knowledge. Some of these representation formats include: database formats, frames, objects and semantic networks [117]. In general, a representation format for CBR systems should:

- enable efficient retrieval so that less effort is required in adapting a case;
- describe accurately the domain description, so that cases can be retrieved when required;
- allow good indexing, so that the retrieval can be done faster;
- be understandable for users so that analysis and interpretation of the knowledge can support the problem-solving task.

The most common used representation format in CBR is the spreadsheet format [7-10], where cases are represented as records, (or rows), and to each attribute variable is represented in an associated column. This format is also referred to as the attribute-value representation.
In particular, product formulation applications have used the attribute-value representation format [118, 119] to represent cases. The attribute-value representation has been the preferred format because formulations are normally presented as a list of ingredients and processing conditions with their corresponding values in formulation trials and therefore no information is lost when codifying formulation into cases.

Bandini and Manzoni [119] used the attribute value format to represent cases in the P-Race system. The system is a CBR tool that supports the formulation of Pirelli tyres in racing championships where Pirelli takes part. Craw and co-workers [118, 120, 121] have also favoured the attribute-value representation in their CBR systems for drug tablet formulations.

In the attribute-value representation a case problem and a case solution are represented in a vector. In most cases, attribute names are omitted as the context unambiguously determines which entry corresponds to which attribute.

For a particular case \( i \) the feature case vector \( \mathbf{c} \) is:

\[
\mathbf{c}^{(i)} = \{ c_1^{(i)}, c_2^{(i)}, \ldots, c_k^{(i)} \}
\]

(3-1)

The set of all \( \mathbf{c}^{(i)} \)'s constitutes the case-base (CB):

\[
CB = \{ \mathbf{c}^{(1)}, \ldots, \mathbf{c}^{(N)} \}
\]

(3-2)

However, for some CBR applications a vector representation solely has not been expressive enough for describing the CBR’s domain knowledge. Thus, more structured representations have been used. These structured representations, include the use of graphs [5, 6], decision trees [2], nested terms [3], and “definitional” structures [4] which have been used to represent knowledge, in such complex applications as design, software engineering, and medicine.

As it was explained in the chapter 2, a PU formulation is described by a set of ingredients and their quantities and once a PU has been produced a set of mechanical properties is
measured. These PU formulation parameters and their values can be used to represent a case. For the purposes of this project, PU formulations do not include any additional information that requires a complex representation format (e.g. PU microstructure); therefore the attribute-valued representation has been used for the CBR system implemented in this thesis.

3.4.3 How to Retrieve Similar Cases?

There has been much research on how to compare the similarity between two cases and much debate about what is meant by similarity. During the recent years most research done on CBR treats retrieval as a classification task (e.g. [8, 122]) where the retrieval of cases is performed by labelling using a classifier. However, for some applications, cases cannot always be labelled with meaningful labels e.g. when the variables values do not relate to each other in a meaningful way (see for instance the example in Table 3-1 p.59). In considering whether one similarity approach or another is best suited for the CB reasoner, it is necessary to consider first what is meant by similarity assessment and second, how the chosen representation format influences similarity computation.

In a CBR system, case retrieval is the process of obtaining from the case base library those stored cases that are most similar to a query case [28]. This process is illustrated in Figure 3-8. It can be observed from this figure that several subtasks have to be fulfilled for retrieval (refer also to Figure 3-4 p. 46). First, an efficient search in the case library has to be performed to find the appropriate case(s). Second, the problem of matching or similarity-assessment is the recognition that a case(s) is applicable to a new situation. Third, ranking heuristics are used to choose those cases partly matching a query that can best address the reasoner's purpose.
Similarity assessment is performed by using a similarity measure. Similarity measures were developed in the area of cluster analysis where clustering is usually viewed as a process of partitioning data into groups of "similar" objects [123]. In considering retrieval of PU cases, it is important, at the outset, to recognise that similarity assessment depends greatly on three features: (i) the point of view (or context) the user chooses to compare two cases, (ii) the retrieval algorithm used, which in turn depends upon (iii) the representation format of the cases (Figure 3-9).
In a given context, the assessment of similarity depends heavily on the format used to represent cases. When using the attribute-value representation, each attribute is numerically compared using a similarity function $sim$, which is specific for a particular attribute-type. Once all the case's attributes have been compared with the queries' attributes, these calculations are added up so that the resulting number represents the degree of similarity between the case and the query.

3.4.3.1 The retrieval algorithm
Retrieval algorithms have been developed in various disciplines (e.g. statistical analysis and information retrieval [124]) for a variety of applications (e.g. archaeology, sociology, taxonomy, and case-based reasoning). For this reason, all theoretical and practical research that has been done to date in retrieval is scattered all over the literature. Attempts to formalise mathematically retrieval models have been recently made in order to give a comprehensive and consistent framework of the different methods (see for instance [125]). Other theoretical frameworks for the systematic construction of similarity measures have also been described [126, 127].

Attribute's values of cases represented using the attribute-value representation can be seen as points in some coordinate space i.e. vectors; the closeness between two points can be measured by means of the nearest neighbour classification algorithm that uses labelled cases. When working with labelled cases, CBR retrieval can be seen as a classification task, and then nearest neighbour classifiers are used to map an unlabelled instance to a label by finding the nearest neighbour in a set of internally stored labelled instances and return its label [128]. That is, new cases will be labelled (classified) with the label (class) of the most similar case retrieved from the case base. Some of the terms in italics, are briefly described in the following paragraph.

In the machine learning (ML) domain, an instance is sometimes called an example. It is a fixed list of feature values. A feature, sometimes called an attribute, describes some characteristics of an instance. The label is a special nominal feature of an instance, which describes the phenomenon of interest i.e. the phenomenon that the user would like to learn and make predictions about. An unlabelled instance is the part of the instance without the label i.e. the list of feature values. A dataset is a set of labelled instances. Table 3-1 shows a dataset with seven instances from a heart-disease domain. The last column, sick, is what is to be predicted given the other features. In some applications, however, cases cannot be properly "labelled" into categories so this approach cannot be used, although the same terms still apply (Table 3-1).
Table 3-1. A dataset with seven instances from a heart-disease domain. Source: [128]. The first line describes the feature names, the second line the feature types (continuous or nominal), and the other seven lines the instances, each described by a list of five feature values and a label value.

<table>
<thead>
<tr>
<th>Age (cont)</th>
<th>Sex {M, F}</th>
<th>Cholesterol (cont) {norm, abn, hyp}</th>
<th>Resting RCG (cont)</th>
<th>max heart rate (cont)</th>
<th>sick {yes, no}</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>M</td>
<td>203 hyp</td>
<td>155</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>M</td>
<td>185 hyp</td>
<td>155</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>M</td>
<td>199 norm</td>
<td>178</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>F</td>
<td>243 norm</td>
<td>144</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>F</td>
<td>294 norm</td>
<td>162</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>M</td>
<td>177 hyp</td>
<td>120</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>F</td>
<td>197 abn</td>
<td>116</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>M</td>
<td>267 norm</td>
<td>99</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>M</td>
<td>274 norm</td>
<td>88</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>M</td>
<td>125 hyp</td>
<td>178</td>
<td>?</td>
<td></td>
</tr>
</tbody>
</table>

Query

Retrieval as a classification task, i.e. working with labelled cases

Table 3-2. A case-problem set with seven cases from the PU-formulation domain. Each case-problem has a case-solution associated to it. The case-problem is used for matching, and the case-solution that best matches the query is retrieved. From column two on, the first line describes the feature names, the second line the feature types (all continuous), and the other seven lines the instances, each described by a list of five feature values.

<table>
<thead>
<tr>
<th>Case-Problem</th>
<th>Density</th>
<th>Tension</th>
<th>Elongation</th>
<th>Tear</th>
<th>Hardness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>245.71</td>
<td>3.08</td>
<td>65.30</td>
<td>0.63</td>
<td>24.83</td>
</tr>
<tr>
<td>2</td>
<td>244.14</td>
<td>3.14</td>
<td>64.27</td>
<td>0.58</td>
<td>24.50</td>
</tr>
<tr>
<td>3</td>
<td>242.02</td>
<td>3.14</td>
<td>61.73</td>
<td>0.61</td>
<td>23.00</td>
</tr>
<tr>
<td>4</td>
<td>270.79</td>
<td>0.59</td>
<td>32.53</td>
<td>0.63</td>
<td>38.83</td>
</tr>
<tr>
<td>5</td>
<td>259.85</td>
<td>0.63</td>
<td>41.00</td>
<td>0.58</td>
<td>39.83</td>
</tr>
<tr>
<td>6</td>
<td>258.36</td>
<td>0.59</td>
<td>38.97</td>
<td>0.61</td>
<td>40.00</td>
</tr>
<tr>
<td>7</td>
<td>255.32</td>
<td>0.56</td>
<td>36.53</td>
<td>0.53</td>
<td>40.67</td>
</tr>
<tr>
<td>Case-Solution?</td>
<td>240.00</td>
<td>3.0</td>
<td>65.0</td>
<td>0.5</td>
<td>30.0</td>
</tr>
</tbody>
</table>

In CBR sometimes cases cannot be labelled
The nearest neighbour problem can be expressed as follows:

Given a set of \( N \) points in \( k \)-space and a distinguished point \( q \), find the \( m \) (\( m < N \)) points closest to \( q \).

In CBR we can apply the model to the situation where \( N \) is the number of cases, each case corresponds to a point in \( k \)-space, \( q \) corresponds to a query and \( m \) corresponds to a user specification of how many retrieved cases are required. Closeness is measured by some distance measure and \( k \)-space corresponds to \( k \)-attributes or features. In the simplest form, \( m \) is specified to 1.

Using the nearest neighbour rule, the local neighbourhood is defined as the training example that lies in the 'immediate neighbourhood' of the query, \( q \):

In particular, the case vector,

\[
e^N \in \{e^0, \ldots, e^N\}
\]

(3-3)

is said to be the nearest neighbour of \( q \) if

\[
\min_i d(e_i, q) = d(e^N, q)
\]

(3-4)

where the \( d(e_i, q) \) is the distance between the case \( e_i \) and the query \( q \) calculated for each case \( i \) in the case-base.

The degree of matching of closeness is therefore computed by using a distance function \( d \). A small distance is equivalent to a large similarity (Figure 3-10).

In product formulation problems, most attributes types are quantitative in nature i.e. are a metric (e.g. quantity of ingredient, temperature of ingredient, ratio of chemicals). For numerical attributes, several distance functions can be used including Euclidean, Canberra, Chebychev, City block distances. However, it is not clear why to choose from one or another distance function. Several distance functions have been defined to handle binary, nominal, and ordinal attributes. For several applications in which the variables describing a set of objects are a combination of different types i.e. mixed variables, heterogeneous distance measures have also been proposed [8].
A binary predicate \( \text{sim}(x, y) \subseteq U^2 \)
"x and y are similar"

A binary predicate \( \text{dissim}(x, y) \subseteq U^2 \)
"x and y are not similar"

A ternary relation \( S(x, y, z) \subseteq U^3 \)
"y is as least as similar to x than z is to x"

A quaternary relation \( R(x, y, u, v) \subseteq U^4 \)
"y is as least as similar to x than v is to u"

A function \( \text{sim}(x, y): U^2 \rightarrow [0,1] \)
measures the degree of similarity between x and y

A function \( d(x, y): U^2 \rightarrow \mathbb{R} \)
measures the distance between x and y

Figure 3-10. Different ways to represent similarity. Source: [129].

There are different ways to measure similarity between attributes. For example, when a distance function \( d \) is used, a real number is obtained. This number is meant to represent the closeness between the two attributes. The smaller the number the closest the attributes are.

Only a fraction of the distance measures available in the literature are presented in Table 3-3, major types relevant to CBR are described in the following section. For the purposes of this thesis, only distance functions that handle numerical attributes are described. This is because the features that normally describe the PU formulations available for this study are numerical. Some comparison studies exist among these similarity measures [8, 9, 122]. Results obtained in these studies suggest that the performance of a similarity measure depends upon the type of attribute constituting the case and on the relative importance of each attribute. The latter aspect is considered in section 3.4.3.3.2 (feature weighting p. 65).

3.4.3.2 Selected measures of similarity

For many real-world applications, variables describing a set of objects are of different types. For instance a PU foam formulation can be described in terms of compression set (quantitative variable), amount of wear (ordinal), type of polyol (nominal) and presence or absence of voids (binary variable). Heterogeneous distance functions are used to handle mixed variables. Reviews of some other distance functions that handle qualitative attributes and heterogeneous attributes include:

- In [123] some distance functions for qualitative binary attributes are given;
- In [130, 131] distances that can handle variables having more than two states or categories i.e. multi-state (e.g. linear if the states are ordered (they can be discrete or
continuous) or nominal if they are not) are presented. A general measure, which can be used to compare objects described by mixed variables is described in [10];

- Several heterogeneous distance functions are reviewed in [8]. The authors presented three distance functions namely, the Heterogeneous Value Difference Metric (HVDM), the Interpolated Value Difference Metric (IVDM), and the Windowed Value Difference Metric (WVDM). The HVDM distance function uses, the Euclidean distance on linear attributes, and the Value Difference Metric (VDM), which was introduced by Stanfill & Waltz [132], on nominal attributes. The IVDM and the WVDM handle quantitative variables in a similar fashion to VDM.

### 3.4.3.2.1 Quantitative variables

For quantitative attributes, a range of distance functions has been used in CBR systems. They include Euclidean, Manhattan, Minkowski, Mahalanobis, Canberra and Chebychev (refer to Table 3-3).

A family of dissimilarity measures indexed by a parameter \( r \), is

**Minkowski metrics**

\[
d_{ij} = \left( \sum_{k=1}^{p} w_k \left( x_{ik} - x_{jk} \right)^r \right)^{1/r} \quad r \geq 1
\]

where \( x_{ik} \) denotes the value that the \( k \)th quantitative variable takes for the \( i \)th object \((i=1, \ldots, n; \ k=1, \ldots, p)\) and \( \left\{ w_k \ (k=1, \ldots, p) \right\} \) are non-negative weights associated with the variables.

The Euclidean and Manhattan (or City block) metrics are particular cases of the Minkowskian metric with \( r=2 \) and \( 1 \), respectively:

**Euclidean distance**

\[
d_{ij} = \left( \sum_{k=1}^{p} w_k^2 \left( x_{ik} - x_{jk} \right)^2 \right)^{1/2}
\]

**City block metric**

\[
d_{ij} = \sum_{k=1}^{p} w_k \left( x_{ik} - x_{jk} \right)
\]
Table 3-3. Selected quantitative distance measures. Adapted from [123], [133]

$x_k$ denotes the value that the $k^{th}$ quantitative variable takes for the $i^{th}$ object ($i=1,...,n; k=1,...,p$) and $w_k (k=1,...,p)$ are non-negative weights associated with the variables. $r_k$ denotes the range of values for the variable $k$.

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euclidean</td>
<td>$d_y = \left( \sum_{k=1}^{p} w_k^2 \left( \frac{x_{ik} - x_{jk}}{r_k} \right)^2 \right)^{1/2}$ (3-8)</td>
</tr>
<tr>
<td>City</td>
<td>$d_y = \sqrt{ \sum_{k=1}^{p} \left( x_{ik} - x_{jk} \right)^2 }$ (3-9)</td>
</tr>
<tr>
<td>Canberra</td>
<td>$d_y = \sum_{k=1}^{p} \left</td>
</tr>
<tr>
<td>Chebychev</td>
<td>$d_y = \max_{k} \left</td>
</tr>
<tr>
<td>Taxonomic</td>
<td>$d_y = \left( \sum_{k=1}^{p} w_k^2 \left( \frac{x_{ik} - x_{jk}}{r_k} \right)^2 \right)^{1/2}$ (3-12)</td>
</tr>
<tr>
<td>Divergence</td>
<td>$d_y = \left( \sum_{k=1}^{p} w_k^2 \left( \frac{x_{ik} - x_{jk}}{x_{ik} + x_{jk}} \right)^2 \right)^{1/2}$ (3-13)</td>
</tr>
<tr>
<td>D7</td>
<td>$d_y = \sum_{k=1}^{p} \left</td>
</tr>
<tr>
<td>D8</td>
<td>$d_y = \sum_{k=1}^{p} \left( x_{ik} - x_{jk} \right) \left/ \left( x_{ik} + x_{jk} \right) \right)$ (3-15)</td>
</tr>
<tr>
<td>D9</td>
<td>$d_y = \sum_{k=1}^{p} w_k \left</td>
</tr>
<tr>
<td>D10</td>
<td>$d_y = \sum_{k=1}^{p} w_k \left( 1 - \frac{\min \left( x_{ik}, x_{jk} \right)}{\max \left( x_{ik}, x_{jk} \right)} \right)$ (3-17)</td>
</tr>
</tbody>
</table>

3.4.3.3 General Considerations for Retrieval

Different data representation schemes impose different constraints on the retrieval algorithms, although some recommendations concerning their use are universal. They relate to the preparation of input data e.g. normalisation, missing values, and feature weighting importance in the calculation of similarity. They are briefly described below:
3.4.3.3.1 Data pre-treatment

Normalisation is usually performed on the data to avoid the effect that large-valued attributes have on the similarity measure when compared to low-valued attributes (Figure 3-11). For quantitative variables, the contribution to the distance by each variable is normalised by dividing the distance for each variable by the range of the variable, so that the distance for each input variable is in the range [0,1]. This is referred to as min-max normalisation. The formula to normalise variables in any given interval $[y_{\text{min}}, y_{\text{max}}]$ is:

$$y = \frac{\left( y_{\text{max}} - y_{\text{min}} \right)}{\left( y_{\text{max}} - x_{\text{min}} \right)} (x - x_{\text{min}}) + y_{\text{min}}$$

(3-18)

where, \( x \) is the value of the variable to be normalised, \( y \) is the normalised value of \( x \), \([x_{\text{min}}, x_{\text{max}}]\) is the range of the variable \( x \) and \([y_{\text{min}}, y_{\text{max}}]\) is the desired interval for normalisation.

This type of normalisation annuls the effect of the magnitude of the data attributes, due to different scales of measurement, on the distance function. For example, if a PU foam has just two attributes, A: Density (kg/m$^3$) and B: Tear strength (N/m), and A can have values from 200 to 300, and B has values only from 0.3 to 1.0, A’s influence on the distance function will usually be overpowered by B’s influence (Figure 3-11).

![Figure 3-11. Effect of variable’s magnitude on the Euclidean distance.](image)

One problem with distance functions is that they are greatly influenced by large-valued variables. For this reason, data are normalised. For instance if both variables are scaled within the range [0,1], the two distances become 0.3 and 0.8, which are more equitable and both variables will contribute significantly to the distance.

If the measured variables have an approximately normal distribution, are uni-modal, nearly symmetrically distributed about its mean, the data set can be standardised to a Z-score by
subtracting its mean and dividing by its standard deviation. This removes the effects of outliers and measurement scale. Missing input values are normally substituted in Euclidean Manhattan, Clark, and Canberra distance measures, by the average value obtained by the instances with valid values [122].

3.4.3.3.2 Feature Weighting

In some circumstances, knowledge about the relevance of the attributes that describe the PU cases in a given context is required to obtain the best possible retrieval. This is because, when calculating a distance assuming that all features are equally important (i.e. $\forall_k \{w_k=1\}$), irrelevant and redundant features influence the distance computation. Mainly, in the machine learning field, several nearest neighbour variants have been proposed to reduce this sensitivity by including feature weights in the distance function (see Table 3-3 p.63).

Wettschereck and Aha [134] introduced a five-dimensional framework for feature weighting. For the generality dimension, they distinguished between weighting methods that learn settings for a single set of global weights, and other methods in which weights differ among local regions. Global weights refer to weights employed over the entire instance space. Local weights refer to weights that change from instance to instance (Figure 3-12).

![Figure 3-12. Global and local weighting. Adapted from [135].](image)

3.4.3.4 Other Retrieval Approaches

Other approaches to retrieval, apart from vector model-based approaches, that are not covered in this work and are used when extensive case-bases are available include the indexing or representational approach [9], and a combination thereof, referred to as the hybrid approach [8]. In the indexing approach, cases are organised by using indexing structures that speed up
the retrieval process. Retrieval approaches in this category include decision trees and supervised and unsupervised neural learning. Hybrid approaches include the use of both indexing structures and distance measures. In this approach, the indexing structure acts as a filter, reducing the search space while the similarity measures may be generated to indicate the relevance of cases. Other research work of practical use in CBR retrieval include work done in Bayesian distance measures, [4], probabilistic similarity measures [136], fuzzy logic [137, 138], neural networks [139] and genetic algorithms [140, 141].

3.4.4 Summary

CBR is a methodology that can be used to solve problems arising in product formulation because its problem solving model is based on how human experts recall, guide and adapt former solutions to solve new problems. In CBR the expert's experiential knowledge is recorded in the form of cases consisting of two parts, namely, the case-problem and the case-solution. The vector representation is the most widely used to represent cases that are simple and consist only of qualitative or quantitative valued features. Similarity assessment between the new problem and former cases is done by means of a similarity measure. The nearest neighbour algorithm is used in CBR to retrieve cases and depending on the attribute type, a weighted distance function is used. However, it is not clear what distance function should be used and why. Most researchers use the Euclidean distance function when only numerical attributes are used because it can handle attributes with any value between the interval [0,1] while for instance other distance measures e.g. Canberra cannot be used for zero-valued features because division by zero occurs (refer to eq. (3-10)). The Euclidean distance calculates the root of the squared difference between the case and query feature vectors and the result is meant to give a measure of "distance" or dissimilarity between the feature vectors. Other complex representation formats along with their retrieval approaches exist that are used when an explicit relationship between the problem and solution features is known (e.g. mathematical models) or when knowledge in the form of graphs, or highly structured knowledge is available. The general considerations when using distance functions are related to first, the normalisation of the data to avoid the effect that large-valued attributes have on the similarity measure when compared to low-valued attributes (recall Figure 3-11), and second, the use of feature weights to reflect feature relevance.

An approach to decide what is the most suitable distance would be to rely in the domain expert’s opinion. An expert could select similar cases to determine which parameters are important so that distances can be clustered with the expert’s opinion. However, in a high dimensional space in which a case is composed of many features, deciding what cases are similar results in a difficult task. Another approach is to take readings from cases retrieved by
different distances and transform them into probabilities. This procedure allows the selection of a measure that will select the case that is most likely to be linked. However, this procedure requires a lot of computation and it is necessary to assess if the likelihood of a sample belonging to a given cluster can generate good results.

### 3.5 Case-Based Reasoners for Product Formulation

The most influential CBR systems developed for product formulation applications are reviewed in this section. CBR systems can be divided into two categories: those that are academic implemented primarily to demonstrate certain aspects of CBR, and those systems that are being used commercially. Academic systems are described in more detail than the commercial demonstrators, highlighting the benefits of CBR to the formulation domain.

In the last two decades there has been active research in the development of CBR systems for process manufacturing [5, 6, 108, 142-150]. In particular, there is a growing body of research which suggests that product formulation is a potential area for the development of KBS [151-153]. In considering what systems have been already developed in product formulation and its relevance to PUs, the first aspect to consider is what is meant by the term “product formulation”.

In process-oriented industries, the term “product formulation” denotes the specification of a list of ingredients (i.e. the formulation) and processing steps and their conditions in a cost-effective manner so that products can meet specific requirements. Formulated products include those produced according to a formula e.g. rubbers, alloys, adhesives, pharmaceuticals. Typically, in formulated products the ingredients interact chemically in a complex way, which is not always known, and consequently product properties are governed by the quantity and type of ingredients. Furthermore, the processing conditions both “internal” (e.g. temperature, pressure) and “external” (e.g. environmental condition such as humidity) to the process also affect product properties. As detailed in section 3.3.3, formulated products are referred to as “weak theory domains” because usually the degree of certainty of the involved relationships between ingredients, processing conditions and product properties is not very strong. Formulated product applications are also characterised by having a large dynamic (i.e. changes over time) body of knowledge.

To the authors best knowledge a case-based reasoning system to enable the formulation of PUs does not currently exist. As has been recognised by experts in the area ([47, 154]) and by the author’s personal experience, there is a current need to adapt and develop methodologies that enable the reuse of PU formulation experiences. PU manufactures also recognise that
more research is required to enable the adoption of the advantages that have been proved successful in other fields. This study is an attempt to fill that gap.

Several attempts that have been made in other areas of product development can be studied and adapted to the case of PUs. Explored topics by academia and industry include CBR systems for dish planning [112], aluminium alloys [155, 156], steel products [157], pharmaceuticals [118, 120], steel product quality design [153], rubber [152] and plastics manufacturing [158]. In general, the drivers for developing these KBS were concerned mainly with:

- The practice of the discipline was chiefly regarded more as an art rather than as a science and therefore the domain was poorly understood;
- There does not exist comprehensive mathematical models to support product design and manufacture;
- Experience of the manufacturing personnel plays a crucial role for successful processes and troubleshooting, however, there is a need to formalise such experiential knowledge for satisfactory future problem solving;
- There is a need to collect, structure, and represent manufacturing knowledge from experts using a computational system that enables the sharing and deployment of solutions in a consistent and referable way when appropriate for future problem solving.

These needs have some resemblance to the motivations presented in chapter 2 for using a computational methodology to assist the PU formulation task. The CBR systems developed in the areas of product formulation mentioned above are analysed in the next section. The review of each CBR system starts with an introductory description of the domain in which the system was implemented, how the authors defined and represented a "case", what algorithm for retrieval was used and if any strategy for adaptation was used. This follows a comment on the performance of the system if available. Other CBR systems developed in the areas of process manufacturing are also reviewed in the section.

3.5.1 Dish Planning

Hammond [112] developed the CHEF program in 1989, which can be considered to be the first CBR system for product formulation. CHEF creates plans based on its experience in the domain of cooking under a series of different constraints such as tastes, textures and type of dish. A plan in CHEF is a recipe for preparing a complex dish that can be transformed by adding ingredients, adding steps, changing cooking times and so forth (a case in CHEF is represented in Figure 3-6 p.52).
The problem description in CHEF for creating BEEF and BROCCOLI is a set of constraints for preparing a dish:

- Include beef in the dish
- Include a crisp vegetable
- Use stir-fry method

The steps of CHEF when creating the BEEF and BROCCOLI recipe are shown in Figure 3-13.

RETRIEVAL
Searching for a plan that satisfies the constraints

Found recipe -> REC2 BEEF WITH GREEN BEANS

Recipe exactly satisfies goals ->
  Use the stir-fry method
  Includes beef

Recipe partially matches ->
  Include broccoli in the dish
  in that the recipe satisfies:
    Include vegetables in the dish

MODIFIER
Modifying recipe: BEEF and BROCCOLI
to satisfy: Include broccoli in the dish
Placing some broccoli in recipe BEEF-AND-BROCCOLI

-- Considering ingredient-critic:
  Before doing step: Stir-fry the -Variable-
  Do: chop the broccoli into pieces the size of chunks
-- Ingredient-critic applied

Projected results:
The beef is now tender
The dish now tastes salty
The dish taste savoury
The dish now tastes sweet
The broccoli is now crisp
The dish now tastes like garlic

SIMULATOR
Executing recipe.

Figure 3-13. Creating "beef and broccoli" using CHEF. Source: [112]

The first step in creating a plan is to retrieve a recipe that meets the constraints. Using the constraints of the dish as features, a recipe BEEF-WITH-GREEN-BEANS is found in the library of recipes. This recipe matches exactly two constraints i.e. it is a stir-fry dish and includes beef, and matches partially a third constraint by way of generalisation (i.e. both green beans and broccoli are vegetables). In the next step, the retrieved case is reused in the new context, i.e. a new recipe is created using the beef and green beans recipe with the substitution of broccoli for the green beans. This is done by using structures called object...
critics. Finally, a test of the proposed recipes with a simulator program designed to provide useful feedback is carried out. Before the simulation, a set of expectations for the recipe is produced. The simulator basically consists of a set of inference rules that are employed to determine the outcome of each step of the plan.

When the simulator completes, one of these expectations is not satisfied i.e. "the broccoli is now crisp". It is instead the case that: the broccoli is now soggy. A causal description of the failure is given serving two purposes: first to identify the parts of the plan that have to be altered and second the places where the planner’s knowledge is faulty. This description will form the explanation of why the failure has occurred. The modifications are controlled by domain rules (in this case the STIR-FRY dish domain), and by the goal that the original plan is being modified to satisfy (i.e. to include broccoli). The domain rules assist in determining that the problem (i.e. sogginess of broccoli) was caused by the broccoli absorbing the liquid of the meat as they cooked. To repair plans, there are many strategies. One strategy splits apart steps of the recipe. By doing this in the current recipe, a recipe with broccoli cooked separately from beef is created. After this sub-process is completed, a step is added to recombine the recipes and achieve the desired result. A second pass of the simulator verifies if the adapted recipe has satisfied all the expectations.

Finally, the successful recipe and the additional knowledge describing the new rule on how to solve this type of interaction problem are retained. Both can be used to solve similar problems in the future.

3.5.2 Drug Tablet Formulation

Craw and co-workers have been researching since the mid 1990s into the development of both CBR systems [118, 151, 159] and RBS [120, 160] for drug tablet formulation. Their work in CBR has resulted in two Ph.D. theses that have researched into three main areas, namely, the feasibility of CBR for formulating drug tablets [118], learning and the use of case-base data for adaptation [140, 141, 159, 161] and more recently the use of visualisation tools to explain CBR solutions [162].

The design of drug tablets involves the identification of compatible excipients (which are substances that balance the properties of a drug) and their quantities whilst satisfying other constraints (e.g. tablet weight). Excipients do not react chemically throughout the manufacturing process and there are no more than twenty types in a given drug formulation in most cases. This makes the formulation of drugs a less complex problem in comparison to the PUs, in which ingredients react chemically, and can also lead to reactive by-products.

Craw et al. [118] developed CBRlFS, a hand crafted CBR prototype system for tablet formulations, in collaboration with Zeneca Pharmaceuticals, UK. The system was
implemented using the ReCall software from ISoft [163]. In CBR/FS, the case formulations are represented by a problem part (i.e. the case-problem) and a solution part (i.e. the case-solution). The case-problem consists of the set of physical features describing the drug (e.g. solubility, mechanical properties), the set of stability values of the drug for each of the available excipients and the dose of the tablet. The case-solution consists of a set of excipient ingredients and their quantities. The CBR retrieval stage in ReCall uses a nearest neighbour algorithm with a voting mechanism to select the most popular solution. It also uses a decision tree (e.g. ID3) to index the partitioned groups of related formulations generated by the induction algorithm (see Figure 3-14).

![Figure 3-14. Facilities in ReCall used in CBR/FS. Source: [118]](image)

The most significant ideas Craw and co-workers’ research has highlighted are:

- **It is possible to use CBR to support complex formulation tasks.**

The development of a RBS [120] and more recently a CBR prototype system for tablet formulations [118] are examples that weak theory domains can be efficiently supported by computational tools. Craw et al.’s work proves that CBR works well in complex formulation domains that can be simplified to attribute-value cases without much compromise (i.e. information lost). However, future research in this domain is required to deal with complex cases and complex case representations. Complex cases in the sense that features are interrelated. Whilst the tablet formulation example described in [118] is represented using a similar layout to the PU formulation problem, it should be noted the interaction between the ingredients throughout PU formulation is more complex. This is due to interrelationships
between components and processing conditions and the chemical interaction between components. However, the kinds of prototypical CBR systems developed by Craw et al. [118] are useful to provide good scientific results in a controlled way allowing for a gradual introduction of complexity that real problems have.

- Further acquisition of adaptation knowledge is required to improve adaptation of compound's (i.e. excipient's) quantities.

In [118], Craw et al. experimented with small amounts of expert knowledge. They found that although the prediction of compounds' quantities were within the limits of the tablets manufacturing process as asserted by the expert, further acquisition of knowledge was required to improve the effectiveness of the CBR system.

In later research, Craw et al. [161] used "knowledge light" approaches to elicit adaptation knowledge from the case-base in a similar way as described by Haney and Keane [164] and Wilke et al. [165]. Specifically, they presented two adaptation methods namely, CBRA and PPROF. Details of these approaches can be found in [161]. Essentially, they found, on a case-base containing 156 formulations, that the PPROF gives better results than CBRA. This is because the latter relies on local feedback, while preference profiles represent the distribution of feature values for a given prediction capturing the notion of optimality directly. Their findings confirm that it is possible to use the case-base for adaptation though some knowledge from the expert is required to refine the task.

- The use of GAs does not improve performance significantly in comparison to when standard C4.5 and k-nearest neighbour algorithms are used to perform feature selection and feature weighting, respectively.

Jarmulak [140, 141] studied automated methods to optimise CBR indexing and feature weighting methods for CBR retrieval in the tablet design application. In this application, a number of irrelevant features may be present. Event though an induction algorithm (e.g. C4.5) itself performs feature selection, a more optimal indexing can be achieved if irrelevant features are eliminated before induction starts [166]. The k-nearest neighbour retrieval algorithm can also be optimised by choosing feature weights for the similarity measure. However, feature weighting can be best done by an expert.

They contrasted two methods: one incorporating feedback and hence learning bias (i.e. the wrapper method based on a genetic algorithm) with one that does not (i.e. the filter method). The GAs wrapper method for feature selection did not improve significantly the C4.5 classification and for feature weighting it did only improve attribute prediction (they restricted

---

*Knowledge light is a term used to refer to approaches that use knowledge that may be already in the case-base.*
attention to one attribute only) by about 7% on a dataset with 312 formulations, thereby showing the robustness of the basic C4.5 and k-nearest neighbour algorithms.

3.5.3 Rubber Compounding

Bandini and Manzoni [119, 167] have applied CBR to support the design of rubber compounds in the production of Pirelli® tyres for car racing, the P-Race Project (Figure 3-15).

In P-Race, the choice of a tyre -for a given race depends on previous race choices. A tyre’s tread is considered to be represented by a recipe determining its major properties. The recipe includes various ingredients such as rubber, active fillers, oils, and accelerants. The involved knowledge represented in P-Race includes, besides the tread recipe, all those features that affect the performance of tyres in a given race (e.g. morphological features of the racing tracks, weather and track conditions), and other general information about a race (e.g. data concerning the car team, time measurements).
A case in P-Race consists of a description of the current problem, its solution and the outcome. The problem description contains both qualitative (e.g. weather forecast, type of track surface) and quantitative information (e.g. time and location of the event). The solution is the coded recipe, while the outcome relates to the performance obtained (e.g. success or failure) when the solution was applied.

In P-Race, data about track descriptions and races are indexed using a fuzzy approach and stored in the “Case Memory” (see Figure 3-15). Fuzzy rules were used to represent features of race circuits that are expressed by experts in degrees of membership to a given class. For instance, a circuit can be considered to be “medium severe” or a road ground can be referred to as “quite smooth”. The fuzzy representation is also used to code rules about weather and track conditions (e.g. “moderately rainy”, “dry road ground”). In the retrieval stage, the similarity between a target problem and a case is assessed by means of a similarity function calculated as the weighted sum over all the attributes. The adaptation of cases is performed by the ACM adapter (see Figure 3-15) by the application of several domain rules including:

- **Performance-properties rules**: define changes required to obtain a given change in performance e.g. increase thermal stability by reducing hysteresis;
Knowledge-Based Systems

- **Ingredients-properties rules**: define which ingredients' attributes are involved to change the properties of a recipe e.g., increase hysteresis by increasing an ingredient;

- **Formulation rules**: define the reuse of a recipe. These rules include three different types i.e., substitution, increases in quantity, and decreases in quantity.

The P-race system was tested on a set of 250 cases selected from two entire championships (about 30 races) focusing in the most competitive teams (Ferrari 333 SP, Riley and Scott, Porsche GT3R) and the experts were satisfied by the cases proposed by the CBR system according to the authors [119].

3.5.4 Others CBR Systems in Manufacturing

3.5.4.1 Steel Industry

Suh et al. [153] proposed a CBR system, the IQS system, to support the design process in the steel industry. Their system consists of four components, namely, the case representation repository, a hierarchical case indexing, a similarity-based retrieval and a knowledge-based adaptation module. For the representation of cases they opted for the attribute-value representation format. The cases in the case-base are hierarchically structured in order to reduce the search time when retrieving for similar cases. This is done by considering steel products categories that are common when customers describe orders (e.g., hot coils steels, cold coils, plates). The similarity between a problem and a past case is defined by a matching factor, which in their system its calculation is schematically represented in Figure 3-16. For a particular property, the matching factor is obtained by calculating first the difference between the minimum values of the upper bound for a query's and a past case's attribute and the maximum value of the lower bound a query and a past case's attribute, and subsequently dividing by the difference between the upper bound and lower bound of the case attribute i.e.:

\[
MF(C, Q)_k = \frac{\text{Min}[UB(Q_k), UB(C_k)] - \text{Max}[LB(Q_k), LB(C_k)]}{UB(C_k) - LB(C_k)}
\]  

(3-19)

where \(UB\) and \(LB\) correspond to the lower and higher value of an attribute \(k\), respectively. \(C\) and \(Q\) stand for the case and query or problem.
Based on the matching factor, the similarity between a query and a case is evaluated by calculating a weighted sum of the matching factors over all attributes i.e.:

$$\text{SIM}(C, Q) = \frac{\sum_k w_k \text{MF}(C, Q)_k}{\sum_k w_k},$$

where $w_k$ is the attribute weight given by an expert.

In IQS, the adaptation module is composed of regression models and domain rules. Regression models available for steel design include linear models that map relationships between steel chemical elements (e.g. C, Si, Cu) to mechanical properties such as yield point and tensile strength.

Iwata and Obama [157] have also developed a quality design system for steel products, the QDES system. The system required 350 person-months for its development under collaboration of Nippon Steel Corporation (NSC). The system integrates a rule-base module with about three thousand domain rules and a CBR module. The authors have not published details of its implementation. However, they have reported on:

- The accuracy of the design of steel products is 30% better than with the conventional method. This was attributed to the optimisation of the combinations of ingredients due to the large number of alternatives the system proposes.

- The development of a neural network architecture using training cases using expert's intuition as the output for the network to enable the system to reason with correlation that cannot be represented by domain rules. They found that a back propagated three
layered network with 10 mid-nodes was suitable to be used when no accurate reasoning models were available between inputs and outputs in any of the steel design stages.

3.5.4.2 P-Truck Project

The P-Truck project is an ongoing project in collaboration with Pirelli [167] that resulted from the application of the P-race CBR system developed by Bandini and Manzoni [119, 167]. Part of the development of the P-Truck project includes the deployment of a CBR system module that aims to support the problems generated in the production line of tyres (e.g. non-uniformity of raw materials, climatic conditions, and lack of resources). In the first year of their investigation, study of the manufacturing domain and the implementation of a CBR prototype were achieved.

3.5.4.3 Cast Refractory Manufacture

Khemani et al. [142] developed InfoFrax, a CBR system to support the quality of fused cast refractory blocks that are used as lining for glass melting furnace tanks, under the collaboration of Carborundum Ltd India.

As in any other manufacturing process the product properties depend on a large number of processing parameters, which are interrelated to components used to manufacture the product (in the case of refractory block these include alumina, silica and zirconia). Furnaces play a key role in the manufacture of refractory block [142], for this reason the parameters related to their operation (e.g. temperature, pressure, charge mix composition, etc) are used in InfoFrax to describe a case. The number of these attributes is therefore significantly large, they include production attributes, quality assessment and administrative attributes which in total sum up about 150 attributes.

A CBR prototype was implemented in C++ using only thirty attributes. The main objective of the prototype systems was to mainly capture data i.e. to build the case-base and use it to assess the feasibility of CBR as a methodology to support the design of fused cast refractory blocks. For the retrieval algorithm a nearest neighbour algorithm was used. No adaptation module was implemented.

With the prototype InfoFrax, the developers [142] demonstrated that the CBR approach worked well in a complex and ill-defined domain, such as a foundry, by allowing the flexible capture of knowledge and expertise and hence its collective deployment within a structured framework for problem-solving.
3.5.4.4 Polymer Injection Moulding

The injection moulding process is a very complex process involving many areas of expertise such as rheology, fluid dynamics, heat transfer, polymer science, and friction. In the last decade, several CBR systems have been proposed to support the operating personnel when confronted to adjusting processing parameter after a defect has been identified [108, 148, 158]. Most of these CBR diagnostic systems also use causal rules for case adaptation.

Malek et al. [158] reported on a CBR system to optimise the injection moulding process of plastics by supporting operating personnel in their decision making process.

Description of processing parameters (e.g. mould parameters, injection parameters) and quality assessment (observed defects of the produced product) was used to represent a case-problem. Correspondingly, a case-solution and case-outcome were described by the action of operators (modification of some injection parameters) and the effect of applying these actions to the actual part, respectively. Using the attribute-value format a problem was represented as a vector consisting of a set of defects with their corresponding values and the set of injection parameters, i.e.

\[
\text{Problem} = P = \{(d_1, d_2, \ldots, d_m), \{p_1, p_2, \ldots, p_m\}\}
\]

(3-21)

The method of the mutual information to determine the necessary subset of parameters for the diagnosis of a given defect was adopted. Hence, parameters that are less than a given threshold are not considered for the diagnosis of a given defect. The mutual information is the reduction in uncertainty of one defect-variable’s value given the value of the other parameter [158]:

\[
W_p = \sum_{v \in V_p} \sum_{d \in D} p(X_d = d \land X_p = v) \times \log \frac{p(X_d = d \land X_p = v)}{p(X_d = d) \times p(X_p = v)}
\]

(3-22)

where \(V_p\) is the set of possible values of a parameter \(p\); \(D\) is the set of possible values of a given defect; \(p(X_d = d)\) is the probability that the value of the defect \(X_d\) of a training case \(x\) is \(d\); \(p(X_p = v)\) is the probability that the value of a parameter \(p\) is \(v\), \(\land\) indicates the joint probability.

Indexing of the case-base was constructed using a prototype-based neural network indexing system [168] that allows the decomposition of a problem into different sub problems that are indexed in specialised memories concerning a given defect. For each defect, a retrieval algorithm generates a list of similar cases to a partial problem. The similarity assessment is performed by computing a weighted Euclidean distance.
Shelesh-Nezhad and Siores [108] and Kwong et al. [148] also implemented CBR prototype systems to support operating personnel in determining operating parameters. Similarly to Malek et al. [158] using the attribute-value format, cases in these systems are described by a problem, solution and outcome part. A description of processing parameters (e.g. mould parameters, injection parameters) and observed defects of the product was used to represent the case-problem. Correspondingly, a case-solution was described by the actions taken by the operating personnel when solving a problem and the case-outcome consists of a description of the effect of applying those actions to the product.

Shelesh-Nezhad and Siores [108] fail to describe their system in detail, hence it is not clear how cases are represented, retrieved or adapted in their system nor the effectiveness of the system when used on the shop floor.

In Kwong et al's system, CBRS [148], the case-base is organised as a combination of a flat and hierarchically structures which are based on a checklist provided by the system builder. As a consequence the retrieval of cases is performed by the combination of two algorithms. One performs a breadth first search on the hierarchical structure followed by a serial search of the linear list, thereby retrieving some potential cases. A nearest neighbour algorithm is then used to assess the matching similarity between two cases using a weighted “aggregated” city block distance function (recall eq. (3-9)) which they defined as:

$$d = \frac{|f_{c,k} - f_{q,k}|}{f_{q,k}}$$

(3-23)

where $f_{c,k}$ and $f_{q,k}$ are the values for feature $k$ of the case and query, respectively.

The adjustment of parameters is done by using domain knowledge in the form of rules of thumb. The authors did not report on the accuracy of their system.

3.5.4.5 Process Control

Park et al. [149] reported on the development of a CBR system to support the production of colour display tubes (CDT) for computer monitors. The system is called KBPCS (Knowledge-based process control supporting system).

To evaluate KBPCS, the authors classified problems into simple (with one or two symptoms) and complex (with more than two symptoms), with a simple and complex problem occurrence of 85% and 15%, respectively [149]. They found, when using a sample of 50 queries, that the KBPCS was 89% accurate for the simple problems and 70% for the complex problems, which is the equivalent accuracy of a three to four years of experience process engineer.
The CDT production process has a large number of parameters. Around thirty features were used in KBPCS to describe symptoms and fifty to describe processing parameters. As it was the case in previous examples of manufacturing processes, relationships between these parameters in the form of explicit rules is difficult to obtain.

In KBPCS a case consists of a description of a problem and the adjustments done by the process control expert. Cases are indexed using expert’s criteria about the diagnostic of a problem. The similarity assessment between two cases is performed using a weighted Euclidean function. For the adaptation phase they considered different expertise knowledge viz. expert’s strategies, cognitive thresholds and sensitive ratios. The adaptation algorithm is defined as follows [149]:

**Figure 3-17. Adaptation algorithm used in the diagnostics CBR system KBPCS. Source: [149]**

3.5.4.6 Aluminium Alloys

Alloy formulation involves the selection of basic elements that are combined at certain compositions and processing parameters resulting in an alloy that displays the desired properties (e.g. mechanical properties). Comparable to the formulation of PUs, alloy design is a problem that mainly depends on the type, amount and ratio of constituents and the selection and sequencing of processing steps.

In the early 1980s, a KBS to aid the design of aluminium alloys for aerospace applications, ALADIN (Aluminium Alloy Design INventor) was developed by Carnegie Mellon researchers and the US-based aluminium company Alcoa [72, 155, 156, 169].
ALADIN is essentially a RBS. Although it is not a CBR it is the first KBS application to product formulation. The first prototype (after two years of commencement of the study) focused solely on narrow areas of alloy design using three additives, two microstructural aspects and five properties [72].

ALADIN works by taking in a description of a desired alloy, and then retrieving alloy candidates (i.e. alloy additives, processing conditions) that meet the desired constraints. The domain of ALADIN required expertise to be developed in the areas of alloy properties, chemical compositions, metallurgical microstructure, and thermo-mechanical processes. To represent this knowledge the following format representations were used:

- Declarative knowledge: to represent alloys, properties, metallurgical data structure and concepts;
- Production rules: used to represent empirical cause-and-effect relationships and to encode rules to control search mechanisms (e.g. among competing hypothesis), ranking, preference ordering.
- Algorithms: to represent physical, thermo-mechanical and statistical calculations.

ALADIN developers treated the alloy design problem as a planning problem because the final alloy design is a sequence of steps to be taken in a production plant in order to produce the alloy. Plans contain one or more spaces, which in turn are subdivided into levels. This multi-spatial reasoning architecture is composed of five spaces, namely, the property space, the structure space, the composition space, the process space and the meta-space (i.e. the planning space that directs all the design processing and control strategies). Concepts on microstructure were a powerful guide for the ALADIN search process since they constrain composition and processing decisions. Therefore, microstructure decisions served as an abstract plan that enabled narrowing down the number of alternatives in the composition processing spaces.

3.6 Summary and Conclusions

The AI discipline has provided tools that can be used to solve practical problems in PU product formulation, by means mainly of the implementation of KBS, which started being developed in the 1970s. The descriptions presented in this chapter, however, only give a partial view of the AI discipline and many important KBS have not been mentioned. Needless to say, the field of AI has become a vast one, and in one short survey it is not possible to cover the entire subject in any detail. Consequently, many significant developments including
some of those mentioned above were not discussed in this thesis. The KBS described were limited to those developed for product formulation applications.

KBS can be divided broadly into two types depending on how the KBS uses domain knowledge to reason. The first model of reasoning is one that human experts reason linking a series of rules and make inferences based on those rules to understand and solve a problem. Conversely, in the second model, human experts solve new problems by adapting solutions that were used to solve old problems. These two types of KBS are known as RBS and CBR systems respectively.

CBR is proposed as the most appropriate methodology to address the problems found in PU formulation. After analysis of the complexity of PU formulation in chapter 2, it became clear that RBS and hence its consequent rule induction task present challenges that cannot be addressed with the data available for this work. The main reasons are due to the fact that (i) knowledge in the PU industry is only held by a few experts and lacks formalisation mainly because of commercial secrecy practices and therefore rule elicitation and any other PU knowledge engineering task is likely to fail; (ii) the complex nature of the formulation task (as it was reviewed in chapter 2) has led to a lack of complete mathematical models that for example could relate compositional PU features to micro structural, processing parameters and measured mechanical properties enabling a more comprehensive formulation of PUs.

These primary difficulties (and other drawbacks expressed in this chapter) indicate that a computational technology (or methodology) can support the PU formulation task while at the same time providing a framework for PU knowledge representation and process formalisation. The CBR paradigm has the potential to address these needs. The main reason stems from the fact that CBR relates naturally to the way formulators reason when solving new problems i.e. the use of past experience to find the solution to the new problem and guide its adaptation. In addition, the growing research in the recent years in implementing CBR systems for the support of weak theory domains [118, 120, 153, 170] suggests that CBR can be easily implemented to support PU formulation providing at the same time a framework for the representation, structure, storage and use of past PU formulations (and hence valuable expertise) that is also valuable for future problem solving. Other advantages rely for example in that CBR is a methodology that capitalises on other computational technologies (e.g. neural networks, genetic algorithms, fuzzy logic) to deal with particular complexities in a problem domain.

The implementation of CBR for product formulation requires the formulating expertise to be coded into cases. A case contains a problem description and its solution. Issues on how PU cases are best represented were discussed. It was found that the attribute-value representation is the simplest and most commonly used in CBR. It also facilitates retrieval, which is
commonly done in the CBR field by using a nearest neighbour classifier that uses a distance function to assess the similarity between two cases. The use of a particular distance function depends greatly on the type of attributes. When only numerical attributes are present the Euclidean distance is the most commonly used.

Additionally, it should be noted that none of the CBR systems reviewed in this chapter have reported on novel similarity measures. Most have used standard distance functions for the retrieval algorithm. The majority of studies reviewed have focused on demonstrating the feasibility of the CBR paradigm to support problem solving tasks. The majority of the CBR systems only perform retrieval. Only a few researchers have reported on approaches to adaptation.

The primary objective of this chapter is to give answer to research question 1: What computational methodologies are available for dealing with complicated problems arising in PU formulation? What are the basic assumptions for using these techniques? What technique is the most appropriate to support the formulation of PUs?

The research discussed in this chapter has illustrated that computational technologies for problem solving fall broadly into two types depending on how they assume human expert reasoning is evoked: these are RBS and CBR. Both methodologies are currently used to build KBS in product formulation domains as they complement each other. In these applications, it is normally assumed that human experts reason by either linking a series of rules or by referring to former solving experiences so that a solution to a new problem can be proposed. In real PU formulation practice, most human experts use both models of reasoning when solving problems. Due primarily to difficult rule elicitation and the nature of the data set available in this project, CBR is the methodology most appropriate to be implemented to support the PU formulating tasks.
PART II. A CASE-BASED REASONING-NEURAL NETWORK APPROACH TO SUPPORT THE FORMULATION OF POLYURETHANES
Chapter 4

MOMENT FEATURE VECTOR MATCHING
Retrieval by Shape Matching

Abstract: The retrieval phase is a crucial step in case-based reasoning systems. To date, most retrieval algorithms in CBR use a modified version of the nearest neighbour rule that uses a distance function as the similarity measure, which in turn depends upon the attribute type. However, the distance measure computes similarity on a local basis, i.e. feature by feature, and their weighted sum is meant to give the measure of similarity between two cases. This chapter reports on an alternative similarity measure based on the moments' description of a case when this can be represented as a two dimensional image. A comparison between the standard measures of similarity and the image's moments based approach is discussed for the retrieval of PU foam formulations. The aim is to determine if there are any advantages of the alternative approach compared to the classical similarity measures when only numerical attributes are used to describe a case.

Key words: Similarity measure, retrieval, distance metric, shape recognition, theory of moments

4. CHAPTER 4. MOMENT FEATURE VECTOR MATCHING

4.1 Introduction

In problem-solving case-based reasoning (CBR) systems, the retrieval phase is a crucial step. CBR is based on the premise that similar problems should have similar solutions [28]. Hence, the effectiveness of a CBR system depends heavily on the system’s ability to retrieve former experiences (i.e. cases) from the case-base that are similar and hence useful and applicable to solve new problems.

A review of the retrieval approaches that have been used and developed in the CBR field for cases represented as feature vectors has been presented in chapter 3. Much research has been focussed on the retrieval of “labelled” cases where retrieval is seen as a classification problem. The most common classifier used in the context of CBR retrieval is the nearest neighbour algorithm. This approach has been applied to a wide range of problems using not only numeric features or attributes (for which nearest neighbour methods are traditionally used) but also using symbolic unordered ones [171]. The approach has also been studied in different contexts (e.g. in geography [172], marketing, psychology, linguistics, archaeology,
and other areas that aim to identify homogeneous subgroups from an heterogeneous sample) under different names (e.g. cluster analysis [123], information retrieval [124], similarity matching [173], classification [130], instance-based learning methods [174]), so the literature is extensive.

Using the nearest neighbour rule with the Euclidean distance, the similarity of a case \( c \) and a query \( q \) is defined as the weighted sum of the distances of their attributes (refer to Figure 3-9 p.57). It can be argued that this similarity calculation is local in nature given that the similarity between cases is assessed by comparison of their constituent local features and the sum is meant to give a measure of the similarity between the two cases. However, the similarity between two cases could be determined in a number of ways. For example, matching could be based on sample representations that depend on the global shape (e.g. shape-based object recognition [29]). It is well known that human perception of shape is based in visual parts to the extent that a single outstanding visual region is enough to recognise the whole object [29]. Consequently, a cognitively motivated shape similarity measure for the retrieval of PU cases could be based on representation of visual parts.

Shape matching as a global similarity measure between cases within the CBR framework is explored in this thesis. A similarity measure based on moment descriptors of shape representations of cases described as vectors is proposed as an alternative retrieval approach that could be beneficial for the PU domain and similar domains that use CBR for problem solving. The usefulness of the global retrieval approach based on shape matching over other local retrieval approaches will be assessed in terms of:

a) the extent to which the retrieved case is a useful solution to a query by the provision of a quantitative indication of this measure;

b) allowing cases differing in the number of attributes to be assessed for similarity (i.e. feature weighting);

c) allowing the similarity assessment of cases composed of different attribute’s type (e.g. quantitative, qualitative) to be similarity assessed;

d) the level of understanding appreciated by the user.

4.2 Problem Statement

Given a user specified query \( q \), and a stored case \( c \), considered as feature vectors \((a_1, \ldots, a_n)\) where \( a_1 \) specifies the value for the \( i \)-th attribute \( A_i \), find an alternative similarity measure that reflects the global characteristics of both the query and the case.
4.3 Objectives and Research Questions

The objective of this chapter is to identify a global similarity measure that can be used for the retrieval of PU formulations. This objective can be satisfied if the following research question and sub-questions are answered (see Figure 4-1).

4.3.1 Research Question 2

Is a global similarity measure more effective than local similarity measures for the formulation of PU foams?

This research question can be best answered if divided into two sub-questions:

Sub-question A. What is the performance of current local similarity measures based on standard metric distances? What are their advantages and disadvantages?

Sub-question B. What global similarity measures exist that can be implemented for the retrieval of PU cases? What are the basic assumptions for using these approaches? How can these similarity measures be used to assess the similarity between PU formulations? What are their advantages and disadvantages? Do they perform better than current local measures based on standard metric distances? If so, how is retrieval enhanced?
CBR retrieval is a topic of intensive research... [Ch 2], because the "Effectiveness of a case-based reasoner depends on the ability to determine former experiences (cases) that are useful and applicable to solve new, similar problems."

Retrieval is key for the effectiveness of a CBR system

Research Question 1

Is a global retrieval approach more effective than a local approach?

Sub question A

What is the performance of the current local similarity measures for cases represented as attribute-value pairs? What are their advantages and disadvantages?

Sub question B

What global similarity measures do exist that can be applied for the retrieval of PU cases? What are their advantages and disadvantages? Do they perform better than current local pairwise measures based on standard metric distances? If so, how is retrieval enhanced?

Figure 4-1. Research question for the chapter 4.
The effectiveness of a case-based retrieval algorithm depends on the performance of the similarity measure used. If retrieval is improved, the CBR system is improved as well. The effectiveness, advantages and disadvantages of alternative global similarity measures based on the shape or global geometry of a case are studied in this chapter.

4.4 Methodology and Experimental Setup

4.4.1 Introduction
The methodology used to design the experiments for two case studies and how the data were obtained to address sub-questions A and B are presented in this section. The development of experimental data for each case study is thoroughly explained. For each case, the statistical design techniques and experimental procedure to obtain the PU foams are described.

The two study cases generated various data that were used for the creation of the case-bases. Data regarding (i) the chemical formulations to produce PU foams, and (ii) testing data showing the mechanical performance of those of PU foams were used to represent the (i) case-solution and the (ii) case-problem respectively.

After the presentation of how the experimental data set was obtained and how it was represented into a Case-Base (CB), the methodology to answer sub-questions A and B is presented.
4.4.2 Case Study 1. Water-Blown Integral Self-Skinning Flexible Polyurethane Foam Formulation

4.4.2.1 Experimental Statistical Design

The aim of this formulation study was to characterise new formulations of water-blown integral self-skinning PU foams in order to suppress the use of HCFC-141b to meet the
deadlines imposed by the Montreal protocol [50, 56] i.e. to eliminate the use of HCFC from the manufacturing of foams. Specifically, the main objective was to study the effect of the polyol’s type on the foams’ mechanical properties. To accomplish this experimental objective a full factorial statistical design [175, 176] was applied.

A brainstorm session was conducted to select factors and levels for investigation. The initial step was to ask a flexible foam expert (Mr Miguel Quintero1°) to determine what parameter settings, if any, produced a reasonable foam product. Suggestions from the expert led to keeping the polyol with higher molecular weight (i.e. polyol B) constant. To meet the objective of this project, i.e. to study the effect of the polyols on the foam blown with water, the ratio between polyols A and C, and the use of either chain extender D or E or a mixture thereof were used as the main factors for investigation. Foams were moulded using two types of polyisocyanate. All the chemical formulations were generated maintaining the average equivalent weight of polyol side constant. The resulting experiment, for the factors and levels described, is a factorial design with 18 observations (refer to Appendix I for detailed information on how the statistical design was developed). The starting chemical water-blown formulation that was used is shown in Table 4-1.

<table>
<thead>
<tr>
<th>Component</th>
<th>Functionality</th>
<th>Equivalent Weight</th>
<th>PPHP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polyol A</td>
<td>3.00</td>
<td>1600.00</td>
<td>48.31</td>
</tr>
<tr>
<td>Polyol B</td>
<td>3.00</td>
<td>2640.00</td>
<td>15.18</td>
</tr>
<tr>
<td>Polyol C</td>
<td>3.00</td>
<td>2000.00</td>
<td>26.22</td>
</tr>
<tr>
<td>Polyol D (Chain Extender)</td>
<td>2.00</td>
<td>31.00</td>
<td>9.25</td>
</tr>
<tr>
<td>Polyol E (Chain Extender)</td>
<td>2.00</td>
<td>45.00</td>
<td>1.04</td>
</tr>
<tr>
<td>Catalyst I</td>
<td>-</td>
<td>-</td>
<td>0.94</td>
</tr>
<tr>
<td>Additive I</td>
<td>-</td>
<td>-</td>
<td>12.42</td>
</tr>
<tr>
<td>Surfactant</td>
<td>-</td>
<td>-</td>
<td>1.66</td>
</tr>
<tr>
<td>Water</td>
<td>2.00</td>
<td>9.00</td>
<td>1.80</td>
</tr>
<tr>
<td>HCFC-141b</td>
<td>-</td>
<td>-</td>
<td>0.00</td>
</tr>
<tr>
<td>Total A-Side</td>
<td>-</td>
<td>-</td>
<td>116.82</td>
</tr>
<tr>
<td>Isocyanate</td>
<td>2.70</td>
<td>196.23</td>
<td>63.38</td>
</tr>
<tr>
<td>Index</td>
<td>0.51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Equivalent Weight B-Side</td>
<td>204.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iso/Polyol Ratio</td>
<td>0.543</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**4.4.2.2 Production of Test Foams in the Laboratory**

The 18 trials were performed in a completely random fashion to avoid experimental bias from unknown sources of variation [175]. Hand batching (see Figure 4-3) was advantageous for

---

10 Chemical Engineering Department, Los Andes University, Colombia
compress moulding the foams because the quantity of processed material was insufficient to justify the cost of a mixing machine. For each trial, the foam was processed according to the following method.

**Formulation of reactants.** To formulate the foams from basic components, standard chemical calculations were undertaken first. Once the processing formulation was obtained, the desired amount of reactants was weighed. First, polyols were weighed into a suitable container that was approximately 2-3 times larger than their volume. Then, the correct amount of catalysts, pigments, surfactants and other additives was carefully weighed and incorporated into the polyols to form the *polyol formulation* or masterbatch. The masterbatch was mixed thoroughly to homogenise it, avoiding when possible the introduction of air into the mix. The walls of the container were scraped during mixing to ensure a homogeneous mix.

**Moulding of the foams.** The temperature of the masterbatch mixtures and the polyisocyanates for each of the 18 trials was kept at 20°C. For each trial, correct amounts of the polyol mixture and polyisocyanate were weighed in a ventilation chamber and mixed thoroughly to mould the foams in quantities adjusted to fill the size of the aluminium mould (volume of the mould = 392.04 cm³). The mixing was carried out at 3000 rpm for 8 seconds, and then the reactive mix was poured into the mould. This mould had been previously treated with a release agent and kept at constant temperature of 40 ± 2°C. After 50 s, the mould was quickly closed to allow the mixture to polymerise. During curing, the temperature of the mould increased significantly (10 ± 2°C) due to the exothermic reaction. After 5 minutes, the cured polymer obtained sufficient strength to be demoulded. The foam was removed and hand-crushed to pop the unopened cells in the foam. This was done to prevent shrinking of the foam by cooling [154]. The foams were stored at room temperature for 24 hours to complete the cure cycle before testing. No reactivity testing was done for this set of formulations. This hand batch processing used for the preparation of the foams is schematically described in Figure 4-3.

---

11 The experiments were carried out by the author of this thesis at Los Andes University, Colombia in 2002.
Figure 4-3. Experimental procedure description to produce moulded PU foam
To formulate a foam from its basic components, the first step is to calculate the quantities of the reactants. Index and ratio of the reactants are normally constrained. After preparation of an homogeneous mixture of polyols and other reactants, this mixture is reacted with the isocyanate and hand mixed in paper cans in order to measure their reactivity profile. After this, the foams are moulded and cured.

4.4.2.3 Measurement of the Foams’ Physical and Mechanical Properties

After 24 hours of curing at room temperature, PU foams were stored in a humidity and temperature controlled environment and after seven days, three testing samples (for each dependant variable) were obtained and mechanical tests were carried out. For characterising PU foams, the variables that were recorded include density, tensile strength, % elongation, tear strength and hardness (measured after 1 second and 30 seconds of indentation). The results of the 18 trials are presented in Appendix I.
Density. The density measured, as indicated in Figure 4-5, is an apparent or bulk density, not the true polymer density. Squared samples were obtained by first removing one centimetre of outer PU skin layer and then each sample was carefully measured and weighed. The density was calculated as the sample weight divided by its volume. Density values varied considerably due to the measurement equipment and due to the difficulty of obtaining standardised samples (i.e. perfectly squared samples) hence giving in most cases inaccurate results. These values were also affected by the nature of the foam i.e. voids due to the blowing using water.

\[
\text{Figure 4-4. Self-skinning PU foam density measurement}
\]

Tensile Test. In this test, dumbbell shaped samples were pulled at a constant rate of deformation until they fractured according to the ASTM D412-98a standard [177]. The force required at the breaking point was reported as the tensile strength of the foam. The maximum extension of the sample as a percentage of its original length is the elongation at break or ultimate elongation.

Tear Strength. In this test rectangular three pieces per foam sample were slit at one end and the force required to continue this tear at constant rate was measured according to the ASTM D624-00e1 standard [178].

Hardness. The ISO 7619:1997 standard [179] was used to measure foam skin hardness on the Shore A scale. This scale has a range of readings from 30 to 95 points. Foam samples with thickness no less than 6 mm were used. The foam pieces were placed on a hard surface with the skin side facing up and the harness meter was positioned 12 mm away from the edges of the test piece. The measurement was made by using a frustoconical indenter with a spring force that decreased with increasing indentation. Two readings at 1 and 30 seconds after indentation were taken. Five measurements were taken at different positions on the test piece at least 6 mm apart to determine the mean value.
4.4.3 Case Study 2. Combustion Modified High Resilient Foam

4.4.3.1 Experimental Statistical Design

For this case study, the number of runs was restricted to a maximum of 34. This was mainly because of costs associated to produce more formulations and because normally in industrial settings this is the maximum number of trials that can be done in one working day. Therefore, the aim of this study was to obtain the maximum number of foam properties with the minimum of experiments (i.e. 34). One of the designs that could meet the experimental objective (i.e. screening out main effects only) while at the same time working under the restriction of few runs is a fractional factorial statistical design [175]. In order to select factors and levels for investigation, a discussion with the foam expert at BASF (Mr Gary Jackson)
was conducted. The levels of the variables that would yield stable foam formulations according to the expert opinion are shown in Table 4-3.

The foam formulation used for this case study provided by Mr Gary Jackson, Elastogran, BASF-UK is shown in Table 4-2. It corresponds to a typical Combustion Modified High Resilient (CMHR) foam, which finds use in various automotive foam parts.

Table 4-2. Combustion Modified High Resilient (CMHR) foam formulation

<table>
<thead>
<tr>
<th>Component</th>
<th>OH Value</th>
<th>Molecular Weight</th>
<th>Water Content [%]</th>
<th>Functionality</th>
<th>Viscosity @25 °C [mPas]</th>
<th>Formulation [ppw]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polyether Polyol 1</td>
<td>28</td>
<td>5400</td>
<td>0.05</td>
<td>2.7</td>
<td>1125</td>
<td>91.25</td>
</tr>
<tr>
<td>Polyether Polyol 2</td>
<td>42</td>
<td>4000</td>
<td>0.1</td>
<td>3</td>
<td>1000</td>
<td>2.5</td>
</tr>
<tr>
<td>Amine Crosslinker</td>
<td>1160</td>
<td>150</td>
<td>0.5</td>
<td>3</td>
<td>600</td>
<td>1</td>
</tr>
<tr>
<td>Amine Catalyst 1</td>
<td>-</td>
<td>-</td>
<td>0.5</td>
<td>-</td>
<td>0.98</td>
<td>0.7</td>
</tr>
<tr>
<td>Amine Catalyst 2</td>
<td>-</td>
<td>-</td>
<td>47</td>
<td>-</td>
<td>8.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Surfactant</td>
<td>-</td>
<td>-</td>
<td>0.2</td>
<td>-</td>
<td>110</td>
<td>0.5</td>
</tr>
<tr>
<td>Water</td>
<td>-</td>
<td>18</td>
<td>2</td>
<td></td>
<td></td>
<td>3.55</td>
</tr>
<tr>
<td>%NCO</td>
<td>33.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Index</td>
<td>85 - 105</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4-3. Foam expert's selected variables levels to yield a stable foam

<table>
<thead>
<tr>
<th>Variable</th>
<th>Variables' level [pbw]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polyether Polyol 1</td>
<td>Vary to keep formulation at 100%</td>
</tr>
<tr>
<td>Polyether Polyol 2</td>
<td>1.5 2 2.5 3</td>
</tr>
<tr>
<td>Amine Crosslinker</td>
<td>0 0.5 1 1.5 2</td>
</tr>
<tr>
<td>Amine Catalyst 1</td>
<td>0.3 0.5 0.7 0.9</td>
</tr>
<tr>
<td>Amine Catalyst 2</td>
<td>0.25 0.5 0.75</td>
</tr>
<tr>
<td>Surfactant</td>
<td>0.25 0.5 0.75</td>
</tr>
<tr>
<td>Water</td>
<td>3 3.25 3.5 3.75</td>
</tr>
<tr>
<td>Index</td>
<td>85 100</td>
</tr>
</tbody>
</table>

All the factors (in total eight) were varied between their maximum and minimum levels according to Table 4-3. Two centre points (i.e. formulation using variables at their medium optimum level) were added. No replications were possible because of the number of runs was restricted by the manufacturer. A $2^{(8-3)}$ fractional factorial design resulted in a total of 32 observations plus two added centre points. The resulting experiment and the details of the experimental design are presented in Appendix I.

4.4.3.2 Experimental Procedure

The procedure to obtain the thirty four trials was performed randomly to avoid experimental bias from unknown sources of variation [175]. It was carried out using conventional PU hand mixing methods in the same way as it is illustrated in Figure 4-3. The trials were carried out at the Elastogran formulation laboratory located at Alfreton Industrial State, Derbyshire, UK.
Reactivity profile and friability (subjective rating) were determined from hand mixing foams prepared in paper cans. Free rise densities were measured on core samples of open blow foams. The results for the 34 trials are shown in the Appendix I. It was observed during these runs that some of the foams produced from the experimental conditions were of poor quality, with some notable exceptions. Such variation was expected, of course, since the purpose of the study was to deliberately induce variation into the results to obtain a spectrum of properties in 34 experiments.

4.4.3.3 Measurement of the Foams' Physical and Mechanical Properties

For this case study, five response variables were recorded. These included the density, tensile strength, elongation percentage, compression set, and hardness. After 24 hours of curing at room temperature, the foams were translated to a humidity and temperature controlled environment. After a week, three test samples for each mechanical testing were obtained mechanical tests were performed.

Density was measured in the same way as it was performed for the foams of case study 1.

Hardness was measured according to the standard test BS ISO 2439:2000. The total force, in newtons, required to produce a specified indentation of test piece was measured. First a preliminary indentation was carried out. Here a force of 5N was applied to the selected test area and the test piece was indented at a rate of 70% of the thickness. After reaching 70% deflection, the load was released at the same rate. This procedure was done three times and then the indentation hardness was determined. Immediately after the third unloading the test piece was indented by 40% of the thickness maintaining this deflection for 30 s and the corresponding force, in newtons was annotated.

Tensile strength (the maximum tensile stress applied during stretching to rupture) and elongation at break (the percentage of elongation at rupture) were measured according to BS EN ISO 1798:2000. Standard test pieces were placed symmetrically between the grips of the testing machine. The maximum force and the distance between previously marked edges were measured immediately prior to break of the test piece. The average tensile strength of each test piece is given by

\[
\text{Tensile strength} = \frac{\text{maximum force at break}}{\text{average initial cross-sectional area}}
\]  

(4-1)

The elongation at break is calculated normally as a percentage of the original gauge length,
\[
% \text{Elongation} = \frac{\text{gauge length at break} - \text{initial gauge length}}{\text{initial gauge length}} \times 100 \quad (4-2)
\]

Compression set was measured according to BS EN ISO 1856:2001 [180]. This test measures the permanent deformation of the PU samples after they have been exposed to compressive stress for a set time period. The difference between the initial and final thickness of a test piece after 75% compression at 70°C and after a recovery time of 22 h was reported. Three samples per formulation were measured. A test sample was placed between the plates of the compression device (see Figure 4-5) and it was compressed 75% of its thickness and maintained in this condition. After 15 min the sample piece was placed in an oven at 70°C and left for 22 h. The sample was then removed from the oven and left to recover for 30 min. Its thickness was re-measured and the compression set percentage as a percentage was reported.

\[
% \text{Compression Set} = \frac{\text{original thickness} - \text{thickness after recovery}}{\text{initial thickness}} \times 100 \quad (4-3)
\]

The results of these tests for the 34 formulations are presented in the Appendix I.

4.4.4 Methodology for Sub-question A: Local-Based Similarity Retrieval

4.4.4.1 Sample: Attribute-Valued Case-Bases

To obtain the case-bases (CB), problem and solution descriptions from each experiment of the two case studies were selected. The case-problem corresponds to a set of mechanical properties, and the case-solution is given in terms of the PU ingredients and their quantities, as it is depicted in Figure 4-6.

For the CB1, the case-problem's attributes consisted of the six physical properties measured for the PUs, i.e. density, tensile strength, % elongation, tear strength and hardness (measured at 1s and at 30s). All these attributes have continuous numerical values. The case-solution's attributes correspond to 22 attributes 11 nominal values identifying the ingredients used and 11 numeric values identifying the quantity of each ingredient. However, all 11 nominal values identifying the ingredients used were constant for all the formulations. Therefore, a simplified representation containing half the attributes was used to represent the CB1. This is, the context unambiguously determined which entry corresponds to which attribute, and therefore the attribute names are omitted.

In a similar fashion for obtaining CB1, the case-base resulting from the second case study (CB2) was also represented by a list of attribute-values. The case-problem's attributes consist of the five physical properties measured for the PUs, i.e. density, tensile strength, %
elongation, hardness, and % compression. All these attributes have numerical values. The case-solution’s attributes correspond to 8 attributes, which are the chemical formulation variables i.e. polyether polyol 1, polyether polyol 2, amine crosslinker, amine catalyst 1, amine catalyst 2, silicone, water, and index. CB2 has different types of attributes to CB1 because the end use of both foams are different and while in self-skinning PU foams in important to measure hardness, for flexible foams it is more important to measure compression set.
4.4.4.2 Preparing the Input Data

Commonly, data pre-treatment is algorithm-driven [181]. This means, that in order to decide whether to normalise or transform data, it is necessary to know how the data are going to be used and what are the likely implications of a chosen type of normalisation. In this thesis, the data set (cases) was used in two main CBR algorithms, i.e. retrieval and adaptation. In retrieval algorithms based on the feature vector model [130], cases are usually normalised (or scaled) by dividing each attribute by the range (i.e., maximum-minimum) of...
that attribute, so that the distance for each attribute is in the approximate range [0,1]. This is referred to as *min-max normalisation* (refer to section 3.4.3.3.1 p. 64).

**Remark:** Particularly, in CBR, the issue of the effects of normalisation on the retrieval and adaptation of cases is normally overlooked. It is not clear in the literature what type of normalisation should be applied to data to improve retrieval, and what is the impact of the choice in retrieval or adaptation.

### 4.4.4.3 The Retrieval Algorithm

Using each case as a query for both case-bases, cases were similarity assessed using the nearest neighbour rule implemented using a set of standard quantitative distance functions (Table 4-4). The pseudo-code is given as follows:

```plaintext
for each feature k of the query
    Find the corresponding feature in the stored case
    Compare the two values to each other and compute the degree of match using different distance functions
    for k = 1 to attribute p (k = 1 ... p)
        switch Distance Function
            case 'Distance Measure Type' (i.e. Canberra, Divergence, D7).
                case 'Euclidean Square'
                    distance = distance + \( (C_k - Q_k)^2 \)
                case 'City'
                    distance = distance + abs \( C_k - Q_k \)
            ...
        end
    end
end

Retrieve most similar case to query (i.e. calculate minimum distance case-query)
for i = 1 to Distance Measure Type
    for j = 1 to all cases
        min = minimum(distance)
    end
end
```

Figure 4-7. Retrieval algorithm using standard distance functions.
Table 4-4. Selected quantitative distance functions applied to CB1 and CB2 for analysis. Adapted from [123, 133]; $x_{ik}$ denotes the value that the $k^{th}$ quantitative variable takes for the $i^{th}$ object ($i=1,...,n$; $k=1,...,p$) and $w_k$ ($k=1,...,p$) are non-negative weights associated with the variables; $r_k$ denotes the range of values for the variable $k$.

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euclidean</td>
<td>$d_{ij} = \left( \sum_{k=1}^{p} (x_{ik} - x_{jk})^2 \right)^{1/2}$</td>
</tr>
<tr>
<td>City</td>
<td>$d_{ij} = \sum_{k=1}^{p}</td>
</tr>
<tr>
<td>Chebychev</td>
<td>$d_{ij} = \max_{k=1}^{p}</td>
</tr>
<tr>
<td>Canberra</td>
<td>$d_{ij} = \sum_{k=1}^{p} \left</td>
</tr>
<tr>
<td>Divergence</td>
<td>$d_{ij} = \left( \sum_{k=1}^{p} \left( \frac{x_{ik} - x_{jk}}{r_{ik} + r_{jk}} \right)^2 \right)^{1/2}$</td>
</tr>
<tr>
<td>D7</td>
<td>$d_{ij} = \sum_{k=1}^{p} \left</td>
</tr>
<tr>
<td>D8</td>
<td>$d_{ij} = \sum_{k=1}^{p} \left</td>
</tr>
<tr>
<td>D9</td>
<td>$d_{ij} = \frac{\sum_{k=1}^{p}</td>
</tr>
<tr>
<td>D10</td>
<td>$d_{ij} = \sum_{k=1}^{p} \left( 1 - \frac{\min(x_{ik}, x_{jk})}{\max(x_{ik}, x_{jk})} \right)$</td>
</tr>
</tbody>
</table>

4.4.4.3.1 Global Feature Weighting

Distance functions were calculated using two sets of feature weights. One set in which all features are equally important (i.e. $\forall_k \{w_k=1\}$), and the second set using feature weights recommended by a PU expert. It was previously discussed in Ch 2 p. 18 and Ch 3 p. 65, that in some circumstances, knowledge about the relevance of the attributes describing the cases in a given context is necessary for retrieval. This might be for instance, the case when the user wants to give relevance to a certain feature(s) over others in retrieval e.g. give more...
importance to retrieve a case to reduce cost, with, for example, certain polyols or amount of fillers.

For the case of the PU formulations used in this project, the feature’s relevance weights were set by the PU expert in decreasing order as: density > tensile strength > compression set > elongation > tear strength > hardness. Feature weights in this case represent the magnitude of the preference of a mechanical property over another in retrieval and are listed in Figure 4-8.

<table>
<thead>
<tr>
<th>Weight rated by a PU expert</th>
<th>Density [kg/m³]</th>
<th>Tensile strength [MPa]</th>
<th>Elongation [%]</th>
<th>Tear [N/mm]</th>
<th>Hardness [Shore A]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scale between [0,1]</td>
<td>1.0</td>
<td>0.8</td>
<td>0.6</td>
<td>0.4</td>
<td>0.2</td>
</tr>
<tr>
<td>Instance (case), C</td>
<td>280</td>
<td>3.15</td>
<td>78</td>
<td>0.7</td>
<td>18</td>
</tr>
<tr>
<td>Query, Q</td>
<td>290</td>
<td>3.40</td>
<td>64</td>
<td>0.9</td>
<td>24</td>
</tr>
</tbody>
</table>

For instance, using the previous feature weights and the Euclidean distance, results in:

\[
d(Q, C) = \sqrt{(-10)^2 + (1.0)^2 + (5)^2 + (0.8)^2 + (14)^2 + (0.6)^2 + (0.2)^2 + (0.4)^2 + (6)^2 + (0.2)^2}
\]

Figure 4-8. Example of retrieval using feature weights

4.4.5 Methodology for Sub question B: Global-Based Similarity

In order to evaluate the impact of adopting global based similarity approaches, cases from both case-bases, CB1 and CB2, were represented as two dimensional images and analysed by using moment functions in a similar fashion to the approach utilised in image analysis in pattern recognition problems [182].

In image analysis, a set of moments computed from a digital image, generally represents global characteristics of the image shape and provides information about different types of geometrical features of the image [182]. Accordingly, if a case is represented as an image (see section 4.4.5.1), a set of moments, representing the global characteristics of a case, can be computed. The feature representation capability of image (i.e. case) moments could then be used for case retrieval. Retrieval, then, can be seen as a pattern recognition problem that aims to classify data or patterns (e.g. a new query) based on prior knowledge (e.g. previous cases).

The purpose of this section is to describe the methodology adopted to investigate the practical advantages and limitations of the use of moments for CBR retrieval. The following questions will be addressed in this section:

1. How an image-case should look? (section 4.4.5.1)
2. What type of moment should be used? (section 4.4.5.2)
3. How many different moments must be computed to distinguish significantly different patterns, such as different queries? (section 4.4.5.3)
4. What is the effect of low and higher moments in the computation of similarity? (section 4.4.5.3)

5. How can cases with different formulations and feature's relevance be recognised as similar? (section 4.4.5.3.1)

4.4.5.1 Representation of Cases using Moment Functions

Inspired by the system of encoding information in biological molecules, i.e. the genetic code represented in deoxyribonucleic acid (DNA), cases were represented as genes and retrieved from the case base using genes' moment features.

As discovered in 1953 by James Watson and Francis Crick, based on the work of Rosalind Franklin and Maurice Wilkins, a DNA molecule is arranged in a double-helix configuration [183]. In double stranded DNA, the two strands are wound about each other in the form of a double helix and are held together by hydrogen bonds between complementary nucleotide (purine and pyridine) bases. A segment of DNA containing the nucleotide sequence for making a particular protein is called a gene (Figure 4-9). By analogy, one strand of DNA can be used to represent the case-problem (i.e. the chemical ingredients that yield a particular polymer). DNA strands are complementary; this means that each DNA region in one strand is linked to a specific region in the complementary DNA strand. This is, each case problem has its complementary case-solution, and both make up a gene (or case), which contains the information for making a particular PU.
A gene represents a case, which is a specific region in the DNA that contains the information for making a particular PU. In double stranded biological DNA, the two strands are wound about each other in the form of a double helix and are held together by hydrogen bonds, between complementary nucleotide bases. The hydrogen bonds can be seen as the unknown component-property relationships that held together the two DNA strands. The complete DNA molecule makes up the case-base, which contain the genes (i.e. cases).

Each PU case has an equivalent gene or two-dimensional image (Figure 4-10). Each coloured pixel of this image represents the value of each attribute normalised in the interval [0, 1]. Figure 4-10 (top view) shows the genes corresponding to the first eight cases from CB1. Each point of the interval was linearly mapped to a palette of colours. Figure 4-10 (bottom) shows part of a gene (i.e. the case-problem) used for retrieval. Each pixel is concentrated in one point and a spline curve is used as a mathematical fit to the data. This
curve represents each gene’s image function \( f(x) \) that is used in global matching. Hence if queries are represented as genes with equivalent shapes generated by spline functions, the problem of finding a similar case to a query is converted to the problem of matching the shapes of the genes corresponding to the case and the query.

![Diagram of gene codification and moment functions](image)

**Figure 4-10. Codification of cases as genes**

a) Genes corresponding to the first eight cases from CB1. Each attribute was linearly mapped to a palette of colours. The value of the properties closest to zero tends to colour blue and the features closest to 1 are red-coloured.

b) A gene’s image function. A *spline* curve is used to join a gene’s pixel points to generate the gene’s image function \( f(x) \). The domain of the function is \( x \) in the interval \([1, k]\), where \( k \) corresponds to the number of attributes.

### 4.4.5.2 Moment Functions

After a case is codified as a gene, the gene’s image function \( f(x) \) can be used to calculate the gene’s moments since this set of moments can be used in turn to describe globally the geometric characteristics of the image. Several moments’ types have been reported in the
image recognition literature [184]. This has been because different moments deal with
different image complexities. For instance, if only boundary features are analysed (e.g.
silhouette images) Fourier-Mellin descriptors can be used [184]. In order to deal with noise
Zernike moments are preferred over Geometric or standard moments [182].

A general definition of moment functions $\Phi_{pq}$ of order $(p+q)$, of a function $f(x,y)$ can be
given as follows:

$$
\Phi_{pq} = \int_{\zeta} \Psi_{pq}(x,y)f(x,y) \, dx \, dy, \quad p, q = 0,1,2,... \quad (4-13)
$$

where, $\Psi_{pq}(x,y)$ is a continuous function of $(x,y)$ in $\zeta$, known as the basis set or moment
weighting kernel. The indices $p, q$ denote the degrees of the coordinates $x$, and $y$ respectively
as defined inside the function $\Psi$. Depending on the basis set $(\Psi_{pq})$ chosen, a variety of
moment functions $(\Phi_{pq})$ have been defined. They range from easy to complex moment
functions having different properties, and used for different applications. These include
moment based algorithms for edge detection, texture segmentation, image reconstruction, and
clustering [184]. Historically, more powerful moment functions have been developed to deal
with image complexity (e.g. noise due to image capture, image rotation). In early applications,
in which silhouettes images were mostly used, simple geometric moment functions were
enough, for instance, to generate a set of invariants which were used for automatic character
recognition [185]. However, as the information contained in images grew, more complex
moments were developed in many diverse applications to address new image requirements.
For instance, Zernike moments enable properties like orthogonality, rotation invariance,
robustness to moderate levels of noise, and efficiency in terms of the non-redundancy
provided by lower order moments to be achieved. Zernike moments contain information
about an image that is independent of the size, planar position and relative angular orientation
[182].

Given that a case is expressed as a "mathematical function" represented graphically as a
2D image (gene) rather than an intensity image\textsuperscript{12}, it is to be expected that these functions will
result in important characteristics that reduce the complexity of the image moment functions
used in this thesis. For instance, because the image was not obtained with a photosensitive
device, there is no geometric distortion associated with the optics when generating the image,
there is no noise associated with its acquisition and there is no need to normalise the image

---

\textsuperscript{12} Intensity images measure the amount of light impinging on a photosensitive device. The input to the
photosensitive device, typically a camera, is the incoming light, which enters the camera's lens and hits the
image plane.
with respect to scale, translation or rotation because the gene is already invariant to these factors. Therefore, simple moment functions can be used to characterise cases described as gene images. They include, geometric moments, central moments, and Legendre moments. These moment functions are described as follows.

**Geometric Moments.** Geometric moments are also referred to as Cartesian moments or regular moments [184]. They are the simplest moment functions; with the kernel function \( \Psi_{pq}(x, y) \) defined as a product of the pixel coordinates. They can be easily implemented and computed. The \((p+q)\) order two-dimensional geometric moments are denoted by \( m_{pq} \), and can be expressed as:

\[
mpq = \int_{\zeta} x^p y^q f(x, y) \, dx \, dy, \quad p, q = 0, 1, 2, \ldots \quad (4-14)
\]

where \( \zeta \) is the region of the pixel space in which \( f(x, y) \) is defined.

If it is assumed that the intensity function \( f(x, y) \) is piece-wise continuous and bounded in the region \( \zeta \), then by the uniqueness, and existence theorem [186]:

- The moments \( m_{pq} \) of all orders exist and are finite.
- The moment sequence \( \{m_{pq}\} \) is uniquely determined by the intensity function \( f(x, y) \), and, conversely.

Variations in the definition of geometric moments given in equation (4-14) have been reported and depending on the application area some commonly used definitions can be found in the literature [184]. For instance, silhouette moments, boundary moments, standard moments and range moments have been defined to calculate moments from binary images, boundary contours of an object, normalised images and range images. A binary image is one in which the function \( f(x, y) \) takes only two values viz. 0 and 1. In an image represented by its edge contour only its boundary points are used. Geometric moments can also been calculated from images which have been normalised with respect to scale, translation and rotation. The intensity function \( f(x, y) \) at pixel of a range image expresses the distance between a known reference frame (e.g. a camera) and a visible point (in the corresponding object point) [184].

**Central Moments.** It is often convenient to evaluate the moments with the origin of the reference system shifted to the intensity centroid of the image [185]. This transformation makes the moment computation independent of the position of the image reference system. Central moments \( (\mu_{pq}) \) of order \((p+q)\) are defined as:

\[
\mu_{pq} = \int_{\zeta} (x - x_0)^p(y - y_0)^q f(x, y) \, dx \, dy, \quad p, q = 0, 1, 2, \ldots \quad (4-15)
\]

The intensity centroid \((x_0, y_0)\) is given by
\[ x_0 = \frac{m_{10}}{m_{00}} \]  

(4-16)

\[ y_0 = \frac{m_{01}}{m_{00}} \]  

(4-17)

where the first order functions \( m_{10} \) and \( m_{01} \), calculated using equation (4-14) provide the intensity moment about the \( y \)-axis and \( x \)-axis of the image respectively.

From the definition of the central moments, \( \mu_{00} = m_{00} \); \( \mu_{10} = \mu_{01} = 0 \).

Central moments approximate the function \( f(x) \) about the centroid of the image (see for example Figure 4-11). For instance, the first central moment order (\( N=1 \)), approximates the function \( f(x) \) as horizontal line that crosses the centroid point. As the moment’s order, \( N \), is increased, the set of central moments will eventually approximate \( f(x) \), this is they will describe the shape of the image.

![Central moments approximation](image)

Figure 4-11. Central moments approximation of image functions

Central moments are used to approximate a function \( f(x) \). They are calculated from the centroid of the image. For a symmetrical image, this point corresponds to the middle point in \( x \) of \( f(x) \) (left). Moments of order \( N \) approximate the function at various degrees. Eventually, when \( N \) tends to infinity \( f(x) \) is fully approximated (black line right).

**Legendre Moments.** Legendre moments are moments with an orthogonal basis to attain zero value of redundancy measure, so that the moments correspond to independent characteristics of the image [184]. The kernels of the Legendre moments are products of Legendre polynomials defined along rectangular image coordinate axes inside a unit circle.

The Legendre moments \( L_{pq} \) of order \( p+q \) are defined as:
where the functions $P_n(x)$ denotes the Legendre polynomial which can be calculated by the recursive relation [184]:

$$P_n(x) = \frac{(2n-1)xP_{n-1}(x)-(n-1)P_{n-2}(x)}{n}$$  \hspace{1cm} (4-19)

### 4.4.5.3 Case Retrieval based on Moment Functions

In pattern recognition applications, the shape representation characteristics of moment functions have been effectively used in recognising object features within images [29]. The most common method deemed for retrieval moment-based recognition algorithms is the comparison of feature vectors of a set of reference images with the given image of an unknown object [184]. A feature vector of a gene (case) is the set of moment functions of that gene. The feature vectors are represented by functions of moments of different orders, the higher the order, the higher the level of detail of the global shape characteristics of the image.

The set of all $V^{(c)}$'s constitute the set feature vectors for the genes used for retrieval (Figure 3-7). Case-gene and query-gene feature vectors are represented as:

$$V^{(c)} = \{v_1^{(c)}, v_2^{(c)}, v_3^{(c)}, \ldots, v_n^{(c)}\}$$ and $$V^{(q)} = \{v_1^{(q)}, v_2^{(q)}, v_3^{(q)}, \ldots, v_n^{(q)}\}$$ respectively.

<table>
<thead>
<tr>
<th>Gene or case</th>
<th>Feature vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c)</td>
<td>1 2 3 ... n</td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>$v_1^{(c)}$, $v_2^{(c)}$, $v_3^{(c)}$, ..., $v_n^{(c)}$</td>
</tr>
</tbody>
</table>

**Figure 4-12.** Feature vector representation of images (genes).

For a feature vector comprising functions of moments, the most common matching algorithm employed is the weighted Euclidean distance between the query or feature vectors...
\[ d(V^Q, V^C) = \left( \sum_{i=1}^{n} \rho_i \left( v_{i}^Q - v_{i}^C \right)^2 \right)^{1/2} \]  

(4-20)

where \( \rho_i \) denotes the weight added to the component \( v_i \) to balance the variations in the dynamic range exhibited by moments of different orders. The inverse of the variance of the column \( v_i^{(k)} \) is frequently used as the weight [184 p. 83]:

\[ \rho_i = \frac{N}{\sum_{i=1}^{n} \left( v_i^c - \bar{v}_i \right)^2} \]  

(4-21)

where,

\[ \bar{v}_i = \frac{\sum_{i=1}^{n} v_i^c}{N} \]  

(4-22)

Other matching algorithms that could be employed include the correlation coefficient method, the logarithmic magnitude distance measure, the minimum mean distance rule and a statistical measure which selects the vector which has minimum difference between the \textit{min} and \textit{max} values among an image and a reference image [184 p. 83]. [184 p. 83]

4.4.5.3.1 Global Feature Weighting

In order to take into account the relative importance of an attribute, a global weighting scheme was used. For local retrieval (section 4.4.4) this was simply done by assigning a weight value, \( w_k \) to each feature \( k \).

For the matching of feature vectors based on image moments, the introduction of feature weights is more complex. If the weighted Euclidean distance is used for moments' feature vector matching (equation (4-21)), the \( \rho_i \) do not correspond to the attribute weights. They account for the variations in the dynamic range exhibited by moments of different orders [184 p. 83].

Feature weights multiply \textit{the square of the difference between features}, for most distance functions used (see section 4.4.4) i.e. feature weights frequently multiply the square of the difference between features for the case-query pair. Accordingly, if the difference between the moment's image-case and the moment's image-query is taken and multiplied it by a feature weight function \( W(x,y) \), this function, will need to take into account the importance of each
attribute (i.e. a function of \(f(x,y)\)). That is, the difference between the case, \(\Phi^C_{pq}\), and query, \(\Phi^Q_{pq}\), moment functions of order \((p+q)\), of the function \(f(x,y)\) is:

\[
\Phi^C_{pq} - \Phi^Q_{pq} = \iint \Psi_{pq}(x,y) W(x,y) [f_c(x,y) - f_Q(x,y)]^2 \, dx \, dy, \quad p, q = 0,1,2,\ldots
\]  

(4-23)

solving, and rearranging,

\[
\Phi^C_{pq} - \Phi^Q_{pq} = \iint \Psi_{pq}(x,y) W(x,y) [f_c^2(x,y) + 2f_c(x,y)f_Q(x,y) + f_Q^2(x,y)] \, dx \, dy \\

+ \iint \Psi_{pq}(x,y) W(x,y) f_c^2(x,y) \, dx \, dy \\

= \iint \Psi_{pq}(x,y) W(x,y) f_c^2(x,y) \, dx \, dy - 2 \iint \Psi_{pq}(x,y) W(x,y) f_c(x,y) f_Q(x,y) \, dx \, dy \\

+ \iint \Psi_{pq}(x,y) W(x,y) f_Q^2(x,y) \, dx \, dy
\]

results in:

\[
\Phi^C_{pq} - \Phi^Q_{pq} = \Phi^C_{pq}^2 - 2\Phi^C_{pq} \Phi^Q_{pq} + \Phi^Q_{pq}^2
\]

(4-24)

Equation (4-24) is a non-linear function of the difference between the case and query moments’ feature vectors, which implies that the difference between the case and query moments’ feature vectors cannot be expressed as a simple arithmetic subtraction of their moments. For the matching of feature vectors based on image moments, the corresponding one-dimensional geometric, central and Legendre moments of order \(p\) that take into account the anisotropy of the data according to eq (4-23) are given by:

\[
m^C_p - m^Q_p = \int x^p W(x) [f_c(x) - f_Q(x)]^2 \, dx,
\]

(4-25)

\[
\mu^C_p - \mu^Q_p = \int (x-x_0)^p W(x) [f_c(x) - f_Q(x)]^2 \, dx,
\]

(4-26)

\[
L^C_p - L^Q_p = \frac{(2p+1)}{2} \int P_p(x) W(x) [f_c(x) - f_Q(x)]^2 \, dx
\]

(4-27)

where \(p=0,1,2,\ldots N\) is the moment’s order; \(W(x)\) is the weight for attribute \(x\); \(f_c(x)\) and \(f_Q(x)\) are the spline functions for the gene case and query; the function \(P_p(x)\) denotes the Legendre polynomial which can be calculated by the recursive relation given in equation (4-19).

4.4.5.3.2 The Global Case Retrieval Algorithm

The retrieval algorithm for matching cases represented as two-dimensional images or genes was implemented in MATLAB [187]. Firstly, the case-problem data were normalised to lie in the interval \([0,1]\) and then, a set of geometrical, central, and Legendre one-dimensional
moments were calculated according to eqs. (4-25), (4-26) and (4-27). The pseudo-code is represented in Figure 4-13 below.

```
for all Genes (cases),
    Make each gene as a query
    Define feature weight vector
        W = [w_l, w_2, ..., w_k]
    Define property index X
        X = [X_1, X_2, ..., X_k]

where
    deltaX_k = Max_Y - Min_Y, Min_Y ≠ 0
    X_k = Min_Y + deltaX_k
    where k is the number of attributes, Min_Y and Max_Y are arbitrary values

Define moment’s order N

Calculate Moments’ Difference

    by using geometric one-dimensional moments
        m_p^c - m_p^q = \int x^p W(x)[f_c(x) - f_q(x)]^2 dx, p = 0,1,... N

    by using central one-dimensional moments
        \mu_p^c - \mu_p^q = \int (x-x_0)^p W(x)[f_c(x) - f_q(x)]^2 dx, p = 0,1,... N
        calculate x_0 using equation (4-17) for the one-dimensional case

    by using Legendre one-dimensional moments
        L_p^c - L_p^q = \frac{(2p+1)}{2} \int x^p W(x)[f_c(x) - f_q(x)]^2 dx, p = 0,1,... N
        calculate P_p (x) using equation (4-19)

Calculate distance

    d(m_p^c - m_p^q) = \sqrt{\sum_{p=0}^{N} (m_p^c - m_p^q)^2}

Select the minimum difference i.e. the most similar case
    min_d(m_p^c - m_p^q)
end
```

Figure 4-13. Pseudo-code for the retrieval algorithm for matching cases represented as two-dimensional images or genes.

4.4.6 Analysis of the Results

A comparison between distance measures used for each retrieval algorithm (i.e. local using standard distance functions and global using moment functions) can be performed using the k-means method of cluster analysis. The reason behind the use of a cluster analysis is due to the fact that there are no probes or test cases i.e. cases with the “correct” formulation to assess
retrieval in terms of which algorithm performs better. For this reason the retrieval methods cannot be compared to each other in terms of which one gives a better result (i.e. which is more accurate when compared to a test case). However, the retrieval algorithms can be contrasted to each other in terms of how close or different their answers are. Before attempting to contrast the local and global retrieval approaches, the distance functions used (i.e. Euclidean, City, Chebychev, Divergence, Canberra, D7, D8, D9 and D10) were compared between them by using a k-means cluster analysis. This clustering method classifies variables into groups and allow for finding some structure or pattern in the results that are very difficult to obtain with a simple data analysis. The k-means cluster analysis has the following objectives:

1. To indicate if there were a statistically significant difference between:
   a) the standard distance functions (i.e. if there were a group of distances retrieving the same cases);
   b) the moment’s type (i.e. geometric, central and Legendre) and moment’s order (low i.e. \( p \leq 3 \) and high \( p > 3 \)) on the retrieval of cases (i.e. how different are the retrieved cases when different moment’s order and types are used);
2. To indicate if the local and global algorithms provide the same answers or if on the contrary they are different (and hence the moment-based algorithm results in an alternative retrieval algorithm when numerical attributes are used).

The k-means cluster analysis therefore indicates if there is a significant difference between the distances i.e. between-group similarity, for all the instances or queries available.

Computationally, the k-means clustering method can be thought of as an analysis of variance (ANOVA) in reverse [188]. Starting with \( k \) random clusters, the algorithm, will allocate objects with the goal of (i) minimising the variability within clusters and (ii) maximising the variability between clusters, which is analogous to ANOVA in reverse. In ANOVA, the significance test evaluates the between-group variability against the within-group variability when computing the significance test for the hypothesis that the means in the groups are different from each other [175]. Therefore, ANOVA results are part of the standard output from a k-means clustering analysis. The magnitude of the F values from the analysis of variance performed on each instance is an indication of how well the respective instance discriminates between clusters. The greater the F value the better the instance discriminates between clusters. The F values are calculated by dividing the mean squares values of the groups (clusters) \( M_{SL} \) and the errors \( M_{SE} \). These in turn are calculated by dividing the sum of squares terms for the groups and errors by their respective degrees of freedom values [175].
Another way to assess "how distinct clusters are" is by examining the means of each cluster on each instance. Ideally, very different means should be obtained for most if not all instances used in the analysis since this would indicate well differentiation between clusters.

The cluster analysis was done using STATISTICA software [189]. The results from a k-means clustering method depend to some extent on the initial configuration (i.e. cluster means). The option used to compute the initial cluster centres was the one that maximises the initial cluster distances\(^ {\S}\). The number of iterations was set to 10. However, most solutions were obtained after one or two iterations. The analysis was performed by choosing initially two clusters and then increasing the number of clusters to three when variables (i.e. distances) presented high variability.

### 4.4.7 Summary

Considering a case's image as a 2D intensity distribution \( f(x) \), different one-dimensional moment functions including geometric, central and Legendre moments, have been defined and subsequently used in case retrieval. The problem of assigning weights to the features (\( W_k \)) has also been studied when matching moments. It was expected that matching moments of higher order would retrieve different cases than lower order moments. This is because moments of order zero up to three can be used to represent gross level image features, whilst higher moment orders (\( p > 3 \)) enable finer details about the image to be represented and hence increase the probability of retrieving a more similar case.

The summary of the method used to answer this chapter research question and sub questions is illustrated in Figure 4-14 and Figure 4-15.

---

\(^{\S}\) In STATISTICA, this option specifically (i) selects the first number of clusters to be the respective cluster centres; (ii) initial cluster centres are replaced if the smallest distance of following cases to any of the cluster centres is larger than the smallest distance between the clusters; if this is not the case, then (iii) initial cluster centres are replaced if the smallest distance of subsequent cases from a cluster centre is larger than the distance between the clusters.
## Collection of Data
From testing lab

<table>
<thead>
<tr>
<th>Record</th>
<th>Polyol A</th>
<th>Polyol B</th>
<th>Polyol C</th>
<th>Polyol D</th>
<th>Polyol E</th>
<th>Additive Black</th>
<th>Silicone A</th>
<th>Water</th>
<th>Isocyanate A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
</tr>
<tr>
<td>1</td>
<td>48.31</td>
<td>15.18</td>
<td>26.22</td>
<td>9.25</td>
<td>14.84</td>
<td>1.04</td>
<td>0.94</td>
<td>14.42</td>
<td>1.66</td>
</tr>
<tr>
<td>2</td>
<td>37.27</td>
<td>15.18</td>
<td>37.27</td>
<td>9.25</td>
<td>14.84</td>
<td>1.04</td>
<td>0.94</td>
<td>14.42</td>
<td>1.66</td>
</tr>
</tbody>
</table>

## Generate Cases
Select sub data set
i.e. select only important features

<table>
<thead>
<tr>
<th>Case</th>
<th>Polyol A</th>
<th>Polyol B</th>
<th>Polyol C</th>
<th>Polyol D</th>
<th>Polyol E</th>
<th>Additive Black</th>
<th>Silicone A</th>
<th>Water</th>
<th>Isocyanate A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
</tr>
<tr>
<td>1</td>
<td>48.31</td>
<td>15.18</td>
<td>26.22</td>
<td>9.25</td>
<td>14.84</td>
<td>1.04</td>
<td>0.94</td>
<td>14.42</td>
<td>1.66</td>
</tr>
<tr>
<td>2</td>
<td>37.27</td>
<td>15.18</td>
<td>37.27</td>
<td>9.25</td>
<td>14.84</td>
<td>1.04</td>
<td>0.94</td>
<td>14.42</td>
<td>1.66</td>
</tr>
</tbody>
</table>

## Formulation Properties

<table>
<thead>
<tr>
<th>Density</th>
<th>Tension</th>
<th>Elongation</th>
<th>Tear</th>
<th>Hardness 1 s</th>
<th>Hardness 30 s</th>
</tr>
</thead>
<tbody>
<tr>
<td>kg/m³</td>
<td>MPa</td>
<td>%</td>
<td>N/mm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>280.62</td>
<td>3.14</td>
<td>77.20</td>
<td>0.73</td>
<td>18.33</td>
</tr>
<tr>
<td>2</td>
<td>290.63</td>
<td>3.53</td>
<td>90.10</td>
<td>0.84</td>
<td>22.67</td>
</tr>
</tbody>
</table>

## Retrieval Using Distances

### A. Using the Euclidean distance

\[
d_{euc}(c_i - q) = \sqrt{\sum_{t=1}^{n} w_t (c_{i,t} - q_t)^2}
\]

## Select case with minimum distance

<table>
<thead>
<tr>
<th>Case</th>
<th>Density</th>
<th>Tension</th>
<th>Elongation</th>
<th>Tear</th>
<th>Hardness 1 s</th>
<th>Hardness 30 s</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0.6545</td>
<td>0.6908</td>
<td>0.6086</td>
<td>0.4768</td>
<td>0.19</td>
<td>0.08</td>
</tr>
</tbody>
</table>

## Retrieve case-solution

<table>
<thead>
<tr>
<th>Case</th>
<th>Polyol A</th>
<th>Polyol B</th>
<th>Polyol C</th>
<th>Polyol D</th>
<th>Polyol E</th>
<th>Additive Black</th>
<th>Silicone A</th>
<th>Water</th>
<th>Isocyanate A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
<td>Quantity</td>
</tr>
<tr>
<td>4</td>
<td>48.31</td>
<td>15.18</td>
<td>26.22</td>
<td>9.25</td>
<td>14.84</td>
<td>1.04</td>
<td>0.94</td>
<td>14.42</td>
<td>1.66</td>
</tr>
</tbody>
</table>

Figure 4-14. Methodology to retrieve cases using the Euclidean distance.

Firstly, the data collected from both the laboratory (i.e. the formulation recipe) and the testing lab (i.e. mechanical properties) are used to generate the case-solution and case-problem respectively. After selecting a subset of data, the cases attribute’s values are normalised. Retrieval is performed by calculating the distance between the case-problems in the case base and the query. The minimum distance is used to retrieve its corresponding case-solution.
## RETRIEVAL USING MOMENTS

<table>
<thead>
<tr>
<th>Calculate attribute index</th>
<th>Query-Problems</th>
<th>Case-Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Query</td>
<td>Case</td>
</tr>
<tr>
<td></td>
<td>Density</td>
<td>Tension</td>
</tr>
<tr>
<td>X</td>
<td>1</td>
<td>1.2</td>
</tr>
<tr>
<td>q</td>
<td>0.9</td>
<td>0.8</td>
</tr>
<tr>
<td>w</td>
<td>1.0</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Calculate moments difference

\[ m^c_p - m^q_p = \sum_{x=1}^{attributes} X^p W(x) \left[ f^c(x) - f^q(x) \right]^2 \]

- Example using Geometric moments
- Example using moment's order \( p=0 \)
- Calculate for low moment's order
  - \( p=3 \)

Calculate distances between moments

\[ d(m^c_3 - m^q_3) = \sqrt{(m^c_1 - m^q_1)^2 + (m^c_2 - m^q_2)^2 + (m^c_3 - m^q_3)^2} \]

- Calculate for low moment's order
- \( p=20 \)

Retrieve case-solution \( min \ d \)

### Retrieved Case-Solution

<table>
<thead>
<tr>
<th>Case</th>
<th>Polyl A</th>
<th>Polyl B</th>
<th>Polyl C</th>
<th>Polyl D</th>
<th>Polyl E</th>
<th>Catalyst A</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>26.22</td>
<td>15.19</td>
<td>48.31</td>
<td>9.25</td>
<td>1.04</td>
<td>0.94</td>
</tr>
</tbody>
</table>

Figure 4.15: Methodology to retrieve cases using Geometric Moments

Firstly, data are normalised and a property index \( X \) is assigned to each attribute. Attribute weights are also specified. Retrieval is performed by first, calculating the difference between the moments corresponding to the case and the query, and then the distance between them is simple obtained by calculating the norm's difference, which corresponds to the Euclidean distance. The minimum distance is used to retrieve the case-solution of the closest case.
4.5 Results and Discussion

The results of the best match to a query for both case-bases CB1, and CB2 using both the feature vector and the genetic representation (i.e. moment function based comparison) retrieval algorithms are presented in this section. All the cases from each CB were chosen as queries for the retrieval to avoid bias in the conclusions. For the local retrieval using a nearest neighbour algorithm results for nine distance functions: Euclidean, City block, Chebychev, Canberra, Divergence, D7, D8, D9 and D10 (see Table 4-4 p.101) were computed. Both local retrieval (nearest neighbour algorithm using quantitative distance functions) and global retrieval (retrieval based on image moment features) were tested with two sets of features weights. The first set corresponds to the case using equal feature weights (i.e. \( w_k = 1 \)) and the second set of weights was given by a PU's expert to indicate feature's relevance i.e. for CB1, \( w_k = \{1, 0.9, 0.6, 0.4, 0.2, 0.1\} \), for CB2, \( w_k = \{1, 0.8, 0.6, 0.4, 0.2, 0.1\} \). The results of the retrieved cases for both case-bases CB1 and CB2 using both local and global retrieval with either equal or expert feature weights are displayed in the following tables.
Table 4-5. Retrieved cases using standard distance functions using equal feature weights for the CB1.

An instance or record corresponds to a retrieved case. Similarity is calculated by comparing a query with each case in the case-base. For example, when the query is the case 1, the Euclidean and Chebychev distance functions retrieve the case No. 4 in the CB1 as the most similar one to that query while the other distance functions retrieve the case No. 6.

<table>
<thead>
<tr>
<th>Instance</th>
<th>EUC</th>
<th>CITY</th>
<th>CANB</th>
<th>CHEB</th>
<th>DIV</th>
<th>D7</th>
<th>D8</th>
<th>D9</th>
<th>D10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>7</td>
<td>7</td>
<td>9</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>8</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td>11</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>11</td>
<td>11</td>
<td>14</td>
</tr>
<tr>
<td>13</td>
<td>15</td>
<td>15</td>
<td>12</td>
<td>15</td>
<td>12</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>12</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>13</td>
<td>13</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>13</td>
<td>13</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>17</td>
<td>15</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>16</td>
<td>14</td>
<td>16</td>
<td>14</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>14</td>
</tr>
<tr>
<td>18</td>
<td>16</td>
<td>16</td>
<td>17</td>
<td>16</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
</tbody>
</table>

Table 4-6. Retrieved cases using equal feature weights from the CB1 for different queries when using low and high order Geometric, Central and Legendre moment functions.

Similarity is calculated by comparing a query-image with each case-image (or case-gene) in the case-base. For example, when the query is the case 17, both low order Geometric and Legendre moments and Central moments of any order retrieved case No. 16 as the most similar to the query whilst high order Geometric and Legendre retrieved case No. 18. However, results vary through different queries. For example, for query 1, all moments retrieved the case No. 4, which was the same retrieved by the Euclidean and Chebychev distances (see Table 4-5 on the left).

<table>
<thead>
<tr>
<th>Instance</th>
<th>GeomLow</th>
<th>GeomHigh</th>
<th>CentLow</th>
<th>CentHigh</th>
<th>LegLow</th>
<th>LegHigh</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>4</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>7</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
<td>11</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td>18</td>
<td>11</td>
<td>11</td>
<td>14</td>
<td>18</td>
</tr>
<tr>
<td>13</td>
<td>15</td>
<td>18</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>18</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>15</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>13</td>
<td>14</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>15</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>18</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>18</td>
</tr>
<tr>
<td>18</td>
<td>16</td>
<td>17</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>17</td>
</tr>
</tbody>
</table>
Table 4-7. Retrieved cases using standard distance functions using expert feature weights for the case-base 1.
When the query is the case 1, all distance functions retrieve the case No. 6 as the most similar case to the query 1. It appears that for most queries all distances retrieve the same query. Although there are a few exceptions, see for example query 4.

<table>
<thead>
<tr>
<th>Instance</th>
<th>EUC</th>
<th>CITY</th>
<th>CANB</th>
<th>CHEB</th>
<th>DIVER</th>
<th>D7</th>
<th>D8</th>
<th>D9</th>
<th>D10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>9</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td>11</td>
<td>14</td>
<td>11</td>
<td>14</td>
<td>14</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>13</td>
<td>15</td>
<td>15</td>
<td>12</td>
<td>15</td>
<td>12</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>12</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>11</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>13</td>
<td>13</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>12</td>
<td>13</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>16</td>
<td>14</td>
<td>16</td>
<td>14</td>
<td>14</td>
<td>16</td>
<td>16</td>
<td>14</td>
</tr>
<tr>
<td>18</td>
<td>16</td>
<td>16</td>
<td>12</td>
<td>16</td>
<td>12</td>
<td>12</td>
<td>16</td>
<td>16</td>
<td>17</td>
</tr>
</tbody>
</table>

Table 4-8. Retrieved cases using expert feature weights from the CB1 for different queries when using low and high order geometric, central and Legendre moment functions.
For query No.17, low order geometric and Legendre moments and central moments of any order retrieved case No. 16 as the most similar to the query whilst high order geometric and Legendre retrieved case No. 18. However, results vary across queries. For example, for query 1, all moments except high order geometric and Legendre (which retrieved case No. 4) retrieved the case No. 6, which was the same retrieved by standard distance functions (see Table 4-7).

<table>
<thead>
<tr>
<th>Instance</th>
<th>GeomLow</th>
<th>GeomHigh</th>
<th>CentLow</th>
<th>CentHigh</th>
<th>LegLow</th>
<th>LegHigh</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>3</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td>14</td>
<td>14</td>
<td>11</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>12</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td>17</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>17</td>
</tr>
<tr>
<td>13</td>
<td>15</td>
<td>18</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>18</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>17</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>17</td>
</tr>
<tr>
<td>15</td>
<td>13</td>
<td>16</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>15</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>14</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>14</td>
</tr>
<tr>
<td>18</td>
<td>16</td>
<td>13</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>13</td>
</tr>
</tbody>
</table>
Table 4-9. Retrieved cases using standard distance functions with equal feature weights for the CB2.

When the query is the case 1, Euclidean and Chebychev distance functions retrieve the case No. 28 as the most similar case while the other distance functions retrieved case No. 6.

<table>
<thead>
<tr>
<th>Instance</th>
<th>EUC</th>
<th>CITY</th>
<th>CANB</th>
<th>CHEB</th>
<th>DIV</th>
<th>D8</th>
<th>D9</th>
<th>D10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>28</td>
<td>6</td>
<td>28</td>
<td>6</td>
<td>28</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>19</td>
<td>19</td>
<td>33</td>
<td>8</td>
<td>33</td>
<td>33</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>3</td>
<td>32</td>
<td>1</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>10</td>
<td>18</td>
<td>23</td>
<td>2</td>
<td>18</td>
<td>2</td>
<td>18</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>21</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>6</td>
<td>28</td>
<td>28</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>12</td>
<td>20</td>
<td>4</td>
<td>12</td>
<td>20</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>8</td>
<td>12</td>
<td>19</td>
<td>19</td>
<td>10</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>10</td>
<td>23</td>
<td>10</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>12</td>
<td>12</td>
<td>8</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>27</td>
<td>27</td>
<td>1</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>13</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>14</td>
<td>23</td>
<td>23</td>
<td>19</td>
<td>23</td>
<td>19</td>
<td>30</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>25</td>
<td>18</td>
<td>17</td>
<td>18</td>
<td>25</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>16</td>
<td>26</td>
<td>30</td>
<td>30</td>
<td>14</td>
<td>26</td>
<td>30</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>18</td>
<td>25</td>
<td>25</td>
<td>21</td>
<td>17</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>19</td>
<td>2</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>20</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>21</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>22</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>22</td>
<td>18</td>
<td>17</td>
<td>18</td>
<td>18</td>
<td>25</td>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>23</td>
<td>4</td>
<td>4</td>
<td>18</td>
<td>4</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>8</td>
<td>8</td>
<td>33</td>
<td>25</td>
<td>19</td>
<td>33</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>25</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>26</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>27</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>28</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>29</td>
<td>21</td>
<td>5</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>30</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>31</td>
<td>27</td>
<td>27</td>
<td>16</td>
<td>30</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>32</td>
<td>27</td>
<td>28</td>
<td>3</td>
<td>27</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>33</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>34</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
</tr>
</tbody>
</table>

Table 4-10. Retrieved cases from the CB2 using equal feature weights for different queries when using low and high order geometric, central and Legendre moment functions.

When the query is the case 1, all moments of any order except high order geometric and Legendre moments (which retrieved the case No 33) retrieved the case No. 28, which was the same case retrieved by the Euclidean distance function (refer to Table 4-9 on the left).
Table 4-11. Retrieved cases using standard distance functions using expert feature weights for the CB2.

For query 1, all distance functions retrieve the case No. 6 as the most similar case to the query 1.

<table>
<thead>
<tr>
<th>Instance</th>
<th>EUC</th>
<th>CITY</th>
<th>CANB</th>
<th>CHEB</th>
<th>DIVER</th>
<th>D7</th>
<th>D8</th>
<th>D9</th>
<th>D10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
</tr>
<tr>
<td>3</td>
<td>24</td>
<td>1</td>
<td>24</td>
<td>32</td>
<td>32</td>
<td>1</td>
<td>32</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>7</td>
<td>23</td>
<td>7</td>
<td>20</td>
<td>23</td>
<td>7</td>
<td>7</td>
<td>18</td>
</tr>
<tr>
<td>5</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
</tr>
<tr>
<td>6</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>13</td>
<td>20</td>
<td>13</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>33</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>19</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>10</td>
<td>23</td>
<td>10</td>
<td>23</td>
<td>23</td>
<td>8</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>10</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>1</td>
<td>27</td>
<td>16</td>
<td>1</td>
<td>27</td>
<td>1</td>
<td>1</td>
<td>27</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>13</td>
<td>20</td>
<td>13</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>13</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>14</td>
<td>19</td>
<td>8</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>8</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>15</td>
<td>25</td>
<td>25</td>
<td>18</td>
<td>25</td>
<td>18</td>
<td>18</td>
<td>25</td>
<td>25</td>
<td>18</td>
</tr>
<tr>
<td>16</td>
<td>11</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>15</td>
<td>21</td>
<td>21</td>
<td>15</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>19</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>20</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>21</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>22</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>23</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>12</td>
<td>19</td>
<td>8</td>
<td>19</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>24</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>25</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>15</td>
<td>15</td>
<td>17</td>
</tr>
<tr>
<td>26</td>
<td>30</td>
<td>30</td>
<td>34</td>
<td>30</td>
<td>34</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>27</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>28</td>
<td>6</td>
<td>27</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>29</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>30</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>31</td>
<td>27</td>
<td>27</td>
<td>6</td>
<td>27</td>
<td>6</td>
<td>6</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>32</td>
<td>31</td>
<td>27</td>
<td>3</td>
<td>31</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>33</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>2</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>34</td>
<td>33</td>
<td>26</td>
<td>26</td>
<td>33</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>33</td>
</tr>
</tbody>
</table>

Table 4-12. Retrieved cases from the CB2 using expert feature weights for different queries when using low and high order geometric, central and Legendre moment functions.

For query 1, all moments of any order except high order geometric and Legendre moments (which retrieved the case No. 33) retrieved the case No. 6, which was the same case retrieved by the Euclidean distance function (refer to Table 4-11 on the left).
4.5.1 Retrieval Using Equal Feature Weights

4.5.1.1 Local retrieval

Results from Table 4-5 corresponding to the retrieved cases using standard distance using equal feature weights for CB1 were plotted in Figure 4-16. It can be discerned from the diagram that some of the distance functions used in the nearest neighbour algorithm for CBR provide the same information since the data points overlap each other. It is also to be noted that some distance functions (e.g. D7 and D8) are mathematically similar (see eqs. (4-9) and (4-10) p. 102) and therefore may retrieve the same case (e.g. if only positive attributes are used the denominator in each equation is the same so the equation give the same answer). To avoid redundancy in the analysis, a k-means cluster analysis was performed to identify if there were any homogeneous subgroups of distance functions that were giving the same information.

Figure 4-16. Retrieval using standard distance functions for CB1.
It can be seen that most distances overlap each other. Even in some queries (e.g. 10, 11, 14) all distances retrieved the same case.

The cluster analysis allows discerning some sort of structure that is very difficult to discern only from Table 4-5. The clusters or groups of distance functions given by a k-means cluster analysis used in the retrieval of cases from CB1 are shown in Figure 4-17 and Figure 4-18. It can be seen from the diagrams that some of the data points overlap each other i.e. some distance functions used in the nearest neighbour algorithm for CBR are providing the same
information. From Figure 4-17 a) it is observed that the Euclidean, City block, Chebychev, D8 and D9 distances retrieve the same case for most queries. Likewise, Divergence, D7, D10, and Canberra’s distance functions (Figure 4-17 b) all converge to a same case, with a few exceptions. Further, adopting a 3-mean clustering, it can be seen from the first cluster (Figure 4-18 a) that either the Euclidean or Chebychev’s distance function will converge to the same case for most queries, i.e. they provide the same information. From the second cluster (Figure 4-18 b) for all the queries except query 18, either city block, D8 or D9’s distance function retrieve the same case. The same pattern is observed in cluster 3 (Figure 4-18 c) in which Canberra, Divergence, D7 or D10 provide the same information, and hence by using either one of them in further analysis is adequate to avoid redundancy.

Figure 4-18 d) and e) show the ANOVA results and the plot of means for the 3-means clustering of distance functions for CB1. It is observed from these that for most queries, the discrimination between clusters is poor. For example, for the 3rd instance (query 3), the test statistic F is 1.69, which is less than the F crit value (5.243 at 5% significance level), indicating a poor cluster discrimination for that query. Note: Only 10 of the 18 queries (i.e. Q_1,2,4,5,6,9,12,13,15,17) have significant F crit values > 5.243. The means of the clusters’ are similar for some instances, Figure 4-18 e), indicating again a poor discrimination between clusters. This indicates that there is not enough statistically significant evidence to support that the cluster formation is valid. Therefore it is concluded that all of the distance-based functions used in the case retrieval for CB1 are retrieving the same case for a given query. To simplify further analysis, only one distance function (i.e. Euclidean) will be reported for the comparison against moment-based retrieval.
Figure 4-18. k-means cluster analysis with three clusters for the distance functions used in retrieval of cases from CB1 when using equal feature weights.

The number of clusters was set to three by the analyst. Starting with two random clusters, the program will move objects (e.g. distances) in and out of groups (clusters) to (i) minimise the variability of clusters and (ii) maximise variability between clusters. The magnitude of the F values from the ANOVA performed on each instance is an indication of how well the respective instance discriminates between clusters. By examining the means of each cluster (or the difference between cluster's means) on each instance an indication of how well differentiated the clusters is also obtained. If the means are equal (i.e. difference is zero) the instance can belong to either cluster i.e. they are not well differentiated.

For the second case-base (CB2), an equivalent distance-based k-means cluster analysis was performed. The results from Table 4-7 corresponding to the retrieved cases using standard distances using equal feature weights for CB1 were plotted in Figure 4-19. Although not clearly as in Figure 4-16, it can be seen from this figure that some of the data points also
overlap each other i.e. some distance functions used in the nearest neighbour algorithm are yielding similar information. A k-means cluster analysis was performed to identify if there were any homogeneous subgroups of distance functions that were giving the same information that cannot be visualised either from Table 4-7 or Figure 4-19.

Figure 4-19. Retrieval using standard distance functions for CB2.

Figure 4-20 and Figure 4-21 show the clusters or groups of distance functions for CB2 given by a k-means cluster analysis seeking two and three clusters, respectively. In comparison with the results obtained previously for CB1, similar clusters of distances are also evident. For example, for a 2-means clustering the distances are grouped into Euclidean, City block, and Chebychev (Figure 4-20 a), and Divergence, Canberra, D7, D8, D9 and D10 (Figure 4-20 b). It can be observed that the similarity within clusters is not the same for all queries (i.e. the retrieved case according to different distance functions vary within queries). In order to assess how well each intake discriminated between clusters, an analysis of variance is summarised in Figure 4-21 d). It can be observed that only a few queries (i.e. Q_1,3,7,8,32) discriminate well between clusters. For instance, for the query 32, the test statistic F is 8.37, higher than the F_{crit} value (5.243 at 5% significance level), indicating a good cluster discrimination for that particular query. In addition, the clusters' means do not show considerable differences for most queries (see Figure 4-20 e) indicating unsatisfactory discrimination between clusters. There are 10 instances in which the means for clusters one and two are identical, indicating that these two groups cannot be differentiated with the distance-based analysis. There are also 15 instances in which the means for clusters two and three are identical as well. It is concluded that although there are some instances in which the
groups are well discriminated; the between-group differentiation is not statistically significant for most queries. Only 5 queries (i.e. Q1,3,7,8,32) out of 34 discriminate statistically well (refer to the F values in Figure 4-21 d) between distance clusters. From this, it can be concluded that all of the distance-based functions used in the case retrieval for CB2 converge to the same case for a given query.

![Figure 4-20. k-means cluster analysis with $k=2$. CB2](image-url)
Figure 4-21. k-means cluster analysis with k=3 for the distance functions used in retrieval of cases from CB2 when using equal feature weights. The distances functions were clustered into 3 groups. Each cluster comprises three variables. The first contains Euclidean and Chebychev; the second, City block and D10; and the third one Divergence, D7, D8, D9, D10 and Canberra. The magnitude of the F values from the ANOVA performed on each instance is not very high indicating a poor discrimination between clusters for most instances. An examination of the difference between cluster’s means on each instance also indicates poor differentiation between clusters of distances.

4.5.1.2 Global retrieval

The results of the best match to a query for both case-bases CB1, and CB2 using the moment function-based retrieval algorithm tabulated in Table 4-6 and Table 4-10 were plotted in Figure 4-22 and Figure 4-23, respectively. These diagrams show the results for the best image-case match for various image-queries using low (p<3) and high (p>3) order moments. From these figures, it can be observed that depending on the moment’s type (geometric, central or Legendre) and order (low or high) different retrieved cases are obtained. However,
Figure 4-22 b) and Figure 4-23 b) show that central moments of any order retrieve the same case for most queries for both case bases.

Another interesting observation is that low order geometric moments converge to the same case as low order Legendre moments and central moments of any order (Figure 4-24 and Figure 4-25). Correspondingly, their respective high order moments also converge to a same case. A k-means cluster analysis using two clusters confirmed these results (see Figure 4-27).
Figure 4-22. Retrieved case for 18 queries by using low and high order moments
a) Geometric b) Central c) Legendre moments.
High order geometric moments retrieve different cases as its low order moments do for most queries. The same pattern is seen for Legendre moments. Central moments of any order always converge to the same case.

Figure 4-23. Retrieved case for 34 queries by using low and high order moments
a) Geometric b) Central c) Legendre moments.
The same tendency as for CB1 (left) high order geometric and Legendre moments retrieve different cases as their low order moments do for most queries central moments of any order converge to the same case for the majority of the queries.
Figure 4-24. Comparison of low and high order moments for retrieval in CB1. Low order geometric and Legendre moments and central moments of any order retrieve the same case for all queries. In the same manner, high order geometric and Legendre moments converge to the same case.

Figure 4-25. Comparison of low and high order moments for retrieval in CB2. The same pattern is observed that in Figure 4-24 i.e. low order geometric and Legendre moments and central moments of any order converge to the same answer in the same manner as high order geometric and Legendre moments do.
### Members of Cluster Number 1

Cluster contains 2 variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEOM_HIG</td>
<td>LEG_HIG</td>
</tr>
</tbody>
</table>

Distance 0.117851131 0.117851131

### Members of Cluster Number 2

Cluster contains 4 variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Variable</th>
<th>Variable</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEOM_LOW</td>
<td>CENT_LOW</td>
<td>CENT_HIG</td>
<td>LEG_LOW</td>
</tr>
</tbody>
</table>

Distance 0.212459147 0.212459147 0.212459147 0.117851131

---

**a) Analysis of Variance**

Figure 4-26. \( k \)-means cluster analysis with \( k=2 \) for the moment-based retrieval functions for CB1 when using equal feature weights.

It can be seen from this ANOVA that most queries discriminate well between clusters as their \( F \) values are high (some values tend to infinity i.e. most Within SS are zero which makes the \( F \) tend to infinity \( F=\text{Between SS/Within SS}/4 \)).
The greater the F value the better the instance (i.e. query) discriminates between clusters

![Table of data](image)

The cluster analysis shown in Figure 4-26 shows two well differentiated clusters. The first cluster contains two members or variables: high order geometric and Legendre moments while the second consists of four variables: low order geometric, central and Legendre moments and high order central moments (Figure 4-26 b). This figure also reports on the distances of each variable from the respective cluster centre (mean). These distances are small indicating that the variables (or cluster members) are not very distant from the cluster centre (i.e. they are further away from the centres of alternative clusters). Most instances (i.e. queries) differentiate well between clusters as evidenced by the high F values in Figure 4-26 a) and by the variation of the cluster’s means shown in Figure 4-26 c). The same general observations can be seen for CB2 (Figure 4-27).

4.5.1.2.1 Visualisation

Formulators need more user friendly decision support systems to understand and evaluate alternative approaches. A decision support system can generally assess and present information graphically so that the validity of the solution provided by a CBR system can be explained or judged. For retrieval, it would be useful for the user to have the information...
about the difference between the target problem and most similar cases that may help in the adaptation stage. This topic has not received much attention by the CBR research community. Only a few research groups are actively working on this area [162].

In CBR systems, retrieval is one of the tasks more difficult to understand for users. This is because the similarity measure is usually presented as a single numerical value, hiding the implicit knowledge used when matching various cases. Making this information available to the user has potential benefits that can help in simplifying the interpretation of the results, exposing deficiencies in the CBR systems and increasing user trust in the results generated by the system. For complex domains such as the PU formulation, it is important to keep track of the changes done during the formulation design process. If a visual explanation of retrieval can highlight these changes, some insight into the relationships between the components and properties can also be gained.

The information provided to the user can readily be provided in visual formats. The fact that the matching of cases using global retrieval, was implemented in this thesis by first graphically coding the cases and queries into images and then calculating the difference between geometric moments of the query and each case in the case-base, the most similar case to a query can also be visualised graphically (Figure 4-28). By looking for the minimum differences (i.e. blue tones) a user can easily visualise which are the closest cases to a given query. This approach enables the user to look for differences between in colours rather than differences in lines as it has been proposed with the standard local retrieval visualisation technique presented by Massie et al. [162] (see Figure 4-29).

Therefore, the assessment of similarity between cases by using the global retrieval approach can also be visualised supporting the user in deciding which range of formulations best suit their needs.
Calculated geometric moment differences for the query 6 using equal feature weights.

Figure 4-28. Global retrieval visualisation
The matching of genes (i.e. cases) using global retrieval was done by calculating the difference between geometric moments of the query and each case in the case-base. The most similar case to a query is visualised in this figure. By taking the minimum difference, a user can easily visualise which is the closest case to a given query. Blue tones represent least difference between the two. For example the most similar cases to the query 6 represented in this figure correspond to case 27 and case 28. Other potential cases is No. 34.
4.5.2 Retrieval Using Expert's Selected Feature Weights

4.5.2.1 Local retrieval

The performance of the retrieval functions using expert selected feature weights is detailed in this section. The expert feature weights used are summarised in the following table.

Table 4-13. Weights rated by a PU expert scaled between [0, 1] for each feature of CBI and CB2.

<table>
<thead>
<tr>
<th>Feature k</th>
<th>Density</th>
<th>Tensile Strength</th>
<th>Elongation</th>
<th>Tear</th>
<th>Hardness</th>
<th>Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wk for CB1</td>
<td>1</td>
<td>0.8</td>
<td>0.6</td>
<td>0.4</td>
<td>0.2</td>
<td>NA</td>
</tr>
<tr>
<td>Wk for CB2</td>
<td>1</td>
<td>0.8</td>
<td>0.6</td>
<td>NA</td>
<td>0.2</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Following the same approach used in the former section, a cluster analysis was performed to identify any variability between distance functions, or if on the contrary, they are converging to the same case in each instance when the expert feature weights (Table 4-13) are used in the retrieval algorithm. Figure 4-30 and Figure 4-32 show the clusters or groups of distance functions given by a k-means cluster analysis seeking two and three clusters, respectively, for the retrieval of cases from CB1 when using a set of expert feature weights. It can be observed from these figures that some of the distance functions are providing the same...
information. From Figure 4-32 a) it can be observed the Euclidean, City block, Chebychev, D8 and D9 distances retrieve the same case for all queries (except for query 7, 8 and 14). The same can be said for the distances D7, Divergence and Canberra, which converge to a same case, with a few exceptions (Figure 4-32 b). Cluster 3 (Figure 4-32 c), contains only one distance function, D10 indicating some variability with respect to the other clusters.

To quantify how statistically significant is the cluster formation, Figure 4-32 d) and e) show the ANOVA results and the plot of means for the 3-means clustering of distance functions for CB1. It can be observed that for most queries, the discrimination between clusters is poor. For instance, for the query 7 (Q_7), the test statistic F is 0.1818 much less than the F_{crit} value (5.143 at 5% significance level), indicating a poor cluster discrimination for that query. This same pattern is repeated for 5 queries (i.e. Q_7,8,11,14,15). Also from Figure 4-32 e) the clusters’ means are very similar. Indeed, the means for cluster 1 and 2 coincide in 6 occasions, evidencing poor discrimination between these two clusters in those instances.

Contrary to the analysis with equal weights (see section 4.5.1 p.123), 11 instances out of 18 (half of the time) discriminate significantly between clusters. It is possible to conclude that from cluster 1, and the 2-means analysis (Figure 4-30 a) any of the distances in this group give the same answer for most instances. From the second cluster (Figure 4-30 b), the Canberra and D7 functions coincide for all queries, and in turn they coincide partially with Divergence (in 13 instances out of 18). Therefore, it can be said that they retrieve the same case for most queries. The distance D10 is the most scattered. By looking at Figure 4-31 D10 is the most distant variable to the cluster centre it belongs to (i.e. explaining its higher variation in comparison with the other cluster members: Canberra, D7 and Divergence). Also, the distances of the cluster members in cluster 1 and 2 are very small indicating good cluster membership (i.e. these variables are retrieving the same case). The implications of these conclusions are shown graphically in Figure 4-33. In this figure, a distance representative of each cluster has been plotted. It can be seen that the distance functions overlap each other for most instances i.e. they retrieve the same case for most queries. Only for 3 queries (i.e. Q_6,15, 18) they give and independent answer.
a) Cluster 1.

b) Cluster 2

Figure 4-30. $k$-means cluster analysis with $k=2$ for the distance functions used in retrieval of cases from CB1 when using expert feature weights.

<table>
<thead>
<tr>
<th>Members of each cluster</th>
<th>and Distances from Respective Cluster Center</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster 1 contains 4 variables</td>
<td>Variable</td>
</tr>
<tr>
<td>CANBERRA</td>
<td>DIVERGENCE</td>
</tr>
<tr>
<td>Distance</td>
<td>0.52</td>
</tr>
<tr>
<td>Cluster 2 contains 5 variables</td>
<td>Variable</td>
</tr>
<tr>
<td>EUCLIDEAN</td>
<td>CITY</td>
</tr>
<tr>
<td>Distance</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Figure 4-31. Members of each cluster and distances from respective cluster centre
Figure 4-32. k-means cluster analysis with k=3 for the distance functions used in retrieval of cases from CB1 when using expert feature weights.

The number of clusters was set 3 by the analyst. Starting with 2 random clusters, the program will move objects (e.g. distances) in and out of groups (clusters) to (i) minimise the variability of clusters and (ii) maximise variability between clusters.
Figure 4-33. Comparison between Euclidean, D7 and D10 distance functions. These distance functions overlap each other. Only for three queries they provide an independent answer.

For CB2, the same cluster analysis was performed. Figure 4-34, Figure 4-35 and Figure 4-36 show the clusters or groups of distance functions given by a \( k \)-means cluster analysis seeking two and three clusters, respectively. In comparison with the results obtained previously for CB2 with equal feature weights, similar clusters of distances can be distinguished. For the 2–means clustering the distances are grouped into Euclidean, City block, Chebychev, D8 and D9 (Figure 4-20 a), and Divergence, Canberra, D7, and D10. As noted in the unweighed case retrieval (section 4.5.1) the similarity within clusters is not the same for all queries. Using an analysis of variance the discrimination between clusters is good only on a few occasions i.e. high F values (e.g. Q_4, 9, 25, 32). For query 25, the test statistic F is 7.00, which is higher than the \( F_{\text{crit}} \) value (5.143 at 5% significance level), indicating a good cluster discrimination for this query. The means of the clusters (Figure 1-26 e) are not well differentiated for most queries, indicating a poor discrimination between clusters. In summary, it can be concluded that by choosing either Euclidean or Chebychev distances, the retrieved case will be the same. The same can be said for the group conformed by D8, D9 or City block distances, and for the group conformed by D7, D10, Canberra and Divergence.
**Moment Feature Vector Matching**

- **Figure 4-34.** $k$-means cluster analysis with $k=2$. CB2

- **Cluster 1**
- **Cluster 2**

**Members of each cluster and Distances from Respective Cluster Center**

<table>
<thead>
<tr>
<th>Cluster 1 contains 5 variables</th>
<th>Cluster 2 contains 4 variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Variable</td>
</tr>
<tr>
<td>EUCLIDEA</td>
<td>CITY</td>
</tr>
<tr>
<td>4.43</td>
<td>4.22</td>
</tr>
<tr>
<td>CANBERRA</td>
<td>DIVERGEN</td>
</tr>
<tr>
<td>2.37</td>
<td>5.48</td>
</tr>
</tbody>
</table>

**Figure 4-35.** Members of each cluster and distances from respective cluster centre
Figure 4-36. k-means cluster analysis with $k=3$ for the distance functions used in retrieval of cases from CB2 when using expert feature weights.

### 4.5.2.2 Global retrieval

Figure 4-37 and Figure 4-38 show the results, for the best *image-case* match for various image-queries using low ($p \geq 3$) and high ($p > 3$) order moments, for both CB1 and CB2 respectively. Similarly to the analysis using equal feature weights (section 4.5.1.2) it can be observed from these figures that depending on the moment’s type (geometric, central or Legendre) and moment’s order (low or high) different retrieved cases are obtained. However, there is one exception. Figure 4-37 b) and Figure 4-38 b) show that central moments of any...
order retrieve the same case for all queries for CB1 and for most queries (except for four instances i.e. Q_6, 8, 16,18) for CB2.

Additionally, low order geometric moments converge to the same case as low order Legendre moments and central moments of any order (Figure 4-39 and Figure 4-40). Correspondingly, their respective high order moments also converge to a same case. A k-means cluster analysis using two clusters confirmed these results (see Figure 4-41 and Figure 4-42).
Figure 4-37. Retrieved case for 18 queries by using low and high order moments when using expert feature weights a) Geometric b) Central c) Legendre moments.

Figure 4-38. Retrieved case for 34 queries by using low and high order moments when using expert feature weights a) Geometric b) Central c) Legendre moments.
Figure 4-39. Comparison of low and high order moments for retrieval in CB1 when using expert feature weights.

Figure 4-40. Comparison of low and high order moments for retrieval in CB2 when using expert feature weights.
The greater the F value the better the instance discriminates between clusters.

a) Analysis of Variance

Figure 4-41. k-means cluster analysis with k=2 for the moment-based retrieval functions using expert feature weights for CB1.

b) Clusters and Members

The greater the F value the better the query instance discriminates between clusters.

Members of Cluster Number 1
and Distances from Respective Cluster Center
Cluster contains 4 variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Variable</th>
<th>Variable</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEOM_LOW</td>
<td>CENT_LOW</td>
<td>CENT_HIG</td>
<td>LEG_LOW</td>
</tr>
<tr>
<td>Distance</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Members of Cluster Number 2
and Distances from Respective Cluster Center
Cluster contains 2 variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEOM_HIG</td>
<td>LEG_HIG</td>
</tr>
<tr>
<td>Distance</td>
<td>0.263523132 0.263523132</td>
</tr>
</tbody>
</table>

Figure 4-41. k-means cluster analysis with k=2 for the moment-based retrieval functions using expert feature weights for CB1.
The greater the F value the better the instance discriminates between clusters.

a) Analysis of Variance

Figure 4-42. k-means cluster analysis with k=2 for the moment-based retrieval functions using expert feature weights for CB2

The cluster analysis (Figure 4-41 b) shows the same clustering as in the case analysed in section 4.5.1.2 for queries using equal feature weights i.e. the first cluster contains two members or variables: high order geometric and Legendre moments while the second comprises four variables: low order geometric, central and Legendre moments and high order central moments (Figure 4-41 b). The distances of each variable from the respective cluster centre (mean) are small indicating that the variables are not very distant from their cluster centre. Most instances (i.e. queries) differentiate well between clusters as evidenced by the high F values in Figure 4-41 a) (i.e. most F values tend to infinity as most "Within SS" are zero refer to eq. (4-28) p. 115) and by the variation of the cluster's means shown in Figure 4-42 c) an identical pattern can be seen in Figure 4-42.

The results of the k-means cluster analysis discussed in the former sections are summarised in the following tables:
Table 4-14. Summary of the results of the cluster analysis performed on the standard distance functions for CB1 and CB2.

For both case-bases using either equal or expert feature weights similar clusters are formed. However, only few instances discriminate significantly well between clusters. This means that there is not difference in by using either any standard distance function (e.g. Euclidean, Chebychev) the local retrieval of PU cases when they are represented as numerical attribute-value pairs.

<table>
<thead>
<tr>
<th>Case-base</th>
<th>Standard distance measures</th>
<th>Discrimination between clusters</th>
<th>Main conclusions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cluster 1</td>
<td>Cluster 2</td>
<td>Cluster 3</td>
</tr>
<tr>
<td>CB1_equal</td>
<td>Euclidean</td>
<td>City</td>
<td>Canberra</td>
</tr>
<tr>
<td>weights</td>
<td>Chebychev</td>
<td>D8</td>
<td>Divergence</td>
</tr>
<tr>
<td></td>
<td>D9</td>
<td>D7</td>
<td>D10</td>
</tr>
<tr>
<td>CB1_expert</td>
<td>Euclidean</td>
<td>Canberra</td>
<td>D10</td>
</tr>
<tr>
<td>weights</td>
<td>Chebychev</td>
<td>City</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>D7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>D8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>D9</td>
<td></td>
</tr>
<tr>
<td>CB2_equal</td>
<td>Euclidean</td>
<td>City</td>
<td>Canberra</td>
</tr>
<tr>
<td>weights</td>
<td>Chebychev</td>
<td>D10</td>
<td>Divergence</td>
</tr>
<tr>
<td></td>
<td></td>
<td>D7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>D8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>D9</td>
<td></td>
</tr>
<tr>
<td>CB2_expert</td>
<td>Euclidean</td>
<td>City</td>
<td>Canberra</td>
</tr>
<tr>
<td>weights</td>
<td>Chebychev</td>
<td>D8</td>
<td>Divergence</td>
</tr>
<tr>
<td></td>
<td></td>
<td>D9</td>
<td>D7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>D10</td>
</tr>
</tbody>
</table>
Table 4-15. Summary of the results of the cluster analysis performed on moment-based retrieval functions for CB1 and CB2.

For both cases using either equal or expert feature weights moment-based retrieval functions are clustered into two groups. The first cluster consists of low order geometric, Legendre, and central moments of any order. The second cluster consists of high order geometric and Legendre moments. Each cluster retrieves different cases.

<table>
<thead>
<tr>
<th>Case-base</th>
<th>Moment-based</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th># instances with high F 0.05 value</th>
<th>Discrimination between clusters</th>
<th>Main conclusions</th>
</tr>
</thead>
<tbody>
<tr>
<td>CB1</td>
<td>Low order geometric</td>
<td>High order geometric</td>
<td>16/18</td>
<td>There is enough statistically significant evidence to suppose that the cluster formation of moment functions is valid. Therefore it is concluded that high order geometric and central moments of any order.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Low order Legendre</td>
<td>High order Legendre</td>
<td>ditto</td>
<td>ditto 29/34</td>
<td>ditto 18/18</td>
<td>ditto 27/34</td>
</tr>
<tr>
<td></td>
<td>Low order central</td>
<td>High order central</td>
<td>ditto</td>
<td>ditto 29/34</td>
<td>ditto 18/18</td>
<td>ditto 27/34</td>
</tr>
</tbody>
</table>

4.5.3 Comparison of Local and Global Retrieval

Comparison between local and global retrieval algorithms using either equal or expert feature weights for the two case-bases is detailed in this section. It was concluded in the former section that most distance functions retrieved the same case for either CB1 or CB2 under equal or different feature weights. From Table 4-5 to Table 4-12 (p.119) similarity between the retrieved case results by local (standard distances) and global (moments) is apparent for some types of moments. In particular, the Euclidean distance and low order geometric, Legendre and central moments appear to retrieve the same case for most queries. Figure 4-43 and Figure 4-44 show that for most queries for both CB the low order geometric, Legendre and central moments of any order (LowGLC for short) retrieve the same query as the Euclidean distance.
Figure 4.43. Comparison of distance functions used in local and global retrieval for CB1.

**EUC MIN** = Euclidean distance

**GL HIGH** = High order geometric and Legendre moments

**Low GLC** = Low order Geometric and Legendre moments and central moments of any order

It can be observed that the Euclidean distance and low order geometric, Legendre and central moments of any order retrieve the same case for most queries. Specifically when using equal feature weights (a), they coincide for all the queries (except Q4). When expert feature weights are used, they coincide for most queries except for Q1, 4, and 6.
Figure 4-44 Comparison of distance functions used in local and global retrieval for CB2.

EUC MIN = Euclidean distance
GL HIGH = High order geometric and Legendre moments
Low GLC = Low order Geometric and Legendre moments and central moments of any order

The same tendency as the one found in CB1 can be observed here. This being that the Euclidean distance and low order geometric, Legendre and central moments of any order retrieve the same case for most queries. For the case in which equal feature weights are used (Figure a), local and global retrieval functions coincide for the all the queries (except Q4, 16, 31 and 32). When expert feature weights are used, they coincide for lesser queries (i.e. Q2, 3, 5, 8, 11, 14, 16, 18, 19, 23, 25 and 32).

From the previous discussion, it can be concluded that:

- The Euclidean distance has a moment counterpart, which corresponds to either low order geometric or Legendre moments or central moments of any order i.e. they are equivalent. This means that for continues numerical attributes either the Euclidean distance or the low order geometric, Legendre or central moments of any order retrieve the same case for a given query.

- Although the computation of high order moments is more intensive than in comparison to when the Euclidean distance is computed, the use of high order moments for retrieval of cases described as images offers an alternative result. This result, however, requires experimental validation to assess whether the higher moments choose cases that are actually closer to a given query in a determined domain. It is expected that, because high order moments approximate the image function, f(x), more accurately
than low order moments, retrieval using high order moments will be better because it uses more accurate case descriptions.

The work presented in this thesis, has data limitations as it was previously stated. An alternative way to determine which distance measure is more advantageous in a given situation is by assessing if the problem and solution spaces are linked. If they are, it is expected that a distance function would retrieve the same case using either space, and hence it would be probabilistically better than other measures that result in different cases in both spaces. However, initial experiments carried out by the author showed that the space of mechanical properties is different to the space of formulation values for PU foams (see Figure 4-45). This result does allow investigating further if the spaces are linked by a distance measure that retrieves the same cases for both.

![Figure 4-45. Schematically representation of retrieval results using both mechanical properties and formulation data sets. In this figure a distance function that retrieved the same cases for a given probe in both spaces is shown in a) and conversely in b).](image)

In real practice, users need friendly decision support systems that present to them options with a measure of confidence associated to it. Apart from the graphical information provided by the moments (see Figure 4-28), a set of measures with associated probabilities can be easily calculated by taking count of the retrieved cases by the different distance measures (see Figure 4-46). By doing this, the calculated standard distance measures can be used in combination with the moment-based measures to generate an index of confidence when a case is chosen from a set of retrieved cases. For example, for the query 1 for CB2, various cases are retrieved by the distance measure (see also Table 4-9 and Table 4-10 p. 16121). However, case No. 6 is the most popular case retrieved between all distance functions and hence it is reasonable to suppose that this case has a major probability of being the closest case to the query. The next popular case is case No. 28. This assumption however, requires further experimental validation for different domains in which can be used.
4.6 Summary and Conclusions

During the last years, similarity assessment has been one of the most active research topics in CBR [139, 140, 190-192]. Several domain specific similarity measures have been proposed based on research work done in other areas such as information retrieval, and classification [125, 130]. Classification tasks have greatly influenced CBR retrieval, and hence retrieval has been based upon labelling cases and classifying queries. However, more studies need to be conducted to evaluate retrieval in alternatives ways (e.g. such as the shape based approach presented in this thesis) rather than as a classification task. In particular, the retrieval of cases from a case-base has been presented in this thesis as the task of matching images that can be used to code the information contained in a case. A “gene-like” representation, in analogy to DNA molecules, was used for efficient graphical representation of case bases. Matching of images was then computed by using the image’s moment features, in which the geometric characteristics of the images are examined for similarity. This alternative approach to retrieval is not only of interest as a retrieval algorithm but also as a way to support user decision through visualisation.

The ultimate objective of this chapter was to solve the research question three. The methodology employed to address this research question involved splitting it in two sub questions (A, and B), which has resulted in the study and comparison between the classical CBR retrieval approach, referred to in this thesis as local retrieval, and a retrieval based on
feature moments, which has been referred to as global retrieval. Local retrieval based on a nearest neighbour algorithm that uses quantitative distance functions to match cases, is an approach that computes dissimilarity feature by feature instead of comparing the cases as a whole. Even though its computation is very simple, it presents other problems such as it is not user friendly and it is not clear what distance function is ought to be used and why, and what are the implications of this choice. Conversely, a global approach as its name suggest, allows a global comparison to be made by using moment features of cases represented as images. As in image recognition problems, the whole geometric characteristics of an image are represented by its moment's features so that a retrieved case will be, by global shape, the closest case. Because the feature relevance is an important parameter in retrieval, this issue was studied for both local and global approaches.

The following sub sections provide the main conclusions.

4.6.1 Local Retrieval

Current local distance functions are readily computed. They are used to compute the similarity of cases with numerical attributes. The findings from this study indicate that the majority of the quantitative distance functions used in the nearest neighbour rule retrieve the same case. There is no differentiation between them and hence it is not clear which is the most appropriate function to use in different situations.

Using clustering analysis, a range of distance measures were grouped into two or three clusters to determine if there was any discrimination between them. It was found that all distance measures used in the nearest neighbour algorithm for local retrieval for the case studies used in this project retrieve within a 5% significance level the same case for a given query. This situation is independent of whether equal feature weights or expert determined feature weights are included. When using equal feature weights on CB1, for 55% of the time the distances were well discriminated, i.e. there was a statistically significant difference between the clusters they formed. For the CB2, this rate went down to 14.7%. For the case with different feature weights, these percentages were 61% and 29.4% for the CB1 and CB2 respectively. These percentages are low, and are a statistical indication that there is not much significance difference between the distance measures in their retrieval performances. For this reason, it was concluded that by using a representative distance function for the case studies in this project was possible and the Euclidean distance (the most widely used distance function in the nearest neighbour algorithm) was adopted for comparison with global retrieval performance.
4.6.2 Global Retrieval

The matching of cases by using image functions to enable global retrieval analysis to be undertaken was performed using three different types of moments, geometric, central and Legendre moments of low \((p \leq 3)\) and high order \((p > 3)\). It was found that the moment functions are clustered into two groups. The first cluster consists of low order geometric, Legendre, and central moments of any order. The second cluster consists of high order geometric and Legendre moments. Each cluster retrieves different cases. High order geometric and Legendre moments retrieved the same case for almost all queries for CB1. Specifically 89% of the queries (with equal features weights) and 100% (with expert features weights) discriminated well between the two moment clusters, which means that the cluster’s answers were significantly different from one another. The same tendency occurred for the second case-base CB2 with instances discriminating well \(85\%\) (equal features weights) and \(79.4\%\) (using expert features weights) of the time.

Correspondingly, low order geometric and Legendre moments converged to the same case using either equal or different feature weights on both case-bases CB1 and CB2. Additionally, central moments of any order converged to the same case for most queries. This means that for the case of central moments, only low order central moments are required to describe fully the image. This can be explained by the fact that central moments approximate the function \(f(x)\) about the centroid of the image (see for example Figure 4-11). As the moment’s order increases, the set of central moments will eventually approximate \(f(x)\) describing the shape of the image. At it was found, only moments up to order three were required to describe the cases described as genes.

Furthermore, it was found that the Euclidean distance and low order geometric, Legendre and central moments of any order retrieved the same case for either CB using any set of weight features. Central moments of any order retrieve the same cases as the Euclidean standard function as well. With this, it is concluded that low order geometric and Legendre moments act as the Euclidean standard function for retrieval of cases represented as feature vectors. Their corresponding high order moments offer an alternative retrieval. However, it cannot be assessed in the present work which option performs better when compared with a probe or test case.

The answers to the research sub-questions of this chapter are:

**Sub-question A. What is the performance of the current local similarity measures based on standard metric distances? What are their advantages and disadvantages?**

The assessment of retrieval by using standard distance functions is computationally simple. However, it is not clear what distance function is ought to be used and why and the
Implications of that choice. This study found that the most common distance functions used in the CBR literature, i.e., Euclidean, City block, Canberra, Divergence, D7, D8, D9, and D10 (refer to Table 4-4) retrieve with a high probability the same query when only numerical positive attributes are used. Although their individual performance (i.e., how accurate the retrieval is when each distance is employed) was not assessed in this study, it is clear that the main disadvantage of these common distance functions is that their answer, i.e., a number represents the similarity between two cases, which cannot be interpreted further.

**Sub-question B.** What global similarity measures do exist that can be applied for the retrieval of PU cases? What are the basic assumptions for using these approaches? How can these similarity measures be used to assess the similarity between PU formulations? What are their advantages and disadvantages? Do they perform better than current local measures based on standard metric distances? If so, how is retrieval enhanced?

In terms of this work, global similarity connotes the matching of a case as a whole (contrary to local retrieval which is a scalar weighted sum of attributes). Shape matching as a global similarity measure between cases within the CBR framework was the approach explored in this thesis. For using this approach, formulations can be represented as two-dimensional images and matching is then performed by using moment functions in a similar fashion to the approaches studied in image analysis in pattern recognition. This is, by calculating a set of moments a case-image is characterised and can be compared to others at different levels of accuracy. Because the two-dimensional representation of a case does not require the use of complex moments that compensate, for example, for problems associated with noise when capturing an image, a set of geometrical, central and Legendre can be used. Matching is performed by first calculating the moments of the case \( m^C_p \) and the query \( m^Q_p \) based on an image function definition for the case \( f^C(x) \) and the query \( f^Q(x) \). Then, the minimum Euclidean distance between the moments in the case-base and the query \( (m^C_i - m^Q_i) \) is assessed to retrieve the closest case to the query (i.e., smaller distance).

Although the computation of moments is mathematically more intensive than in comparison to when standard distance functions are used, the use of moments for retrieval offers various advantages. For example, different moment orders can be computed offering various levels of detail in the assessment of similarity. The higher the order, the higher the level of detail of the global shape characteristics of the image and therefore, if two PU formulations are represented as images, they can be similar in at least two ways, namely, on a fine basis (when high order moments are used) or at a gross degree (when low order moments are used). Additionally, the problem of assigning weights to features was addressed in this
thesis, so different expert weights can be also taken into account when matching cases using moments.

Research Question 2. Is a global similarity measure more effective than local similarity measures based on standard distances functions for the formulation of PU foams?

The global similarity measure based on moment descriptors presented in this chapter allows:

1. To quantify the extent to which a retrieved case is a useful solution to a query by the provision of a quantitative indication of this measure. This result is two fold as two degrees of similarities can be obtained, which refer to whether low or high order moments are used. However, the meaning of the results offered by high order moments remains unknown. Theoretically, high order moments contain more information about an image and therefore, it is intuitively assumed that they would provide a closest case when an image-query is presented. This hypothesis could not be evaluated experimentally because the data could not be gathered, but it can be validated if future studies are carried out.

2. To assess similarity in cases differing in the number of attributes to be compared for similarity by simply giving a weight of zero to the attributes that are not present in a query.

The effectiveness of the global in comparison with the local one, however, could not be assessed. This was due to the fact that there were no probes or test cases i.e. cases with the “correct” formulation to assess retrieval in terms of which algorithm performs better. For this reason the retrieval methods could not be compared to each other in terms of which one gives a better result (i.e. which is more accurate when compared to a test case) but in terms of how close or different their answers are. By means of a cluster analysis a comparison between distance measures used for each retrieval algorithm (i.e. local using standard distance functions and global using moment functions) was performed. It was found that low order geometric and Legendre moments and central moments of any order retrieve the same case as the Euclidean distance for both of the data sets used. This means that the Euclidean distance is analogue to a low order moment function that represents gross level image features. High order geometric and Legendre moments while enabling finer details about an image to be represented retrieved different cases than their low order counterparts did and did not find any standard distance function counterpart.

It is suggested, that there is not single distinguished approach to similarity in CBR. Rather, CBR systems should allow the integration of different approaches to similarity and the selection of different concepts.
Chapter 5

CBR ADAPTATION

Feasibility of an Artificial Neural Network Approach for Solving the CBR Adaptation Problem in Polyurethane Formulations

Abstract: CBR adaptation in the design domain is a knowledge-intensive task that has impeded the development of complete CBR systems due to the adaptation knowledge acquisition bottleneck. However, inductive techniques can ease this problem by eliciting useful knowledge from the case-base. The aim of this chapter is to determine if a neural network can learn a desired mapping from experimental PU formulations in order to prove (i) that the case-base has useful implicit knowledge that can be exploited by a neural network and (ii) that the knowledge can be used to adapt retrieved cases. A two-hidden layer feed-forward neural network was trained to map case feature differences to PU formulation properties using the back-propagation algorithm. Results indicated that a 16000 epochs trained network is capable of learning the desired mapping enabling to (i) reduce the adaptation knowledge acquisition effort and (ii) allow the adaptation of formulation ingredients. Although a small data set was used (34 initial formulations) the network error was only 4%.

Key words: CBR Adaptation, Artificial Neural Networks (ANN)

5. CHAPTER 5. CBR ADAPTATION

5.1 Introduction

During recent years there has been a need to develop complete CBR systems in design domains i.e. systems that do not only serve as retrieval systems such as those found in help-desk applications but also serve as problem solvers. Progress has been made to the point that small CBR prototypes are available for various design tasks. Some of these still rely on a human-based approach to adaptation in which a domain expert proposes the changes that are required to adapt the retrieved case to solve a problem.

Most researchers seem to agree that adaptation is one of the challenges in the development of complete performing CBR systems. Börner [193] suggested that adaptation becomes essential especially in domains such as design because solutions are never identical to past solutions and frequently two or more previous solutions must be combined to solve a new problem. A recent review by Wilke and Bergmann [194] suggests that the adaptation stage is
still considered to be the least developed in the implementation of CBR systems. The findings of studies examining approaches to adaptation suggest that adaptation is a difficult task largely because of the complexity in eliciting or acquiring adaptation knowledge and further compounded by the fact that this task requires a deep understanding (at expert level) of the problem domain [119, 159].

Recent studies by Hanney and Keane [164] and Craw et al. [159] indicate that the adaptation acquisition problem can be reduced by using the knowledge that may be already present in the case-base. Like inductive learning programs, artificial neural networks can learn domain knowledge from examples. However, the neural network approach has received little attention as a tool for CBR adaptation and more studies need to be conducted to study how neural networks can induce knowledge from a case-base to reduce the adaptation knowledge acquisition task and perform adaptation of cases.

The purpose of this chapter is (i) to study the likelihood of a neural network approach to induce knowledge from a case-base to adapt cases and (ii) to discuss various issues regarding neural system design, implementation and operation in CBR systems for adaptation of cases in formulation applications.

5.2 Problem Statement

Within the CBR framework, adaptation stands for the modification of a retrieved case that partially matches a user-query to solve a proposed problem. In particular, for a PU formulation application, adaptation is expressed in terms of how to modify a retrieved formulation. This encompasses the identification of both the formulation ingredients and their quantities that meet a set of performance constraints.

As reviewed previously in section 2.3.2 (p. 23) the formulator draws upon the composition-structure-manufacturing knowledge triad to seek for appropriate relationships so that changes in either the materials or processing conditions can be proposed to achieve desired end-properties i.e. meet constraints. This adaptation task is a trial and error process of formulation guided by the formulator’s experience, which is instinctive in nature. This “adaptation knowledge” is difficult to elicit, structure, and deploy. Even though it could be made explicit, its acquisition is a hard task. Particularly, for the PU domain several limitations restrain effective knowledge elicitation from PU experts, including:

- Experts are normally reticent to reveal their expertise. This might be due to two possible reasons including, first, that experts are restrained by the companies they work for to publish their findings in the open literature, as evidenced for the lack of detail when reporting the methodology and chemicals used (e.g. [60]); and second, that
experts take several years to gain an understanding of their field so they naturally feel reticent to disclose their experience to beginners.

- Even if there is a willingness to formalise experts’ knowledge (heuristic knowledge\(^{14}\)), for example, for knowledge engineering tasks or for apprenticeships, this process normally can take a long time. In addition, valuable knowledge can be lost because experts tend to convey their knowledge with fuzzy terms and hence knowledge might often be transmitted imprecisely.

- In most companies the formulation of PUs has developed into an art rather than a science. This has been mainly due to commercial secrecy, compounded by the lack of a systematic approach to discover what formulations work and why for each PU application. This greatly hinders knowledge engineering tasks because it prevents the easy formalisation of PU expertise.

The problem that this chapter aims to solve can be stated as:

*Given (i) the user requirements for a PU formulation, (ii) retrieved cases that partially match a required PU formulation and (iii) no other domain knowledge than the case-base, the goal is to adjust a retrieved formulation to find the ingredients and their quantities that result in a PU formulation that approximate the user requirements.*

## 5.3 Objectives and Research Questions

The objective of this chapter is to study the feasibility of using a neural network to adapt retrieved cases in the context of CBR (Figure 5-1) so that the problem formerly stated can be solved. This objective can be satisfied if the following research question is answered:

### 5.3.1 Research Question 3

_How can the problem of computer-assisted formulation of PUs be fully solved? In terms of the CBR framework this translates into: How can a retrieved PU case be adapted so that a solution to a formulation problem is obtained?_

This research question can be best answered if divided into three sub-questions:

**Sub-question A.** *What approaches have been used to CBR adaptation using no other domain knowledge than the case-base? What are the advantages and disadvantages of these approaches?*

**Sub-question B.** *Can the artificial neural network approach be used to elicit knowledge from the case-base and use it to guide adaptation of cases? If so, what kind of artificial neural

---

\(^{14}\) In this work, heuristic knowledge has been used in the context of expert knowledge that is loosely defined and mainly developed by trial and error.
network can be used and why? What are the basic assumptions for using this technique; what are its limitations?

**Sub-question C.** How can an artificial neural network be implemented to adapt retrieved PU cases when these are represented as feature vectors? How can the knowledge learnt by the artificial neural network be used to adapt retrieved cases? How reliable is the adapted case found by the network?

Figure 5-1. Research question for the chapter 5.
In view of the difficulty to acquire knowledge to support adaptation tasks, most current CBR systems do not perform adaptation. In this thesis, it is hypothesised that the PU formulation case-base contains tacit knowledge and that a neural network can be used to elicit that knowledge and to adapt cases. Note: The graph above is inspired by the work of [153].

### 5.4 Methodology

Sub-question A is answered by reviewing existing approaches to adaptation that use knowledge from the case-base to adapt cases (section 5.5).

Sub-question B is answered through a discussion about the use of the artificial neural network (ANN) approach as an inductive technique that has potential to capitalise on
imprecise incomplete knowledge to solve problems in complex domains. As there is a vast amount of research and writing, which is relevant to ANNs, the discussion mainly emphasises on: (i) How and why could artificial neural networks deal with the limitations stated in section 5.2; (ii) How can ANNs be applied to understand complex relationships between input and output data (e.g. retrieved PU case-problems i.e. mechanical properties and PU ingredients).

Sub-question C is answered through
1. An implementation of an ANN to adapt retrieved PU cases (section 5.7)
2. A method to use the knowledge learnt by an ANN to adapt PU cases retrieved using both local and global retrieval algorithms as defined in chapter 4 (section 5.8)
3. A critical assessment of the ANN technique within the CBR-adaptation framework (section 5.9).

Answer to the research question three is presented in the last section of this chapter.

5.5 Adaptation Using the Case-Base: Related Work

The aspects of how to extract knowledge that might already exist in the case-base to adapt cases represented as feature vectors using an ANN approach are covered in this section, as this is the main objective of this chapter. First, the problem of adaptation of cases represented as feature vectors is reviewed, followed by the motivations for the use of an ANN approach to solve this problem. Classical reviews and introductions on other adaptation strategies are available elsewhere [28, 194, 195]. Information about formal adaptation frameworks not discussed in this thesis can also be obtained from a number of other sources such as the papers by Bergmann and colleagues [194, 196], and the surveys by Smyth and co-workers in [113, 164, 197, 198].

Different case representations require different adaptation strategies. For cases that are represented as attribute-value vectors, adaptation involves to replace a value(s) corresponding to the solution in the retrieved case with a value that can fit the desired solution (Figure 5-2).
### CBR Adaptation

<table>
<thead>
<tr>
<th>Task</th>
<th>Result</th>
</tr>
</thead>
</table>
| **1. Retrieval**  
 a) Match problem features between a query and the cases in the case-base  
 b) Retrieve j closest cases | $c_1(r) = \{c_{p1}(r), c_{p2}(r), c_{p3}(r), \ldots, c_p(r)\}$  
 $\vdots$  
 $c_j(r) = \{c_{p1}(r), c_{p2}(r), c_{p3}(r), \ldots, c_p(r)\}$  
 $\{c_{s1}(r), c_{s2}(r), c_{s3}(r), \ldots, c_s(r)\}$ |
| **2. Adapt retrieved case**  
 a) Adjust solution features  
 b) Propose a solution | $c(a) = \{c_{p1}(a), c_{p2}(a), c_{p3}(a), \ldots, c_p(a)\}$  
 $\{c_{s1}(a), c_{s2}(a), c_{s3}(a), \ldots, c_s(a)\}$ |

**Figure 5-2. Adaptation of cases represented as feature vectors**

The goal is to find the solution feature values for a given query $q$. A feature case vector $c$ is composed of both a case-problem ($c_p$) and a case-solution ($c_s$). The case-problem set contains the features that describe a PU problem including the mechanical properties for a given PU foam. The case-solution is the set of features that describe a PU solution i.e. the formulation ingredients for a given foam. In retrieval, $q$ is matched against cases stored in the case-base, so that the closest match is retrieved, i.e. $c(r)$. Its solution set is then adapted so that a solution to $q$ can be found.

Knowledge adaptation in domains such as design is usually performed by an expert because this task requires a deep understanding (at expert level) of the problem domain. Usually, this adaptation knowledge is difficult to elicit or acquire to support CBR systems. The main reasons behind difficult adaptation knowledge elicitation are related to the intrinsic complexity of design tasks (as it was explained in chapter 2) and the difficulty to express experts' knowledge i.e. intuition and experience into symbolic form [199].

In order to overcome the knowledge engineering problems that arise in the acquisition of adaptation knowledge while providing complete CBR systems several researchers have recently proposed the use of inductive techniques to ease the adaptation knowledge bottleneck by learning from cases [161, 164]. Some of these ideas are related to data-mining too; see for instance, the fusion of traditional knowledge-elicitation (knowledge elicited from human experts) and data-mining (knowledge extracted from data) in the construction of knowledge-based systems in manufacturing domains [200].

In the development of complete CBR systems, Hanney and Keane [164] have reported on the use of learning adaptation rules from cases represented as nominal feature vectors. The antecedent part of an adaptation rule consisted of the feature differences between each pair of cases, with the differences between the solutions as the consequent part. Adaptation is carried out by applying these rules to deal with feature differences that are found between a query and a retrieved case.

In particular for the formulation domain, Jarmulak et al. [161] recently proposed a "knowledge light" approach for learning adaptation knowledge from the cases in the case-base. Similarly to the approach used by Hanney and Keane [164], they constructed adaptation cases by storing differences between the problem features. For numerical attributes these differences correspond to feature differences between a probe case and the corresponding retrieved case, and the difference between their solutions (refer to Figure 5-3). This adaptation
approach, deals with feature differences that are found between a query and a retrieved case, by using single or combined suggested corrections.

![Diagram](image)

Figure 5-3. Constructing adaptation cases. Source: [161]

### 5.6 Artificial Neural Network Applications within the Formulation Domain

Artificial Neural Networks research is a topic in very much flux. The aim of this section, therefore, is to review how PU formulations can be utilised by an ANN that enables generation of adaptation knowledge to develop a complete functional CBR system in an effective manner i.e. by using only the experiential knowledge available (i.e. PU formulations from the case-base). Discussion about the application of ANNs in the formulation domain is given first. The ANN models discussed in this chapter that have been used in these applications are limited to those used for function approximation in problems which are likely to be influenced by (i) imprecision and noise in the data and (ii) a difficulty in the development of analytical models that can study complex relationships between an extensive amount of variables.

VerDuin [201] is one of the first researchers that have pointed out the advantages of neural computing to assist complex tasks in formulated products. Like him, several researchers have evaluated the use mainly of ANNs, to address the complexity of the formulation domain due to the large number of interacting variables and the lack of analytical models of formulating problems such as the formulation of pharmaceuticals [184-187], rubbers [1] and pigments [178-182]. The ability of ANNs to learn from partial knowledge about the manufacturing process has been an important feature for its implementation.

Applications of ANNs within general manufacturing and product formulations domain have increased during the last decade in particular. Example target applications are referred to and links to the literature are given in Figure 5-4.
Figure 5-4. Applications of neural networks in product formulation.
In each of these domains ANNs have enabled the reduction of the complexity of the product formulation process by generating a mapping from measurable inputs (e.g. formulation ingredients and processing conditions) to desired outputs (e.g. final product properties). The following examples show the major advantages that have been claimed as a result of the application of ANNs in formulation development.

**Adhesives**

Artificial Neural networks have been used for the design of water dispersible granules [15] and aqueous suspension concentrates [16]. Both ANN models proved effective in designing improved adhesive formulations.

**Dyes and Pigments**

Initial experiments for predicting colour of dye mixtures for fibres were carried out by Bishop et al. [18, 19]. The implemented ANN proved successful leading to further research and application of ANN in pigments [20] and fluorescent dyes [21, 22].

**Rubber Compounding**

Problems of quantitative composition property were studied by using a back-propagation ANN [1]. This study showed that the ANN approach is suited to model quantitative nonlinear relations between ingredients and product properties.

**Pharmaceuticals**

Applications in pharmaceuticals have been reported for controlled release tablets [11-14, 202] to model in vitro release characteristics of a number of drugs. Initial studies by Hussain [202] found that a single hidden layer was able to predict drug-release properties. Later studies supported these findings [11-14].

Researchers at Procter and Gamble pharmaceuticals and Purdue University used an ANN to model a caffeine table formulation to relate processing and formulation parameters to tablet properties. Their research was compared to conventional statistical methods proving that the former performed better.

More recently Rowe and Woolar [203] have evaluated the impact of the neuro-fuzzy approach to model film coatings that are applied to tablets to protect the active drug against environmental conditions or control its release. Similar studies by Khan [204] using ANN have also led to an improvement in tablet formulations.

Although these studies have highlighted major advantages of the application of ANN models, there has been little discussion of the rationale behind their implementation i.e. data pre-treatment, network design and training methods used. This has been due to the use of commercial software for ANN implementation. ANN software that has been used in product
formulation studies includes windows-based software packages such as the STATISTICA ANN toolbox, CAD/Chem (AI Ware Inc. Cleveland, Ohio), and the Trajan Neural Network Simulator (Washington, Tyne and Wear UK).

In the application of ANN to product formulation the multi-layer perceptron (MLP) with sigmoidal nonlinearities has been the most common neural network used. They are trained in a supervised manner with the error back-propagation algorithm. Although other network architectures are available, most researchers have implemented MLP to map a desired set of variables using experimental formulation data for several reasons. First of all, product formulation problems are characterised by a large number of variables that interact in intricate ways not always known. MLP have demonstrated the ability to learn from examples to map formulation ingredients and formulation properties to improve the formulation process as it was described formerly. In addition, it is well known in the neural computing field that a two-layer (i.e. one hidden layer) feed-forward network can approximate any non-linear continuous function to an arbitrary degree of accuracy if the hidden layer contains sufficient nodes [205]. This assumes that the necessary data are available and that the network can be trained until a global minimum is reached.

**Polyurethane Formulation**

In chapter 4, case retrieval was studied by using two approaches, namely, local retrieval based on standard distance functions, and global retrieval based on the use of image moment descriptors. Adaptation of a retrieved case by either local or global algorithm involves the replacement of feature values that correspond to the chemical formulation of a PU foam. We have referred to these features as case-solution vectors. With no other domain knowledge available but the case-base, an ANN that can learn from the case-base is proposed here. The network is intended to generate a mapping so that the retrieved chemical formulation can be adjusted using this knowledge to meet desired properties. The process of using such network as a knowledge-base to assist CBR adaptation is shown in Figure 5-5.
According to Figure 5-5, a trained ANN that has generated a mapping between case-problem feature differences (i.e. delta properties) and case-solution feature differences (i.e. delta ingredients) is used to adjust the discrepancies in the formulation between a retrieved case and a given query. Specifically, what it is desired from the network is to learn how changes between pairs of property features affect formulation ingredient features. So that when a retrieved case needs to be adapted because it is not a 100% match to the query, the trained ANN can propose a better formulation.

For the purposes of this research, the goal is to find an ANN that can do well enough to prove that experimental formulation data contained in the case-base can be used to guide CBR adaptation.

### 5.6.1 Multi-Layer Networks

#### 5.6.1.1 Learning and Generalisation

In the context of neural networks, learning is defined as the process by which the weights and other free parameters (e.g. bias) of a network are adapted based on a definition of a suitable error function [206]. This is done by first collecting samples to serve as exemplars. Each sample contains the inputs and the outputs (or targets) that are desired when those inputs are presented. Then, a subset of training examples is presented to the network to approximate a function between the input and output pattern. For each sample, a comparison between the actual output of the network and the target is performed. After the entire training set samples have been processed, the weights that connect the network are updated in such a way that the
network's measure of error is reduced. The assessment of how well the network has learnt to approximate the function is performed by using an independent validation set. The network with the lowest error with respect to the validation set is selected. The performance of the selected network is confirmed by measuring its performance on a third independent set of data called a test set.

When a small number of noisy data is used, networks are prone to over-fitting the data [181]. This means that while the error on the training set is small, the error when the test data is presented to the network is large (see Figure 5-6). This lack of generalising ability (i.e. the ability of interpolation and extrapolation) and high prediction of errors depends also on the network's size. A sufficiently large network (i.e. with a high number of hidden units) is able to model complex relations but at the same time it can memorise the training examples and also model the noise present in the data. On the other hand if the network’s size is very small (i.e. with an insufficient number of hidden units used), the predictions will be smooth but also biased.

![Figure 5-6. Low training error need to imply good network performance. Source: [181 p.11]](image)

In spite of the fact that an ANN requires large numbers of exemplars for learning, in practice only a limited number of samples are available. Consequently, there is an insufficient number of data to generate the training and test sub-datasets. The other problem is that in practice it is also difficult to know before hand how large the network should be for a specific application.

When data is scarce, several approaches to handling over-fitting have been proposed in the literature [206]. The most common approaches include regularisation, early stopping, cross-validation, and leave-k-out.

Without enough formulations to justify splitting the data into training and test sets, the cross-validation approach to training can be adopted. In this approach, the training set is partitioned into R distinct segments. The network is trained R times, each time using data
from R-1 of the segments and its performance is tested by evaluating the error function with the remaining segment. This procedure is schematically represented in Figure 5-7.

Figure 5-7. Cross-validation data sets.
In this figure, an ANN is trained 5 times, each time omitting a segment (training-test set), which corresponds to one of the segments shown shaded. Each trained network is then tested using the test set and the results are averaged over the 5 networks.

5.7 Implementation of an Artificial Neural Network to Guide CBR Adaptation

5.7.1 Methodology and Experimental Setup

The application of a feed-forward back-propagation ANN for adaptation of retrieved cases has three phases, namely, pre-treatment, training and testing (Figure 5-8). Phase one is the pre-treatment task, as its name suggests a pre-processing of the input pattern is performed so that the reliability of the data can be determined. Some basic procedures include the elimination of outliers, normalisation, standardisation, transformation and data reduction. Phase two is the training process, during which a set of training examples are presented to the network, and the parameters of the network are iteratively adjusted to learn input and output relationships. Phase three is a testing process, during which a known input pattern is presented to the trained network and this is required to “discover” the possible output so that a validation of the answer can be made.

---

15 This anthropomorphic language (i.e. the network “discovers” or “learns”) is common in neural network discussions. It also indicates that the neural network learning is non-algorithmic in nature (i.e. a neural network is not programmed like a computer; it is trained to learn a pattern).
Phase one is a vital task that should not be taken for granted. To utilise ANNs effectively (and in general any computation technique) to support the analysis and modelling of complex problems it is important to know as much as possible how reliable the input data is and to have a sense of robustness of the conclusions.

Phase two is usually a lengthy task. It takes many iteration steps to satisfy the required conditions (defined in the following sections). Most ANN application studies usually concentrate on this phase, where the network architecture and input variables can intentionally be changed to optimise the network's performance. Once the network's parameters are adjusted, the network is ready for testing.

The testing phase is very simple. It only requires fast calculations using the trained ANN to predict an output according to new inputs.
5.7.1.1 Pre-treatment Task

There are several procedures that can be applied to the input patterns in order (i) maximise insight into the data set, (ii) anticipate any potential problems that can affect the accuracy of the network and most importantly (iii) ease the learning of the neural network [181]. It is well known that the network’s ability to learn any nonlinear function depends heavily on the quality of the data presented to it [181]. For ANN applications, the most common input data features that are worth exploring relate to the quality and quantity of the data (i.e. is there enough relevant information for the network to learn?), and its representativeness (i.e. can the data be used to solve the problem we are trying to solve?). This question can be answered through an exploratory analysis before commencing the ANN implementation. This is mainly because one of the important features of an ANN resides in their ability to improve their performance through learning. Therefore, the network’s learning is hampered when unreliable, unrepresentative data is used.

For supervised learning, the network is required to approximate a desired input-output mapping (refer to Figure 5-9). If the network learns with the wrong inputs the resulting nonlinear function does not reflect the real relationship between the variables. It may also be possible that there is no simple mapping between the two feature sets or that there are potential conflicts between the different relations that contribute to this mapping e.g. highly correlated data, in which case several of the various techniques that have been proposed in the literature can be used. One of them is the principal component analysis (PCA). PCA has been widely used, for instance, in pattern recognition applications to find facial features, which are important for identification [207]. Generally, PCA is used in ANN with very large data sets to reduce the dimensionality of the data set i.e. eliminating those components that contribute to the least in the data set [208].
Different network models impose different constraints on how data (i.e. input and output patterns) need to be fed to the network. In order to commence the analysis with reliable data some recommendations regarding data pre-treatment are outlined in [181 ch. 16]. The most relevant aspects to be taken into consideration in this work include (i) the selection of data sets, (ii) removal of outliers and (iii) scaling of data. The selection of data sets is discussed in the following section. Large variations in the data prevent the neural network from learning the information content in the variable. Therefore, outliers are removed or a non-linear transformation is used [181 p.267]. Scaling of data becomes very important in most training algorithms. For example, if the output’s attributes are unequally scaled, those attributes with the largest variations will be favoured as they will dominate the error sum [181 p. 267]. Therefore, scaling is frequently performed on both the inputs and the outputs usually by normalising the data in an appropriate range or by standardising the data set.

5.7.1.1.1 Selection of the neural input patterns

The data sets that were used in the training and validation sets to implement the ANN as well as the normalisation that the input and output patterns received are described in this section.
The difference between the "problem-part" of two cases i.e. the delta property, is an exemplar of an input pattern for the neural network. Similarly, other researchers [161, 164] have suggested the use of these differences between the case-problem features to build adaptation cases (recall section 5.5 above on page 162).

In order to obtain a large set of input patterns for the network, delta property values were calculated using all but the probe cases of the CB. In CB2 test cases correspond to cases No. 11, 12, 22 and 25, which were randomly selected (Figure 5-10). In addition, one record from CB2 was deleted (record No. 14) as it corresponded to the central point in the design of the experiments when making the foams and therefore its formulation appeared twice in the case base. Each input pattern \( \delta p \) is obtained by taking the differences between the case-problem attributes for the cases stored in the case-base.

\[
\text{Input patterns } = \delta p = c^{(i)} - c^{(j)} = \left\{ \text{case } i_{\text{problem attribute } k} - \text{case } j_{\text{problem attribute } k} \right\}, \text{ for case } i \neq j.
\]

where \( c^{(i)} \) is the \( i^{th} \) feature vector, which contains only the problem attributes of a case i.e. the mechanical properties.

In a similar fashion, the outputs of the network are obtained by calculating the differences between the solution attributes of a case; this is between the formulation ingredients.

\[
\text{Targets Network } = \delta f = c^{(i)} - c^{(j)} = \left\{ \text{case } i_{\text{solution attribute } k} - \text{case } j_{\text{solution attribute } k} \right\}, \text{ for case } i \neq j.
\]
The data for training the ANN corresponds to a subset from the original data set. The original data set consisted of 34 PU formulations. By taking out 5 cases (the testing subset), the remaining 29 cases were used to calculate differences with each other obtaining a total of 812 exemplars. The differences corresponding to the mechanical properties were used as the network input pattern while the differences corresponding to the chemical formula were the network output pattern.

The resulting data sets are shown in Table 5-1.

Note: Data in CB1 come from experiments using two different types of isocyanates and therefore eight cases correspond to the formulations prepared with one isocyanate and the other eight with the other. These two different isocyanates lead to mechanical properties in the foam and therefore partition the data into two groups (refer to Appendix I). For this reason, each subset of relevant data is too small (i.e. only eight formulations) to train a network that can generate a mapping from such complex formulations as the relation between the formulation ingredients and the PU measured properties. Therefore, only the data from CB2 was used in this chapter.
Table 5-1. Dataset for neural network feasibility study

Inputs for the network correspond to the problem feature (i.e., mechanical properties) differences between a pair of cases \((i,j)\), for \(i \neq j\). The targets for the network were calculated as the corresponding the solution feature (i.e., formulation ingredients) differences between a pair of cases \((i,j)\), for \(i \neq j\).

<table>
<thead>
<tr>
<th>Data set</th>
<th>Case Base 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original No Cases</td>
<td>34</td>
</tr>
<tr>
<td>Cases deleted</td>
<td>1 (Case No. 14)</td>
</tr>
<tr>
<td>Cases for validation</td>
<td>11,12,22,25</td>
</tr>
<tr>
<td>No of input patterns</td>
<td>812</td>
</tr>
<tr>
<td>No of net inputs</td>
<td>5</td>
</tr>
<tr>
<td>No of net outputs*</td>
<td>8 real-valued</td>
</tr>
<tr>
<td></td>
<td>24 binary-valued</td>
</tr>
</tbody>
</table>

*The outputs for the network are binary codified, and therefore each data record in the network output corresponds to a triad of ones and zeros. If the formulation increase the data record is codified as 001, if decrease as 100, and if it does not change as 010.

A second data set can be obtained to train the network by taking only the differences between a probe case and its first five closest retrieved cases. This will result in a smaller data set than the one previously described but it would include significantly smaller differences that the network could be able to learn more easily. As in the previous example, test cases correspond to cases No. 11, 12, 22 and 25. The total number of input patterns \(8p\) is obtained by (i) calculating the first five closest cases when each case is a query and (ii) calculating the differences between the case-problem attributes for each query and its first five retrieved cases (Figure 5-11).
Figure 5-11. Datasets for the artificial neural network feasibility study
Two data sets were chosen for the network training. The first one, consists of feature differences between all the cases in the data set (after the validation set has selected) leading to 812 exemplars. The second one consist of feature difference between the each case and its first five retrieved cases (after taking out the validation set) leading to 145 training exemplars.

5.7.1.1.2 Data pre-treatment

Once calculated the differences between attribute values, the input patterns for the network were normalised using the min-max normalisation explained in chapter 3 p. 64 in the interval \([-1, 1]\). As it has been said, the network’s training is influenced by the type of input and output patterns used. It this thesis both real-valued and binary-valued outputs were used in training.

The binary-valued outputs were obtained by codifying each difference in formulation ingredients between two cases into a triad of ones and zeros according to either a negative, positive or null difference.

- / +

- Negative 1 0 0
- Positive 0 0 1
- Null 0 1 0

An example is given in Figure 5-12.
Differences between the formulation ingredients between two cases (e.g. case i and case j) were calculated. This difference was codified into binary values depending on whether the change in ingredients was negative (case’s ingredient should be increased), positive (case’s ingredient should be decreased), or zero (leave a case’ ingredient as it is).

### 5.7.1.1.3 Network data sets

Without enough formulations to justify splitting the data into training and test sets, the cross-validation approach was adopted. By following this approach the training set was partitioned into five distinct segments. Each sample in the training set completely specifies all the inputs as well as the outputs that are desired when those inputs are presented. Five networks therefore were trained, each time using random chosen training sets with 792 exemplars and training-test sets with 20 exemplars. The network with the lowest error with respect to the training test set was selected. The performance of the selected network was confirmed by measuring its performance on a third independent set of data called a test set. If the error in the test set is not acceptable, a common practice is to joint the training and training test set and re-train [181 p.183]. This is shown in Figure 5-14.

<table>
<thead>
<tr>
<th>Table 5-2. Data sets used for network model</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Training</strong></td>
</tr>
<tr>
<td>Training Set</td>
</tr>
<tr>
<td>Training Test Set</td>
</tr>
<tr>
<td><strong>Validation on selected network</strong></td>
</tr>
<tr>
<td>Validation Set</td>
</tr>
<tr>
<td>Case ID 11, 12, 22 and 25</td>
</tr>
</tbody>
</table>

For networks using binary-valued outputs, the cross-validation and test errors were quantified in the following manner. Two types of errors were distinguished, namely, “major errors” and “minor errors”. They differ in how significantly the difference between the network’s output from the desired output is. This is best illustrated by an example. Let's suppose that the desired output is 'increase formulation variable' codified as (1 0 0). The possible network’s output can be either (1 0 0), (0 1 0) or (0 0 1). The first output would have null error as it corresponds to the desired response. The second output corresponds to a minor
error. This is because the network is considering that "no change" in formulation should be made when in fact an increase should be performed. The third output corresponds to a major error, as the network is considering decreasing the formulation variable when in fact it should increase it. This error is considerably more serious than "making no change" (i.e. option two) and therefore is counted as a major error (Figure 5-13).

<table>
<thead>
<tr>
<th>Output is</th>
<th>Real Value is</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decrease</td>
<td>Decrease</td>
<td>No error</td>
</tr>
<tr>
<td>No change</td>
<td>Decrease</td>
<td>Minor Error</td>
</tr>
<tr>
<td>Increase</td>
<td>Decrease</td>
<td>Major Error</td>
</tr>
</tbody>
</table>

Figure 5-13. Definition of error for networks with binary-valued outputs.

5.7.1.1.4 Network architecture

There has been much debate about how to find the correct network architecture for a particular problem. A large body of research exists which suggest several rules of thumb that can be applied to the problem at hand [181, 206, 209, 210]. In considering how to find the correct network architecture that can map the feature sets studied in this chapter, the first point to consider is what is meant by the term network architecture.

The network architecture is the description of the number of the layers in an ANN, each layer's transfer function, the number of neurons per layer, and the connections between layers. Finding the correct network architecture is a not an easy task. Aside from the number of neurons in a network's output layer, the number of neurons in each layer can be found by trial and error. One rough guideline for choosing the number of hidden neurons in many problems is the geometric pyramid rule [181 p. 176]. For a one-hidden layer feed-forward network, with n input neurons and m target neurons the rule states that the hidden layer would have \( \sqrt{mn} \) neurons. However, the problem studied in this thesis is highly complex and this rule may underestimate the number of hidden neurons required. Therefore, the systematic change of the number of number of neurons in each layer to find an optimum network's architecture was the approach used in this study (Figure 5-14).
5.7.1.2 Algorithm in the MATLAB Artificial Neural Network Toolbox

The implementation of the algorithm to train the ANN is described in this section.

To implement a two-hidden layer feed-forward ANN that could guide CBR adaptation the MATLAB 7 Artificial Neural Networks Toolbox was used. The systematic change in the number of neurons in each layer was performed using MATLAB (Figure 5-14).

Figure 5-14. Methodology followed to find an optimum network architecture.
When using MATLAB 7 ANN Toolbox, the number of input and output variables is first chosen. This is accomplished by specifying the two matrices $P$ and $T$, where $P$ is a $n \times m$ matrix; $m$ is the number of input variables, and $n$ the number of sets of training data; and $T$ is a $n \times l$ matrix; $l$ is the number of output variables. First, one hidden layer was used with a number of hidden neurons ranging from 1 to 100 but with a monotonically unitary increase in each iteration. Typically by increasing the number of hidden neurons the modelling ability of the network increases and the error decreases. Therefore, an optimum number of hidden layers can be obtained by looking at the iteration that gives a low error while at the same time maintains the training time low.

To create a feed-forward back-propagation network; the function newff was used for this purpose:

\[
\text{net} = \text{newff}([PR],[Si S2...SN],[TF1 TF2...TFN],BTF,BLF)
\]

newff takes

- $PR$ - Rx2 matrix of min and max values for R input elements
- $Si$ - Size of $i$th layer, for $NI$ layers
- $TFi$ - Transfer function of $i$th layer, default = ‘tansig’
- $BTF$ - Back-propagation network training function, default = ‘trainlm’
- $BLF$ - Back-propagation weight/bias learning function, default = ‘learngdm’

and returns an $N$ layer feed-forward back-propagation network.

MATLAB offers various options to choose for the parameters and algorithms to implement a network. For the transfer functions $TFi$ any differentiable transfer function such as ‘tansig’ (hyperbolic tangent sigmoid), ‘logsig’ or ‘purelin’ (linear) can be used. Training with Levenberg-Marquardt back-propagation algorithm is the training function that was used because it is very fast, although it requires a lot of memory to run. The back-propagation learning used was the LEARNGDM (gradient descent with momentum weight and bias learning function). This choice was appropriate to perform the preliminary tests.

5.7.1.2.1 Training

Early experiences using newff indicated that it is a random process since it was found that the result of the execution of this algorithm each time is different. Other conditions kept the same; two executions of this function usually gave quite different converging histories of training by the training algorithm. The MATLAB 7 ANN Toolbox does not longer use the standard back-propagation for training feed-forward networks because is obsolete [210 ch. 12]. Instead the default training function is based on the Levenberg-Marquardt back-propagation algorithm [210 ch. 12] using the MATLAB function train:
\[
[\text{net, TR, Y, E}] = \text{train(NET, P, T)}
\]

The function `train` trains a network `net` according to `net.trainFcn`, using the training parameter values indicated by `net.trainParam`.

`train` takes,
- `net` - Artificial Neural Network
- `P` - Network inputs
- `T` - Network targets, default = zeros
and returns,
- `net` - New network
- `TR` - Training record (epoch and perf)
- `Y` - Network outputs
- `E` - Network errors.

Input optional arguments include `T` which need only be used for networks that require targets.

The presentation of all input vectors to the network is typically done through one *epoch* of training. An *epoch* is one sweep through all the records in the training set. The network is then updated according to the results of all those presentations. The number of epochs in training is usually set by the user and then when the maximum number of epochs is reached, training stops. Training also stops if a performance goal is met, or any other stopping condition of the function `net.trainFcn` occurs.

### 5.7.1.2.2 Simulation

After the ANN is trained, one can predict output from input by using simulation algorithms in terms of the obtained parameters `net`, `TR`, `Y`, `E`. For feed-forward networks one of the functions in MATLAB is `sim`:

\[
[Y, E, perf] = \text{sim(net, T)}
\]

`sim` takes,
- `net` - Network
- `P` - Network inputs
- `T` - Network targets, default = zeros
and returns,
- `Y` - Network outputs
- `E` - Network errors
- `perf` - Network performance

The pseudo-code to create, train and simulate the neural network using the MATLAB 7 ANN Toolbox used in this project to assist CBR adaptation is presented as follows:
Present the input patterns \( P \) and targets \( T \)
inputs = \( P \)
targets = \( T \)

Create a two-layer feed-forward network
for iteration =1:50
    for Number of Neurons in Layer2 = 0:100
        for Num Neurons Layers = fixed at optimum number
            net = newff([Inputs range],[No units of each layer],{'transfer function for each layer'});
        end
    end
end

Network Training
net.trainParam.epochs = 50;
[net,tr,Yo,E] = train(net,P,T);
end
end

Network Simulation
Present the input patterns \( P \) to simulate the network.
inputs = \( P \)
\[ Y \] = sim(net,P)

Figure 5-15. Pseudo-code to create and train the neural network using MATLAB.

The network's functions and training parameters used in MATLAB for the feed-forward back-propagation networks are presented in Table 5-3.

<table>
<thead>
<tr>
<th>Function</th>
<th>Matlab reference</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Function</td>
<td>trainlm</td>
<td>Levenberg-Marquardt back-propagation algorithm</td>
</tr>
<tr>
<td>Performance function</td>
<td>mse</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>Transfer Function</td>
<td>tansig</td>
<td>Hyperbolic tangent sigmoid transfer function</td>
</tr>
<tr>
<td>Learning Function</td>
<td>learngdm</td>
<td>Gradient descent with momentum weight and bias learning function</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Network Parameters</th>
<th>Matlab name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of epochs to train</td>
<td>net.trainParam.epochs</td>
<td>100</td>
</tr>
<tr>
<td>Performance goal</td>
<td>net.trainParam.goal</td>
<td>1e-5</td>
</tr>
<tr>
<td>Maximum validation failures</td>
<td>net.trainParam.max_fail</td>
<td>5</td>
</tr>
<tr>
<td>Minimum performance gradient</td>
<td>net.trainParam.min_grad</td>
<td>1e-10</td>
</tr>
</tbody>
</table>

5.7.1.3 Algorithm in GHOST

Experiences using the MATLAB 7 ANN Toolbox indicated that when the number of iterations increases, the computing time arises leading to a lengthy training process. This is due to the fact that the TRAINLM is suited only for networks with a small number of weights [208]. An alternative neural net software programme, GHOST was used in parallel to MATLAB. GHOST is ANN software written in C by Dr Chris Hinde at Loughborough University. Because the software has been developed locally and the source is available changes may be made to it
easily to accommodate the current research. The network functions and training parameters available in GHOST and used for the feed-forward back-propagation networks are presented in Table 5-4.

Initially, MATLAB was used to find the optimum architecture when real-valued outputs were used. Once an idea of the optimum architecture was found, further training was carried out using GHOST. Several networks were trained with different training and test sets as mentioned formerly.

<table>
<thead>
<tr>
<th><strong>Function</strong></th>
<th><strong>Name</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Function</td>
<td>Back-propagation algorithm</td>
</tr>
<tr>
<td>Performance function</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>Transfer Function</td>
<td>Logistic sigmoid transfer function</td>
</tr>
<tr>
<td>Learning Function</td>
<td>Generalised delta function</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Network Parameters</strong></th>
<th><strong>Value</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of epochs to train</td>
<td>&gt;1000</td>
</tr>
<tr>
<td>Performance goal</td>
<td>1e-3</td>
</tr>
<tr>
<td>Maximum validation failures</td>
<td>5</td>
</tr>
</tbody>
</table>

5.8 Using an Artificial Neural Network to Adapt Cases

The final aim of this chapter is to adapt a retrieved case, which in terms of CBR stands for the suggestion of a formulation that can have the mechanical properties that are required by a user. This is also the ultimate objective of a formulator (see Figure 5-16 and Figure 5-17).

Figure 5-16. The PU formulation problem.
The ultimate objective of a formulator is to choose the right ingredients, their quantities and the process conditions that would meet the desired specifications.
It was explained in the preceding section that by training an ANN, a mapping between the network's input and the output patterns could be obtained.

The trained network is intended to be used to adapt cases in the following manner (refer to Figure 5-18). First, the problem features difference between a retrieved case and a query is presented to the network. The trained network has learnt that for various delta properties, certain changes in the output variables result. The trained network, therefore, should report on how the retrieved case solution features (i.e. formulation ingredients) should be adapted. However, the network does not know "how much" the ingredients should be adjusted. It only knows that certain ingredients in the retrieved case's formulation need to be either increased, decreased or left unchanged in order to propose a solution for the query that potentially can meet the requirements or at least be better than the retrieved solution.
Figure 5-18. Artificial Neural network for adaptation of PU retrieved cases

First, normalised problem features difference between query and a retrieved case are calculated. Then, this difference is presented to the network. The ANN which has learnt that for various delta properties, certain changes in the output variables result, report on adjustments that the retrieved case solution features (i.e. formulation ingredients) required. However, these adjustments are qualitative in nature. A quantitative adaptation is required in order to propose a PU formulation for the query that potentially can meet the requirements or at least be better than the retrieved solution.

A method to adjust the amount of ingredients that need to be changed consists of a “vote mechanism” that uses ranked retrieved cases. Their contribution to the answer is determined by a relevance factor defined in terms of the distance between the retrieved case and the query. The relevance factor for each case is determined by the following relation:

\[ f_i = 1 - \frac{d_i}{\sum_i d} \]  

(5-1)

where \( r \) is the number of retrieved relevant cases; and \( d \) is the distance function used in the similarity assessment.

The definition of this factor allows to take into account the similarity between the query and the retrieved cases i.e. the similar the retrieved case to the query is (smaller the distance), the higher the contribution of that retrieved case to the answer should be.

An example of the adaptation of cases using this vote mechanism is shown as follows:
One approach to adjust quantitatively case-solution features (i.e. ingredients amount) is by using a voting mechanism. First a relevance factor defined in terms of the distance of between the retrieved case and the query is calculated. This factor reflects the contribution of retrieved cases to the answer. Once the relevance factors are calculated, case-solution features are calculated based on the contribution to the answer by the retrieved case.

5.9 Results and Discussion

The results of the training of several feed-forward neural networks and the use of a trained network with the lowest training error to adapt cases are presented in this section. Results regarding training of the networks, using as the training examples the differences between all cases in the case base with each other (i.e. 812 exemplars), are presented first. Two types of outputs were studied i.e. real-valued outputs and binary-valued outputs as was described in the methodology section (p.170). The normalised real-valued outputs preliminary experiments served to narrow down the search space for an adequate network architecture, therefore, these results are presented first. Six binary-valued output network training runs were carried out in total:

1. five using cross-validation training subsets each one with 792 exemplars and its respective training-test subsets each one with 20 exemplars,
2. one using the whole data set (i.e. 1122 exemplars) with training and training subset with 1102 and 20 exemplars, respectively.

Results regarding training of the networks, using the training examples as the differences between all cases in the case base with their first five retrieved cases (i.e. 145 exemplars), are presented in section 5.9.1.3. In this case, only two networks were trained:

1. one using real-valued outputs with training and training-test subsets with 125 and 20 exemplars, respectively,
2. another using binary-valued outputs with training and training-test subsets with 125 and 20 exemplars, respectively.

### 5.9.1 Artificial Neural Network Implementation

#### 5.9.1.1 Real-valued outputs

A systematic change in the number of neurons in two layers was used in this study to train a feed-forward ANN with real-valued outputs normalised in the interval \([-1,1]\) in MATLAB. Early tests of the perceptron network (e.g. 5 inputs-8 outputs or simply 5-8) yielded high mean square errors. Figure 5-20 a) shows the performance (i.e. mean square error) of the network against the number of hidden neurons in the first layer. It can be seen that as the number of neurons is increased, the error of the network decreases monotonically. Figure 5-20 b) shows the computing effort (training time in seconds) versus the number of hidden neurons in the first layer. By increasing the number of neurons the training time is also increased in a linear fashion for the chosen interval. In addition, it can be observed that by using more than 80 neurons the error does not decrease significantly but it results in a high computation training time.

![Figure 5-20. Artificial neural network performance.](image)

- a) It can be seen in that as the numbers of neurons increase the performance index of the network i.e. the mean squared error decreases monotonically. However, after 80 neurons, this reduction starts to show an asymptotic tendency.

- b) The training time increases as the number of neurons in the hidden layer increase.

In order to find a balance between computational effort (i.e. low training time), good generalisation (i.e. not many hidden neurons) and good modelling ability (i.e. enough hidden neurons), 80 hidden neurons were chosen in each of the two hidden layers. A partial justification for choosing this number of neurons relies also in the fact that the two-hidden layer architecture is known to approximate any continuous mapping provided that the number of hidden neurons is sufficiently large [206]. Additionally, it is expected that if the network with this size shows good performance, the network’s configuration can be later optimised by using growing and pruning algorithms if required (see for instance [206 p.353]).
5.9.1.2 Binary-valued output results

5.9.1.2.1 Cross-validation

After 20 thousand epochs, five trained networks using binary-valued outputs gave the following results

<table>
<thead>
<tr>
<th>Feature</th>
<th>NET 1</th>
<th>NET 2</th>
<th>NET 3</th>
<th>NET 4</th>
<th>NET 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max No. of errors</td>
<td>0.901375</td>
<td>0.991220</td>
<td>0.991027</td>
<td>0.915647</td>
<td>0.769905</td>
</tr>
<tr>
<td>maxsamp</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>maxname</td>
<td>output2</td>
<td>output2</td>
<td>output2</td>
<td>output24</td>
<td>output21</td>
</tr>
<tr>
<td>Num wrong</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Max sq. errors</td>
<td>6.251496</td>
<td>6.640180</td>
<td>6.294469</td>
<td>5.834857</td>
<td>6.344229</td>
</tr>
<tr>
<td>Mean sq. errors</td>
<td>4.907088</td>
<td>5.276645</td>
<td>5.362985</td>
<td>4.875774</td>
<td>4.981290</td>
</tr>
</tbody>
</table>

The measure of network's performance (the mean square error) for these networks is high when compared to the performance goal of 0.001 indicating that convergence was not reached. For instance, for NET 1 the minimum error reached was 4.90 after twenty thousand iterations, which is about 4.90 thousand times the performance goal. In addition, the error after twenty thousand epochs was about the same for all networks (approximately 5.0 in average) indicating that these errors may not depend on the data subdivision. It is probable that either the training set or the training-test set were not representative for the relationship to be modelled i.e. they did not contain important information the network should have learnt. Especially for the training-test sets that contained only twenty samples, a high chance of selecting a non-representative subset was considerable. In addition, these results are not enough evidence for concluding that the networks' low performance is due to (i) cross-validation subset selection alone, (ii) strong noise in the data and/or (iii) an error optimisation problem (i.e. a local minimum instead of the global minimum was incidentally reached).

Although cross-validation is useful when the number of data is limited, it has shown to have several drawbacks with the data set used in this thesis. First, it requires the network's training process to be repeated several times (in this case 5) with the consequently computing demands and it does not guarantee reliable and reproducible results. In the networks trained, irreproducibility is due to a number of factors. Primarily, the fact that learning is an iterative process using different random starting weights, small changes in the weights or training set can result in different networks. Second, the network's learning depends, to a great extent, on the choice of the training sets. Thus, the training set needs to contain the necessary information for the network to both learn the desired mapping and be able to generalise. The selection of the training sets for the networks studied was done randomly. Reasons for
choosing random training sets stem from the fact that it is difficult to know what features make a training subset characteristic of a population.

### 5.9.1.2.2 Merged cross-validation sets

It was mentioned in the preceding sections that if the cross-validation error is not acceptable, a common practice is to join the training and testing set and re-train [181 p.183]. This assumes the training set to be representative of the population. It was assumed that by using the whole data set, a set representative of the population was obtained. The results of the re-trained network using training and training-test sets merged are shown below.

<table>
<thead>
<tr>
<th>Feature</th>
<th>NET 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. Iterations</td>
<td>17000</td>
</tr>
<tr>
<td>Max No. of errors</td>
<td>1</td>
</tr>
<tr>
<td>maxsamp</td>
<td>19</td>
</tr>
<tr>
<td>maxname</td>
<td>output 11</td>
</tr>
<tr>
<td>Num wrong</td>
<td>5</td>
</tr>
<tr>
<td>Max sq errors</td>
<td>6.577664</td>
</tr>
<tr>
<td>Mean sq errors</td>
<td>1.567303</td>
</tr>
<tr>
<td>Av. Absolute errors</td>
<td>2.239573</td>
</tr>
</tbody>
</table>

The results when an independent test set was presented to the network are presented in Table 5-7.

<table>
<thead>
<tr>
<th>Retrieved</th>
<th>Probe</th>
<th>Ing 1</th>
<th>Ing 2</th>
<th>Ing 3</th>
<th>Ing 4</th>
<th>Ing 5</th>
<th>Ing 6</th>
<th>Ing 7</th>
<th>Ing 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>24</td>
<td>decrease</td>
<td>increase</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
<td>increase</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>decrease</td>
<td>no change</td>
<td>increase</td>
<td>increase</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
</tr>
<tr>
<td>22</td>
<td>28</td>
<td>increase</td>
<td>no change</td>
<td>no change</td>
<td>decrease</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
<td>no change</td>
</tr>
<tr>
<td>25</td>
<td>17</td>
<td>decrease</td>
<td>no change</td>
<td>increase</td>
<td>no change</td>
<td>increase</td>
<td>increase</td>
<td>no change</td>
<td>no change</td>
</tr>
</tbody>
</table>

The results described above indicate that a trained ANN is able to recognise important patterns in formulation data and therefore can be used to diagnose formulation changes required for adaptation of retrieved cases. However, this is not enough evidence that the network is able to generalise when presented with more data independent from the original data set. Given the large number of hidden units, and the small training set the network is likely to be over-fitting the data.

The results shown above indicate that the performance goal was mainly limited by the ability of collecting new data and by the computational resources available. More PU formulations are required if a trained network free of generalisation issues is expected. In addition, these results reiterate the importance of the relationship between (i) the number of
hidden neurons and (ii) the training set, in the implementation of ANNs. This relationship can lead towards two extremes as it has been illustrated. First, the network can learn both the data and the noise present in the data and hence it does not generalise well. Second, the scarcity of the training set is partially compensated by limiting the number of hidden neurons but this prevents the network from learning as it should.

5.9.1.3 Training with CBR retrieved differences

A second data set obtained by taking only the differences between probe cases and its first five closest retrieved cases was used to train a two-hidden layer feed-forward back-propagation network, as it was described in section 5.7.1.1.1 (Figure 5-11). The idea behind this approach is that because smaller differences are present when similar cases are compared, the network should learn more rapidly compared to the case when higher differences are present even though the training set is reduced significantly from 812 to 145 exemplars.

After 20 thousand epochs, the trained network (NET 8) using binary-valued outputs gave the following results:

<table>
<thead>
<tr>
<th>Feature</th>
<th>NET 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max No. of errors</td>
<td>0.927057</td>
</tr>
<tr>
<td>maxsamp</td>
<td>14</td>
</tr>
<tr>
<td>maxname</td>
<td>output9</td>
</tr>
<tr>
<td>Num wrong</td>
<td>20</td>
</tr>
<tr>
<td>Max sq. errors</td>
<td>7.143233</td>
</tr>
<tr>
<td>Mean sq. errors</td>
<td>4.908544</td>
</tr>
<tr>
<td>Av. Absolute errors</td>
<td>9.339203</td>
</tr>
</tbody>
</table>

From the results present above it can be seen that the error did not decrease. This suggests that the data presented to the network is not representative of the population and confirms the conclusions from the preceding paragraphs in which it was suggested that collection of more independent PU formulation data is necessary.

5.9.2 Adaptation using the Artificial Neural Network

Despite the fact that the NET 6 is not able to generalise, it is a good example that the trained neural network can learn important patterns in PU formulation data, as a PU expert would do. It can remember relationships within the data to diagnose formulation changes required for adaptation of retrieved cases but it is not able to apply its learning capabilities to other sets of data accurately.

For the purposes of this research, the main goal was to find a network that could do well enough to prove that experimental formulation data contained in the case-base could be used
as a knowledge container to guide CBR adaptation. Despite the fact that NET 6 does not
generalise accurately, it proves that it is capable of reuse PU real formulation data to guide
CBR adaptation. The way in which it can do so is illustrated as follows.

The results from the validation test shown in Table 5-7 were used to find the formulation
ingredients according to the method explained in section 5.8, when the first two retrieved
cases using the Euclidean distance were used. The method of how to arrive at the validation
results for the four tests samples is described below:

1. Retrieve Cases

A user-query is presented to the CBR system, and this retrieves the closest formulations to
it according to some measure of similarity (e.g. Euclidean distance, low order moments or
high order moments)

<table>
<thead>
<tr>
<th>Sample</th>
<th>Euclidean distance</th>
<th>Low Order Geometric Moment (i.e. N=3)</th>
<th>High Order Geometric Moment (i.e. N=20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>24</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>27</td>
<td>0.513</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>0.142</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>0.189</td>
<td>2</td>
<td>27</td>
</tr>
<tr>
<td>18</td>
<td>0.184</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>17</td>
<td>0.195</td>
<td>21</td>
<td>26</td>
</tr>
<tr>
<td>17</td>
<td>0.119</td>
<td>17</td>
<td>26</td>
</tr>
<tr>
<td>15</td>
<td>0.162</td>
<td>18</td>
<td>18</td>
</tr>
</tbody>
</table>

2. Calculate feature differences between case-problem and probe

Normalised difference between mechanical properties of a probe and its first retrieved
cases is calculated.
<table>
<thead>
<tr>
<th>Probe Retrieved Cases</th>
<th>Normalised problem feature difference between case and probe</th>
<th>Δ Hardness</th>
<th>Δ Density</th>
<th>Δ Tensile Strength</th>
<th>Δ Elongation</th>
<th>Δ Compression Set</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Euclidean Distance</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>-0.22453</td>
<td>0.24878</td>
<td>-0.04790</td>
<td>0.28571</td>
<td>-0.05747</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>0.06563</td>
<td>0.00488</td>
<td>0.29940</td>
<td>0.41071</td>
<td>-0.01149</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>-0.00864</td>
<td>-0.03415</td>
<td>-0.00599</td>
<td>0.05357</td>
<td>0.12644</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>-0.00173</td>
<td>-0.02927</td>
<td>0.02994</td>
<td>0.01786</td>
<td>0.18391</td>
</tr>
<tr>
<td>22</td>
<td>28</td>
<td>0.60104</td>
<td>-0.33171</td>
<td>0.55090</td>
<td>-0.25000</td>
<td>0.37931</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td>0.05527</td>
<td>0.02439</td>
<td>0.02395</td>
<td>-0.14286</td>
<td>-0.11494</td>
</tr>
<tr>
<td>25</td>
<td>17</td>
<td>0.00864</td>
<td>-0.07805</td>
<td>-0.01198</td>
<td>0.03571</td>
<td>-0.08046</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>0.02591</td>
<td>0.02439</td>
<td>0.04790</td>
<td>0.01786</td>
<td>-0.14943</td>
</tr>
<tr>
<td><strong>Low Order Moments</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>-0.22453</td>
<td>0.24878</td>
<td>-0.04790</td>
<td>0.28571</td>
<td>-0.05747</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>0.02418</td>
<td>0.02927</td>
<td>0.26347</td>
<td>0.25000</td>
<td>0.27586</td>
</tr>
<tr>
<td>12</td>
<td>8</td>
<td>0.13990</td>
<td>0.11220</td>
<td>0.01796</td>
<td>0.01786</td>
<td>-0.08046</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>0.14853</td>
<td>-0.03415</td>
<td>0.14970</td>
<td>0.05357</td>
<td>-0.02299</td>
</tr>
<tr>
<td>22</td>
<td>18</td>
<td>0.07945</td>
<td>0.07317</td>
<td>0.13772</td>
<td>-0.03575</td>
<td>-0.02299</td>
</tr>
<tr>
<td>21</td>
<td></td>
<td>0.05181</td>
<td>0.12195</td>
<td>0.14970</td>
<td>0.10714</td>
<td>0.03747</td>
</tr>
<tr>
<td>25</td>
<td>17</td>
<td>0.00864</td>
<td>-0.07805</td>
<td>-0.01198</td>
<td>0.03571</td>
<td>-0.08046</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>0.02591</td>
<td>0.02439</td>
<td>0.04790</td>
<td>0.01786</td>
<td>-0.14943</td>
</tr>
<tr>
<td><strong>High Order Moments</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>-0.22453</td>
<td>0.24878</td>
<td>-0.04790</td>
<td>0.28571</td>
<td>-0.05747</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>0.14853</td>
<td>-0.03415</td>
<td>0.14970</td>
<td>0.05357</td>
<td>-0.02299</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>0.54404</td>
<td>0.22927</td>
<td>0.47305</td>
<td>0.03571</td>
<td>0.01149</td>
</tr>
<tr>
<td>22</td>
<td>18</td>
<td>0.07945</td>
<td>0.07317</td>
<td>0.13772</td>
<td>-0.05357</td>
<td>-0.02299</td>
</tr>
<tr>
<td>26</td>
<td></td>
<td>0.27634</td>
<td>-0.34634</td>
<td>0.31737</td>
<td>-0.12500</td>
<td>-0.03448</td>
</tr>
<tr>
<td>25</td>
<td>26</td>
<td>0.22971</td>
<td>-0.44878</td>
<td>0.28144</td>
<td>0.05357</td>
<td>0.00000</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>0.02591</td>
<td>0.02439</td>
<td>0.04790</td>
<td>0.01786</td>
<td>-0.14943</td>
</tr>
</tbody>
</table>

3. Present problem feature differences to the network to obtain solution feature adjusting parameters.
<table>
<thead>
<tr>
<th>Probe - Retrieved</th>
<th>Adapt case [neural network retrieved data]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ing 1</td>
</tr>
<tr>
<td>Euclidean Distance</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>decrease</td>
</tr>
<tr>
<td>27</td>
<td>decrease</td>
</tr>
<tr>
<td>12</td>
<td>decrease</td>
</tr>
<tr>
<td>20</td>
<td>increase</td>
</tr>
<tr>
<td>22</td>
<td>increase</td>
</tr>
<tr>
<td>25</td>
<td>decrease</td>
</tr>
<tr>
<td>Low Order Moments</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>decrease</td>
</tr>
<tr>
<td>1</td>
<td>decrease</td>
</tr>
<tr>
<td>12</td>
<td>decrease</td>
</tr>
<tr>
<td>2</td>
<td>decrease</td>
</tr>
<tr>
<td>22</td>
<td>increase</td>
</tr>
<tr>
<td>25</td>
<td>decrease</td>
</tr>
<tr>
<td>High Order Moments</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>decrease</td>
</tr>
<tr>
<td>12</td>
<td>increase</td>
</tr>
<tr>
<td>2</td>
<td>decrease</td>
</tr>
<tr>
<td>22</td>
<td>increase</td>
</tr>
<tr>
<td>25</td>
<td>decrease</td>
</tr>
</tbody>
</table>

4. Obtain formulation ingredients by using the "vote mechanism"

a) Calculate relevance factors
### CBR Adaptation

#### Retrieved case Relevance factor \([f]\)

<table>
<thead>
<tr>
<th>Retrieved case</th>
<th>Euclidean Distance</th>
<th>Low Order Moments</th>
<th>High Order Moments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ing 1</td>
<td>Ing 2</td>
<td>Ing 3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Retrieved case formulation [CBR output data]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ing 1</td>
</tr>
</tbody>
</table>

### Adjust CBR retrieved formulations using network output and relevance factors

#### Sample Predicted Formulation Using Neural Network Output data and Relevance factor \([f]\)

<table>
<thead>
<tr>
<th>Sample</th>
<th>Euclidean Distance</th>
<th>Low Order Moments</th>
<th>High Order Moments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ing 1</td>
<td>Ing 2</td>
<td>Ing 3</td>
</tr>
</tbody>
</table>

#### 5. Calculate absolute error with known probe data

The absolute errors of the normalised formulation variables were calculated.
Absolute error = (predicted value - known value)^2

a) Euclidean Distance

<table>
<thead>
<tr>
<th>Sample</th>
<th>Predicted Formulation Using Neural Network Output data and Relevance factor f</th>
</tr>
</thead>
<tbody>
<tr>
<td>adapt ID11</td>
<td>92.171 2.198 1.500 0.300 0.250 0.250 3.750 97</td>
</tr>
<tr>
<td>adapt ID12</td>
<td>92.95 1.5 0 0.3 0.75 0.75 3.75 85</td>
</tr>
<tr>
<td>adapt ID22</td>
<td>92.94 1.5 0.770 0.3 0.25 0.51 3 85</td>
</tr>
<tr>
<td>adapt ID25</td>
<td>92.31 3 1.5 0.3 0.25 0.25 3 85</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Known Formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample 1 ID11</td>
</tr>
<tr>
<td>sample 2 ID12</td>
</tr>
<tr>
<td>sample 3 ID22</td>
</tr>
<tr>
<td>sample 4 ID25</td>
</tr>
</tbody>
</table>

The results below show that NET 6 has learnt from the PU formulation data and has enabled the knowledge contained in the CB to be used to adapt retrieved cases. Results for the four probe cases show that the prediction error (i.e. square error) is low for most of the formulation variables and hence is an indication that the network. NET 6 has learnt the set of PU formulation data presented to it and it is able to assist in the development of PU formulations. NET 6 is outperforming human PU expert capabilities to retrieve a formulation given a set of mechanical properties (i.e. a user-query) in the sense that a human expert cannot produce at such detail a PU formulation.

5.10 Summary

A two-hidden layer feed-forward back propagation ANN with binary-valued outputs met the goal of this chapter

"the goal is to adjust a retrieved formulation to find the ingredients and their quantities that result in a PU formulation that approximate the user requirements" p. 160

The trained network NET 6 was able to recognise relationships between case-problem feature differences and case-solution feature differences using real PU formulation data and it was used successfully to diagnose formulation changes required for adaptation of retrieved cases using a sample of four probe cases. Although this network is unable to generalise well when presented with more data independent from the original data set, this network proves that real formulation data in the case-base can be used as a "knowledge repository" that can guide CBR adaptation without human expert intervention.
The two-hidden layer feed-forward back propagation ANN with binary-valued outputs was selected in contrast to other networks because:

1. For real-valued outputs MATLAB was used for the creation of a single-layer perceptron network. Early tests of the perceptron network (5-8) yielded high mean square errors.
2. Experimentation with three-hidden layer feed-forward back propagation neural networks generated significantly higher mean square errors than two-layer networks.
3. Two-hidden layer feed-forward back propagation ANN with 80 neurons in each layer gave the lowest error when real-valued outputs were used. This architecture was used as a starting point to train networks with binary-valued outputs that could result in a lower error using an alternative ANN software GHOST.
4. For binary-valued outputs the creation of (5-80-80-24) networks gave improved error values but still they did not converge to an optimum required of 0.001. Therefore, this network can only be used as an adaptation knowledge tool to deal only with the experiences that are presented to it, just as a PU expert would do. An ANN trained with PU formulations acts as a memory repository that can used to assist the CBR system but it cannot be used to perform predictive functions outside the scope of the formulation data it learns from.

5.11 Conclusion

This chapter addresses the research question three included in Chapter 1. The main question to be solved is as follow, how can a retrieved PU case be adapted so that a solution to a formulation problem is obtained?

The direct problem is as follows: given (i) the user requirements for a PU formulation, (ii) retrieved cases that partially match a required PU formulation and (iii) no other domain knowledge than the case-base, adjust a retrieved formulation to find the ingredients and their quantities that result in a PU formulation fulfilling the user requirements. In addition, it was hypothesised that this problem might be solved by the application of the ANN paradigm.

The methodology implemented to solve the above question involved splitting the question in three sub-questions (A to C). These sub-questions pointed out to the implementation of an ANN, and a method to use the trained network to adapt cases.

5.11.1 Adaptation Using the Case-Base: Sub-question A

What approaches have been used in case adaptation using no other domain knowledge than the case-base? What are the advantages and disadvantages of these approaches?
It has been agreed by the CBR community that the adaptation of cases is a difficult task that has made difficult the deployment of complete CBR systems. In order to overcome this problem, several researchers have proposed to exploit the knowledge that might be already contained in the case-base by means of inductive techniques that elicit knowledge from the case-base and ease the adaptation knowledge bottleneck by learning from cases. For cases represented as feature vectors researchers [161, 164] have used feature differences between pair of cases, to propose adaptation rules [164] or adaptation cases [161]. However, it is not clear from the literature (i) what cases need to be compared to generate adaptation rules and cases and (ii) what rules or cases need to be applied to make the algorithm effective and efficient.

5.11.2 Using Artificial Neural Networks as Inductive Technique: Sub-question B

Can the artificial neural network approach be used to elicit knowledge from the case-base and use it to guide adaptation of cases? If so, what kind of artificial neural network can be used and why? What are the basic assumptions for using this technique; what are the limitations of this approach?

The ANN approach is a computational technique that has been used in product formulation application to map formulation ingredients or processing conditions to final product properties. Multilayer perceptrons with sigmoidal transfer functions have been the most common ANN used in product formulation applications to demonstrate their ability to learn from examples. Most studies agree in that adaptation (e.g. optimisation) of a formulation is a highly complex non-linear mapping problem because both sets of variables are composed of multiple inputs and outputs. These studies have illustrated that a two-layer (i.e. one hidden layer) feed-forward ANN can approximate the non-linear function between measured input and outputs to optimise formulations. This suggests that MLPs have the potential to capitalise on PU knowledge available in the form of formulations to solve the CBR adaptation problem this chapter proposed in section 5.2 p.159. However, the research available has failed to discuss in detail the issues regarding neural systems design and its implementation in formulating application.

5.11.3 An Artificial Neural Network Approach for Adaptation of Cases: Sub-question C

How can a neural network be implemented to adapt retrieved PU cases when these are represented as feature vectors? How can the knowledge learnt by the artificial neural network be used to adapt retrieved cases? How reliable is the adapted case found by the network?
A feed-forward back-propagation ANN that can adapt retrieved PU cases as well as a method to use the knowledge learnt by the network was implemented in this chapter. A three-layer feed-forward ANN trained using a modified back-propagation algorithm was found to be able to generate adaptation knowledge from a case-base composed of real PU formulation data. The network was trained to learn a relationship between the change in formulation properties and formulation ingredients. When a new query exists, the CBR system retrieves cases that partially match the query and the difference between the query and the closest retrieved case is presented to the trained network for it to propose what ingredients of the retrieved formulation need to be adjusted to compensate for the differences found.

In order to know how much to change the formulation ingredients, a vote mechanism was proposed. This method worked by taking into account the "weight" or contribution to the answer by defining a relevance factor based on the measure of distance used in the retrieval algorithm in a way that the similar the case to the query is (smaller the distance), the higher the contribution of the case to the answer should be.

Although the network has a low global error (4%), the network does not have good generalisation capabilities. This could be due to the fact that the network was trained with a very small data set and hence it is not able to discover the relationships between the large number of variables involved, which make the problem complex.

In addition, as it was explained in chapter 2, the chemistry of PUs is very sensitive to changes in any PU reactive chemical group (e.g. isocyanate, hydroxyl, amine) produced either by changes in processing conditions (e.g. temperature) or by introduction of chemicals intentionally (i.e. increase in formulation ingredients) or unintentionally (i.e. contamination, environmental humidity changes). A small change (proportional to the PU batch size) in any ingredient for a given formulation is likely to have strong effects on the properties of the foams. Therefore, it is difficult to find a suitable data set to train an ANN. A network likely to learn any mapping between formulation ingredients and mechanical properties would require a vast collection of data covering small variations in the ingredient's quantities. The size of the training data set is also likely to increase as more ingredients are used.

From the PU chemical point of view, certain ingredients have a major impact on the final PU properties. For instance, the catalyst used, and the amount of hydroxyl groups (i.e. coming from water, humidity, polyols, or contaminants) have the highest impact on the stoichiometry of the reaction and hence on the final properties of a foam.

The data sets used for training the ANNs in this thesis, came from a fractional statistical experimental design as explained in chapter 4. In this design of experiments, eight formulation variables were changed simultaneously between the levels suggested by the PU
expert to cover the whole range of property values. However, it has become apparent that the choice in the change of levels was too gross and did not allow mapping the property-formulation spaces continuously because the changes in properties were widely scattered. This suggests that for such a large number of formulation variables, more data need to be captured at smaller changes within the expert's suggested levels to be able to map the space continuously. In additions, changes in the quantity of reactants depend heavily on the type of reactant used, i.e. a change of 5 g in the amount of catalyst used has a greater effect on the foam final properties than the same change in the amount of the bulk polyol (in for example a 1 kg flexible PU foam formulation batch size). Additionally, some ingredients react and produce reactive by-products, which can at the same time lead to unexpected changes in the foams' final properties. As it is, for example, the case of the addition of water and the production of reactive amine that lead to a subsequent production of urea (refer to p. 16)
PART III. CONCLUSIONS AND FUTURE WORK
Chapter 6

CONCLUSIONS AND FUTURE WORK
A CBR-ANN Hybrid System for Polyurethane Formulation

6. CHAPTER 6. CONCLUSIONS AND FUTURE WORK

This thesis has studied the application of the CBR methodology and ANN approach to support the problems arising in the formulation of PUs. Specifically the hybrid CBR-ANN system circumvents the lack of understanding and quantitative models for calculating the mechanical properties in PUs, through the structure and representation expert's PU formulation experiences (i.e. successful and unsuccessful formulation recipes) into a CBR framework for problem solving. Given a user query, which is a problem description, the system retrieves similar experiences with the ultimate objective of predicting what formulation conditions (ingredients and their quantities) will allow the resulting product to meet the desired performance criteria. The CBR-ANN system therefore would allow reducing the time and hence the cost of the PU product development process while at the same time enables carrying out simulation studies. It also serves as an "institutional memory" or knowledge base that supports problem solving without direct human expert intervention.

This chapter is divided into three sections. The conclusions about the research questions that were drawn at the end of each chapter are presented again for convenience in the first section. The main contributions of this thesis are listed in section 6.2, and finally, suggestions for future work are detailed.
6.1 Conclusions about Research Questions

6.1.1 Research Question 1

What computational methodologies are available for dealing with complicated problems arising in PU formulation? What are the basic assumptions for using these techniques? What technique is the most appropriate to support the formulation of PUs?

Computational technologies for problem solving fall broadly into two types depending on how they assume human expert reasoning is evoked: these are RBS and CBR. Both methodologies are currently used to build KBS in product formulation domains as they complement each other. In these applications, it is normally assumed that human experts reason by either linking a series of rules or by referring to former solving experiences so that a solution to a new problem can be proposed. In real PU formulation practice, most human experts use both models of reasoning when solving problems. For example, a PU foam expert always reasons from past experience about the lessons learned when developing past formulations. Successes as well as errors come to mind and guide the whole formulation process. In addition, when refining the formulation, experts tend to look at pre-established rules. They are generally related the chemical nature of the ingredients to effects in mechanical properties of the material. Recall several rules described in chapter 2 e.g. “Rule 1: IF ester group separation (chain length) in linear adipate polyesters increase THEN tear strength decrease AND elongation (% and set) decrease AND density decrease”. However, due primarily to difficult rule elicitation and the nature of the data set available in this project, CBR is the methodology most appropriate to be implemented to support the PU formulating tasks. In particular, the data set used in this project, as explained in chapter 4, come from an industrial application that was developed by a trial an error process along various years of experimentation [211] and therefore, there is only a limited amount of causal knowledge that relates to its development.

6.1.2 Research Question 2

6.1.2.1 Sub-question A

What is the performance of the current local similarity measures based on standard metric distances? What are their advantages and disadvantages?

The assessment of retrieval by using standard distance functions is computationally simple. However, it is not clear what distance function is ought to be used and why and the implications of that choice. This study found that the most common distance functions used in the CBR literature i.e. Euclidean, City block, Canberra, Divergence, D7, D8, D9 and D10
(refer to Table 4-4) retrieve, with a high probability, the same query when only numerical
positive attributes are used. Although their individual performance (i.e. how accurate the
retrieval is when each distance is employed) was not assessed in this study, it is clear that the
main disadvantage of these standard distance functions is that their answer i.e. a number
represents the similarity between two cases, which cannot be interpreted further. For instance,
the distance number does not tell anything about the kind of similarity between two cases e.g.
if there are features that are most similar than others.

6.1.2.2 Sub-question B

What global similarity measures do exist that can be applied for the retrieval of PU cases?
What are the basic assumptions for using these approaches? How can these similarity
measures be used to assess the similarity between PU formulations? What are their
advantages and disadvantages? Do they perform better than current local measures based on
standard metric distances? If so, how is retrieval enhanced?

In terms of this work, global similarity connotes the matching of a case as a whole
(contrary to local retrieval which is a scalar weighted sum of attributes). Shape matching as a
global similarity measure between cases within the CBR framework was the approach
explored in this thesis. For using this approach, formulations can be represented as two-
dimensional images and matching is then performed by using moment functions in a similar
fashion to the approaches studied in image analysis in pattern recognition. This is, by
calculating a set of moments a case-image is characterised and can be compared to others at
different levels of image accuracy. Because the two-dimensional representation of a case does
not require the use of complex moments that compensate, for example, for problems
associated with noise when capturing an image e.g. by optical means, a set of geometrical,
central and Legendre can be used. Matching is performed by first calculating the moments of
the case $m^c_p$ and the query $m^q_p$ based on an image function definition for the case $f^c(x)$ and
the query $f^q(x)$. Then, the minimum Euclidean distance between the moments in the case-
base and the query $(m^c_p - m^q_p)$ is assessed to retrieve the closest case to the query (i.e.
smaller distance).

Although the computation of moments is mathematically more intensive than in
comparison to when standard distance functions are used, the use of moments for retrieval
offers various advantages. For example, different moment orders can be computed offering
two levels of detail in the assessment of similarity. The higher the order, the higher the level
of detail of the global shape characteristics of the image and therefore, if two PU formulations
are represented as images, they can be similar in at least two ways, namely, on a fine basis
Conclusions and Future Work

(when high order moments are used) or at a gross degree (when low order moments are used). Additionally, the problem of assigning weights to features was addressed in this thesis, so different expert weights can be also taken into account when matching cases using moments.

Research Question 2. Is a global similarity measure more effective than local similarity measures based on standard distances functions for the formulation of PU foams?

The global similarity measure based on moment descriptors presented in this chapter allows:

1. To quantify the extent to which a retrieved case is a useful solution to a query by the provision of a quantitative indication of this measure. This result is two fold as two degrees of similarities can be obtained, which refer to whether low or high order moments are used. However, the meaning of the results offered by high order moments remains unknown. Theoretically, high order moments contain more information about an image and therefore, it is intuitively assumed that they would provide a closest case when an image-query is presented. This hypothesis could not be evaluated experimentally because of data limitations issues, but it can be validated if future studies.

2. To assess similarity in cases differing in the number of attributes to be assessed for similarity by simply giving a weight of zero for the attributes that are not present in a query.

The problem of assigning weights to features when using moments for case retrieval was addressed in this thesis. Therefore, the assessment of similarity of cases differing in the number of attributes can be simply done by giving a weight of zero to attributes that are not present in a query.

The effectiveness of the global in comparison with the local one, however, could not be assessed. This was due to the fact that there were no probes or test cases i.e. cases with the “correct” formulation to assess retrieval in terms of which algorithm performs better. For this reason the retrieval methods could be compared to each other in terms of which one gives a better result (i.e. which is more accurate when compared to a test case) but in terms of how close or different their answers are. By means of a cluster analysis a comparison between distance measures used for each retrieval algorithm (i.e. local using standard distance functions and global using moment functions) was performed. It was found that low order geometric and Legendre moments and central moments of any order retrieve the same case as the Euclidean distance for both of the data sets used. This means that the Euclidean distance is analogue to a low order moment function that represents gross level image features. High order geometric and Legendre moments while enabling finer details about an image to be
represented retrieved different cases than their low order counterparts did and did not find any standard distance function counterpart.

It is suggested, that there is not single distinguished approach to similarity in CBR. Rather, CBR systems should allow the integration of different approaches to similarity and the selection of different concepts.

6.1.3 Research Question 3

6.1.3.1 Sub-question A

*What approaches have been used in case adaptation using no other domain knowledge than the case-base? What are the advantages and disadvantages of these approaches?*

Adaptation of cases is a difficult task that has made difficult the deployment of complete CBR systems. In order to solve this problem, several researchers have proposed to exploit the knowledge that might be already contained in the case-base by means of inductive techniques that elicit knowledge from the case-base and ease the adaptation knowledge bottleneck by learning from cases. For cases represented as feature vectors researchers [161, 164] have used feature differences between pair of cases, to propose adaptation rules [164] or adaptation cases [161]. However, it is not clear from the literature (i) what cases need to be compared to generate adaptation rules and cases and (ii) what rules or cases need to be applied to make the algorithm effective and efficient.

6.1.3.2 Sub-question B

*Can the artificial neural network approach elicit knowledge from the case-base and use it to guide adaptation of cases? If so, what kind of artificial neural network can be used and why? What are the basic assumptions for using this technique; what are the limitations of this approach?*

The artificial neural network approach is a technique that has been used in the product formulation domain to map formulation ingredients or processing conditions to final product properties. Multilayer perceptrons with sigmoidal transfer functions have been the most common ANN used in product formulation applications to demonstrate the ability to learn from examples. Most studies agree in that adaptation (e.g. optimisation) of a formulation is a highly complex non-linear mapping problem because both sets of variables are composed of multiple inputs and outputs. These studies have illustrated that a two-layer (i.e. one hidden layer) feed-forward ANN can approximate the non-linear function between measured input and outputs to optimise formulations. This suggests that MLPs have the potential to capitalise on PU knowledge available in the form of formulations to solve the CBR adaptation problem
Conclusions and Future Work

207

this chapter proposed in section 5.2 p.159. However, the research available in the literature has failed to discuss in detail the issues regarding neural systems design and its implementation in formulating applications.

6.1.3.3 Sub-question C

How can an artificial neural network be implemented to adapt retrieved PU cases when these are represented as feature vectors? How can the knowledge learnt by the artificial neural network be used to adapt retrieved cases? How reliable is the adapted case found by the network?

A feed-forward back-propagation AAN that can adapt retrieved PU cases as well as a method to use the knowledge learnt by the network was implemented. A three-layer feed-forward ANN trained using a modified back-propagation algorithm was found to be able to generate adaptation knowledge from a case-base composed of real PU formulation data. The network was trained to learn a relationship between the change in formulation properties and formulation ingredients. When a new query exists, the CBR system retrieves cases that partially match the query and the difference between the query and the closest retrieved case is presented to the trained network for it to propose what ingredients of the retrieved formulation need to be adjusted to compensate for the differences found.

In order to know how much to change the formulation ingredients, a vote mechanism was proposed. This method worked by taking into account the “weight” or contribution to the answer by defining a relevance factor based on the measure of distance used in the retrieval algorithm in a way that the similar the case to the query is (smaller the distance), the higher the contribution of the case to the answer should be.

Although the network has a low global error (4%), the network does not have good generalisation capabilities. This could be due to the fact that the network was trained with a very small data set and hence it is not able to discover the relationships between the large number of variables involved, which make the problem complex.

In addition, as it was explained in chapter 2, the chemistry of PUs is very sensitive to changes in any PU reactive chemical group (e.g. isocyanate, hydroxyl, amine) produced either by changes in processing conditions (e.g. temperature) or by introduction of chemicals intentionally (i.e. increase in formulation ingredients) or unintentionally (i.e. contamination, environmental humidity changes). A small change (proportional to the PU batch size) in any ingredient for a given formulation is likely to have strong effects on the properties of the foams. Therefore, it is difficult to find a suitable data set to train an ANN. A network likely to learn any mapping between formulation ingredients and mechanical properties would require
Conclusions and Future Work

a vast collection of data covering small variations in the ingredient's quantities. The size of the training data set is also likely to increase as more ingredients are used.

From the PU chemical point of view, certain ingredients have a major impact on the final PU properties. For instance, the catalyst used, and the amount of hydroxyl groups (i.e. coming from water, humidity, polyols, or contaminants) have the highest impact on the stoichiometry of the reaction and hence on the final properties of a foam.

The data sets used for training the ANNs in his thesis, came from a fractional statistical experimental design as explained in chapter 4. In this design of experiments, eight formulation variables were changed simultaneously between the levels suggested by the PU expert to cover the whole gamut of property values. However, it has become apparent that the choice in the change of levels was too gross and did not allow mapping the property-formulation spaces continuously because the changes in properties were widely scattered. This suggests that for such a large number of formulation variables, more data need to be captured at smaller changes within the expert's suggested levels to be able to map the space continuously. In addition, changes in the quantity of reactants depend heavily on the type of reactant used, i.e. a change of 5 g in the amount of catalyst used has a greater effect on the foam final properties than the same change in the amount of the bulk polyol (in for example a 1 kg flexible PU foam formulation batch size). Additionally, some ingredients react and produce reactive by-products, which can at the same time lead to unexpected changes in the foams' final properties. As it is, for example, the case of the addition of water and the production of reactive amine that lead to a subsequent production of urea (refer to p. 16)

6.2 Main Contributions of this Thesis

This thesis has made contributions to both the PU domain as well as to the case-based reasoning community. The original contributions to the advancement of knowledge that have been made in the course of this research project are:

To the polyurethane community:

1. A conceptual framework for the formulation of PU based on historical formulation data.

   The case-based reasoning methodology has proven useful for supporting the formulation of new PU recipes that uses historical formulation data as its case-base and allows retrieving former similar recipes and modifying the retrieved case using an ANN. This framework for PU formulation problem solving also allows a structured capture of PU expertise in the form of cases alleviating the problems stated in chapter 1 regarding knowledge monopolisation by a few experts and its implications in the design of new cost effective PU products.
To the case-based reasoning community:

2. A novel alternative estimation of similarity between objects (e.g. PU cases) based on moment descriptors of shape representations of cases described as vectors.

3. A computational mechanism for the use of the ANN approach as a tool to elicit knowledge from the case-base (i.e. composed of historical PU formulation data) that can be further employed to overcome the CBR adaptation knowledge bottleneck. In this thesis, the adaptation problem regards the adjustment of ingredients and their quantities of PU complex foam formulations.

Specifically, the implementation of this knowledge light approach for the adaptation of cases integrates the concept of ANN learning with a vote mechanism that uses the knowledge of how close are the retrieved cases to the query. It was demonstrated that ANN are only able to memorise the mechanical property-formulation knowledge rather than learn, free of generalisation issues, a mapping between the two. Although ANN is a robust technique that has been used in applications which are complex and noisy, it was demonstrated that they are sensitive to the selection of the PU formulation examples.

6.3 Future Work

The approach outlined in this study can be extended in various promising directions if the lessons learnt are taken into account. The future work activities can be categorised in terms of those that relate to the application domain and those that relate to the retrieval algorithm and the machine learner.

6.3.1 Application Domain

6.3.1.1 Industrial implementation and validation

Most of the work done in this thesis has used a scarce laboratory data set provided by a major PU manufacturer viz. BASF. The extension of this work in an industrial setting could capitalise on the historical formulation data that most manufacturers can have access to, in order to support PU experts in formulating new PUs under the CBR-ANN framework described in this thesis. The implementation of this problem solving framework requires taking into account the issues regarding adequate coverage of the data the domain is being described with.

For instance, the formulation data and hence the description of cases could be complemented with processing information (e.g. mould temperature, mixing time) and chemical data (e.g. %NCO in the formulation, molecular weight of ingredients). By enriching
Conclusions and Future Work

the cases with this information, retrieval of cases tailored from different views could be performed and evaluated. In addition, it is expected that if the number of formulations is larger, which can be expected in an R&D department in an industrial setting, improvements in the ANN approach can be made. For example, the results from chapter 5 suggested that the main drawback in developing a predictive tool (i.e. able to generalise) that could map mechanical properties to formulation ingredients was the scarcity of data. The cross-validation approach failed to found any network that could learn from the training data presented to the network. Therefore, not only a large data set but careful selection of it is required.

Further research is encouraged in order to put into practice the CBR problem solving framework. Experiments that involve manufacturing and testing of predicted formulations generated by the CBR tool can be carried out in order to meet target properties. The validation experimental of the tool will shed some light on the improvement of the machine learner (i.e. ANN) and the definition of the PU variables (e.g. formulation ingredients, chemical nature of ingredients, final properties and processing conditions) that can be taken into account in an industrial setting.

6.3.1.2 Greater scope

This thesis has shown that AI technologies such as hybrid CBR-NN computational techniques can support the problems that make the PU formulation problem a complex task and hence lead to reduced formulation time and costs when compared to traditional approaches to formulation. For this reason, it would be useful to use the CBR-ANN framework system to support other PU applications such as rigid foams or adhesives.

6.3.2 Case-Based Reasoning-Artificial Neural Network System

6.3.2.1 Experimental Validation of the Retrieval Algorithm

This research has investigated an alternative similarity measure based on the moment’s description of a case when this can represented as a two dimensional image. A comparison between the standard measures of similarity and the image’s moments based approach was discussed for the retrieval of PU foam formulations using the k-means method of cluster analysis. The reason behind the use of a cluster analysis is due to the fact that there were no probes or test cases i.e. cases with the “correct” formulation available that enabled to assess retrieval in terms of which algorithm performs better. Therefore, there is a need to collect further test cases in order to compare each algorithm in terms of which one gives a better result (i.e. which is more accurate when compared to a test case).
Conclusions and Future Work

In addition, the global similarity measure developed in this thesis, which is based on a case's moment descriptors, used cases represented as numerical-valued feature vectors. It would be useful to integrate the use of qualitative attributes into the retrieval algorithm by either transforming the qualitative variables into quantitative ones or by using a fuzzy approach and evaluating its performance. Moreover, it is being demonstrated by cognitive sciences that human beings are reasoning based on cases and similarities in many ways. Thus, in the author's belief, there should be more sophisticated and flexible approaches to similarity that require experimental and theoretical investigations. Therefore, further experiments need to be carried out to try to exploit the moment-based similarity approach in diverse domains.

6.3.2.2 The Use of Inductive techniques

Classification-rule learning algorithms can be used to find rules or decision trees that partition given data into predefined classes. These predefined classes could be for example used to index the case-base, further optimising retrieval.

6.3.2.3 Producing Evidence for the Hypotheses of the Artificial Neural Networks

There are several different methods of describing the output function of a neural network. For example, (i) through direct experimental validation or (ii) by the interpretation of neural networks as rule based systems such as the early approaches studied by Fletcher and Hinde [212, 213].

This research has explored the use of an ANN architecture capable of using PU formulation data to generate adaptation knowledge to propose changes in formulation ingredients of retrieved cases to match a query problem. However, there is a need to assess other ideal network capabilities such as (i) the network's sensitivity to noise and errors in data entry or to changes due to introduction of new formulation in the data set (ii) network's optimality in adjusting outcomes under confounding influences (e.g. under correlated data) (iii) greater scope i.e. improved generalisation capabilities.

6.3.2.4 Other Network Architectures

Other neural network systems can be studied for modelling the difference between mechanical properties as a function of formulation ingredients. Recently, Bayesian neural networks have been used to enable prediction of mechanical behaviour of powder metal parts as a function of processing conditions, indicating a level of confidence in the result [214].
6.3.2.5 Extended Capabilities

This thesis has investigated how desired PU final properties can be obtained by changes in formulation (predictive function). However, the CBR system capabilities could be extended to provide information of how changes in formulation affect product properties (deductive function). This can be implemented by for instance retraining the neural network to learn a different mapping i.e. formulation ingredients to property changes.
References


31. BAYER, Baypreg® F - Polyurethane plus natural materials in cars. 2000, Bayer AG.
47. Personal Communication with Heath, R. Loughborough University, Loughborough, UK, 2005


154. Interview with Miguel Quintero Bogota, Colombia, 2005


194. Wilke, W. and R. Bergmann. Techniques and Knowledge Used for Adaptation During Case-Based Problem Solving. in 11th International Conference on Industrial and Engineering


211. Interview with Gary Jackson Alfreton Industrial State, Derbyshire, UK, 2003


Appendix I. Manufacturing of Foams in the laboratory

This appendix collects detailed information for the statistical design and results of the foam testing carried out for both case studies. This information is presented in the following order:

CASE STUDY 1
Statistical Design
Table 6-1. Factor description and levels
Table 6-2. Experimental design with 18 observations.
Testing
Table 6-3. Measured Foam Physical Properties.

CASE STUDY 2
Statistical Design
Table 6-4. $2^{8-3}$ Fractional Factorial Design for the CMHR foam
Testing
Table 6-5. Foams’ reactivity profile. Average of two values reported
Table 6-6. Measured Foam Physical Properties. Average of three values reported

1. CASE STUDY 1. Water-Blown Integral Self-Skinning Flexible Polyurethane Foams
   a) Statistical Design

<table>
<thead>
<tr>
<th>Factor</th>
<th>Number of levels</th>
<th>Description of the levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polyol A / Polyol C ratio</td>
<td>3</td>
<td>2:1 ratio 1:1 ratio 1:2 ratio</td>
</tr>
<tr>
<td>Polyol D / Polyol E absence</td>
<td>3</td>
<td>Polyol D and Polyol E in equal parts Polyol D, no Polyol E Polyol E, no Polyol D</td>
</tr>
<tr>
<td>Isocyanate type</td>
<td>2</td>
<td>Isocyanate 1 Isocyanate 2</td>
</tr>
<tr>
<td>Index</td>
<td>0.5</td>
<td>Average equivalent weight of polyol side was kept constant</td>
</tr>
</tbody>
</table>
Table 6-2. Experimental design with 18 observations.

<table>
<thead>
<tr>
<th>Trial</th>
<th>Factor 1</th>
<th>Factor 2</th>
<th>Factor 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>
</tbody>
</table>
b) Testing

Table 6-3. Measured Foam Physical Properties.

After 24h conditioning at room temperature of the foam samples, six variables were measured. Averages of three repetitions are reported. This table is also referred to as the Problem-Case in which trial corresponds to a case composed of the six numerical attributes.

<table>
<thead>
<tr>
<th>Trial /Case</th>
<th>Density [kg/m³]</th>
<th>Tensile Strength [MPa]</th>
<th>Elongation [%]</th>
<th>Tear Strength [N/mm]</th>
<th>Hardness 1s [Shore A]</th>
<th>Hardness 30s [Shore A]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Average 280.62</td>
<td>3.14</td>
<td>77.20</td>
<td>0.73</td>
<td>18.33</td>
<td>7.00</td>
</tr>
<tr>
<td>Std Dev</td>
<td>4.22</td>
<td>0.37</td>
<td>7.08</td>
<td>0.05</td>
<td>0.58</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>Average 290.63</td>
<td>3.53</td>
<td>90.10</td>
<td>0.84</td>
<td>22.67</td>
<td>7.83</td>
</tr>
<tr>
<td>Std Dev</td>
<td>10.44</td>
<td>0.15</td>
<td>6.17</td>
<td>0.06</td>
<td>0.58</td>
<td>0.58</td>
</tr>
<tr>
<td>3</td>
<td>Average 283.07</td>
<td>3.36</td>
<td>64.17</td>
<td>0.83</td>
<td>24.00</td>
<td>9.00</td>
</tr>
<tr>
<td>Std Dev</td>
<td>3.08</td>
<td>0.28</td>
<td>34.62</td>
<td>0.08</td>
<td>1.73</td>
<td>2.00</td>
</tr>
<tr>
<td>4</td>
<td>Average 269.38</td>
<td>2.87</td>
<td>67.47</td>
<td>0.69</td>
<td>22.83</td>
<td>8.83</td>
</tr>
<tr>
<td>Std Dev</td>
<td>1.97</td>
<td>0.50</td>
<td>22.75</td>
<td>0.03</td>
<td>0.76</td>
<td>0.76</td>
</tr>
<tr>
<td>5</td>
<td>Average 292.78</td>
<td>3.91</td>
<td>90.30</td>
<td>0.86</td>
<td>28.00</td>
<td>11.00</td>
</tr>
<tr>
<td>Std Dev</td>
<td>6.85</td>
<td>0.26</td>
<td>3.28</td>
<td>0.04</td>
<td>1.73</td>
<td>1.73</td>
</tr>
<tr>
<td>6</td>
<td>Average 283.76</td>
<td>3.41</td>
<td>76.77</td>
<td>0.73</td>
<td>26.50</td>
<td>12.67</td>
</tr>
<tr>
<td>Std Dev</td>
<td>13.26</td>
<td>0.12</td>
<td>7.54</td>
<td>0.11</td>
<td>1.32</td>
<td>1.04</td>
</tr>
<tr>
<td>7</td>
<td>Average 245.71</td>
<td>3.08</td>
<td>65.30</td>
<td>0.62</td>
<td>24.83</td>
<td>13.17</td>
</tr>
<tr>
<td>Std Dev</td>
<td>1.83</td>
<td>0.61</td>
<td>10.67</td>
<td>0.08</td>
<td>0.76</td>
<td>1.26</td>
</tr>
<tr>
<td>8</td>
<td>Average 244.14</td>
<td>3.14</td>
<td>64.27</td>
<td>0.58</td>
<td>24.50</td>
<td>13.50</td>
</tr>
<tr>
<td>Std Dev</td>
<td>2.75</td>
<td>0.27</td>
<td>1.76</td>
<td>0.03</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>9</td>
<td>Average 242.02</td>
<td>3.14</td>
<td>61.73</td>
<td>0.61</td>
<td>23.00</td>
<td>12.50</td>
</tr>
<tr>
<td>Std Dev</td>
<td>2.22</td>
<td>0.24</td>
<td>9.80</td>
<td>0.01</td>
<td>0.00</td>
<td>0.50</td>
</tr>
<tr>
<td>10</td>
<td>Average 270.79</td>
<td>0.59</td>
<td>32.53</td>
<td>0.63</td>
<td>38.83</td>
<td>28.67</td>
</tr>
<tr>
<td>Std Dev</td>
<td>11.85</td>
<td>0.11</td>
<td>10.02</td>
<td>0.07</td>
<td>1.04</td>
<td>1.26</td>
</tr>
<tr>
<td>11</td>
<td>Average 259.85</td>
<td>0.63</td>
<td>41.00</td>
<td>0.58</td>
<td>39.83</td>
<td>28.00</td>
</tr>
<tr>
<td>Std Dev</td>
<td>3.10</td>
<td>0.06</td>
<td>2.98</td>
<td>0.08</td>
<td>1.89</td>
<td>1.73</td>
</tr>
<tr>
<td>12</td>
<td>Average 258.36</td>
<td>0.59</td>
<td>38.97</td>
<td>0.61</td>
<td>40.00</td>
<td>29.67</td>
</tr>
<tr>
<td>Std Dev</td>
<td>4.60</td>
<td>0.04</td>
<td>3.27</td>
<td>0.06</td>
<td>1.00</td>
<td>0.29</td>
</tr>
<tr>
<td>13</td>
<td>Average 255.32</td>
<td>0.56</td>
<td>36.53</td>
<td>0.53</td>
<td>40.67</td>
<td>29.33</td>
</tr>
<tr>
<td>Std Dev</td>
<td>8.32</td>
<td>0.08</td>
<td>5.09</td>
<td>0.05</td>
<td>0.58</td>
<td>0.58</td>
</tr>
<tr>
<td>14</td>
<td>Average 263.88</td>
<td>0.60</td>
<td>37.27</td>
<td>0.63</td>
<td>41.67</td>
<td>30.33</td>
</tr>
<tr>
<td>Std Dev</td>
<td>6.78</td>
<td>0.09</td>
<td>1.01</td>
<td>0.08</td>
<td>0.58</td>
<td>1.53</td>
</tr>
<tr>
<td>15</td>
<td>Average 256.58</td>
<td>0.55</td>
<td>35.27</td>
<td>0.57</td>
<td>41.00</td>
<td>30.50</td>
</tr>
<tr>
<td>Std Dev</td>
<td>1.56</td>
<td>0.04</td>
<td>4.57</td>
<td>0.09</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>16</td>
<td>Average 229.35</td>
<td>0.55</td>
<td>31.97</td>
<td>0.59</td>
<td>42.00</td>
<td>31.17</td>
</tr>
<tr>
<td>Std Dev</td>
<td>0.98</td>
<td>0.04</td>
<td>1.11</td>
<td>0.08</td>
<td>0.00</td>
<td>2.02</td>
</tr>
<tr>
<td>17</td>
<td>Average 231.18</td>
<td>0.62</td>
<td>35.83</td>
<td>0.62</td>
<td>41.67</td>
<td>29.83</td>
</tr>
<tr>
<td>Std Dev</td>
<td>3.22</td>
<td>0.01</td>
<td>2.40</td>
<td>0.06</td>
<td>0.58</td>
<td>1.04</td>
</tr>
<tr>
<td>18</td>
<td>Average 225.03</td>
<td>0.58</td>
<td>35.33</td>
<td>0.55</td>
<td>41.50</td>
<td>29.67</td>
</tr>
<tr>
<td>Std Dev</td>
<td>5.43</td>
<td>0.01</td>
<td>0.58</td>
<td>0.06</td>
<td>0.87</td>
<td>1.15</td>
</tr>
</tbody>
</table>

2. CASE STUDY 2. Combustion Modified High Resilient Foam
### a) Statistical Design

Table 6-4.2 (8-3) Fractional Factorial Design for the CMHR foam

<table>
<thead>
<tr>
<th>Run</th>
<th>Polyether Polyol 1</th>
<th>Polyether Polyol 2</th>
<th>Amine Crosslinker</th>
<th>Amine Catalyst 1</th>
<th>Amine Catalyst 2</th>
<th>Surfactant</th>
<th>Water</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>91.95</td>
<td>1.5</td>
<td>1.5</td>
<td>0.3</td>
<td>0.75</td>
<td>0.25</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>2</td>
<td>92.7</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
<td>0.75</td>
<td>0.25</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>3</td>
<td>93.35</td>
<td>1.5</td>
<td>0</td>
<td>0.9</td>
<td>0.25</td>
<td>0.25</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>4</td>
<td>89.85</td>
<td>3</td>
<td>1.5</td>
<td>0.9</td>
<td>0.25</td>
<td>0.75</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>5</td>
<td>93.6</td>
<td>1.5</td>
<td>0</td>
<td>0.9</td>
<td>0.25</td>
<td>0.75</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>6</td>
<td>91.35</td>
<td>1.5</td>
<td>1.5</td>
<td>0.9</td>
<td>0.25</td>
<td>0.75</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>7</td>
<td>91.35</td>
<td>3</td>
<td>0</td>
<td>0.9</td>
<td>0.75</td>
<td>0.25</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>8</td>
<td>91.95</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
<td>0.25</td>
<td>0.75</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>9</td>
<td>91.2</td>
<td>3</td>
<td>1.5</td>
<td>0.3</td>
<td>0.25</td>
<td>0.75</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>10</td>
<td>90.45</td>
<td>3</td>
<td>1.5</td>
<td>0.3</td>
<td>0.25</td>
<td>0.25</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>11</td>
<td>90.95</td>
<td>3</td>
<td>1.5</td>
<td>0.3</td>
<td>0.25</td>
<td>0.75</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>12</td>
<td>90.85</td>
<td>1.5</td>
<td>1.5</td>
<td>0.9</td>
<td>0.75</td>
<td>0.75</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>13</td>
<td>92.95</td>
<td>1.5</td>
<td>0</td>
<td>0.9</td>
<td>0.75</td>
<td>0.75</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>14</td>
<td>91.25</td>
<td>2.5</td>
<td>1</td>
<td>0.7</td>
<td>0.5</td>
<td>0.25</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>15</td>
<td>91.1</td>
<td>3</td>
<td>1.5</td>
<td>0.9</td>
<td>0.25</td>
<td>0.25</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>16</td>
<td>89.35</td>
<td>3</td>
<td>1.5</td>
<td>0.9</td>
<td>0.75</td>
<td>0.75</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>17</td>
<td>93.2</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
<td>0.25</td>
<td>0.25</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>18</td>
<td>92.7</td>
<td>1.5</td>
<td>1.5</td>
<td>0.3</td>
<td>0.25</td>
<td>0.75</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>19</td>
<td>92.6</td>
<td>1.5</td>
<td>1.5</td>
<td>0.9</td>
<td>0.25</td>
<td>0.25</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>20</td>
<td>93.1</td>
<td>1.5</td>
<td>0</td>
<td>0.9</td>
<td>0.75</td>
<td>0.75</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>21</td>
<td>92.1</td>
<td>1.5</td>
<td>1.5</td>
<td>0.9</td>
<td>0.75</td>
<td>0.25</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>22</td>
<td>94.2</td>
<td>1.5</td>
<td>0</td>
<td>0.3</td>
<td>0.75</td>
<td>0.25</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>23</td>
<td>94.7</td>
<td>1.5</td>
<td>0</td>
<td>0.3</td>
<td>0.25</td>
<td>0.25</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>24</td>
<td>92.45</td>
<td>1.5</td>
<td>1.5</td>
<td>0.3</td>
<td>0.25</td>
<td>0.25</td>
<td>3.75</td>
<td>85</td>
</tr>
<tr>
<td>25</td>
<td>90.7</td>
<td>3</td>
<td>1.5</td>
<td>0.3</td>
<td>0.75</td>
<td>0.75</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>26</td>
<td>91.25</td>
<td>2.5</td>
<td>1</td>
<td>0.7</td>
<td>0.5</td>
<td>0.25</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>27</td>
<td>91.85</td>
<td>3</td>
<td>0</td>
<td>0.9</td>
<td>0.25</td>
<td>0.25</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>28</td>
<td>92.85</td>
<td>1.5</td>
<td>0</td>
<td>0.9</td>
<td>0.75</td>
<td>0.25</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>29</td>
<td>91.6</td>
<td>3</td>
<td>0</td>
<td>0.9</td>
<td>0.75</td>
<td>0.75</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td>30</td>
<td>92.2</td>
<td>1.5</td>
<td>1.5</td>
<td>0.3</td>
<td>0.75</td>
<td>0.75</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>31</td>
<td>91.45</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
<td>0.75</td>
<td>0.75</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>32</td>
<td>93.45</td>
<td>1.5</td>
<td>0</td>
<td>0.3</td>
<td>0.25</td>
<td>0.75</td>
<td>3.75</td>
<td>110</td>
</tr>
<tr>
<td>33</td>
<td>90.6</td>
<td>3</td>
<td>1.5</td>
<td>0.9</td>
<td>0.75</td>
<td>0.25</td>
<td>3</td>
<td>110</td>
</tr>
<tr>
<td>34</td>
<td>92.1</td>
<td>3</td>
<td>0</td>
<td>0.9</td>
<td>0.25</td>
<td>0.75</td>
<td>3</td>
<td>110</td>
</tr>
</tbody>
</table>
### Appendix I. Manufacturing of Foams in the Laboratory

b) Testing

Table 6-5. Foams' reactivity profile. Average of two values reported

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>50:34.5</td>
<td>19</td>
<td>113</td>
<td>39.1</td>
</tr>
<tr>
<td>2</td>
<td>50:26.7</td>
<td>17</td>
<td>104</td>
<td>44.4</td>
</tr>
<tr>
<td>3</td>
<td>50:25.0</td>
<td>11</td>
<td>75</td>
<td>41.9</td>
</tr>
<tr>
<td>4</td>
<td>50:26.7</td>
<td>13</td>
<td>51</td>
<td>Collapsed</td>
</tr>
<tr>
<td>5</td>
<td>50:20.5</td>
<td>12</td>
<td>94</td>
<td>45</td>
</tr>
<tr>
<td>6</td>
<td>50:34.5</td>
<td>15</td>
<td>60</td>
<td>Collapsed</td>
</tr>
<tr>
<td>7</td>
<td>50:25.0</td>
<td>8</td>
<td>54</td>
<td>38.7</td>
</tr>
<tr>
<td>8</td>
<td>50:25.1</td>
<td>24</td>
<td>154</td>
<td>53.8</td>
</tr>
<tr>
<td>9</td>
<td>50:28.7</td>
<td>20</td>
<td>188</td>
<td>47.5</td>
</tr>
<tr>
<td>10</td>
<td>50:26.7</td>
<td>17</td>
<td>105</td>
<td>39.95</td>
</tr>
<tr>
<td>11</td>
<td>50:34.6</td>
<td>26</td>
<td>113</td>
<td>Collapsed</td>
</tr>
<tr>
<td>12</td>
<td>50:26.6</td>
<td>9</td>
<td>64</td>
<td>37.35</td>
</tr>
<tr>
<td>13</td>
<td>50:25.0</td>
<td>16</td>
<td>119</td>
<td>37.5</td>
</tr>
<tr>
<td>14</td>
<td>50:29.3</td>
<td>14</td>
<td>80</td>
<td>43.2</td>
</tr>
<tr>
<td>15</td>
<td>50:22.2</td>
<td>12</td>
<td>79</td>
<td>46</td>
</tr>
<tr>
<td>16</td>
<td>50:34.5</td>
<td>13</td>
<td>44</td>
<td>Collapsed</td>
</tr>
<tr>
<td>17</td>
<td>50:20.6</td>
<td>22</td>
<td>190</td>
<td>48.35</td>
</tr>
<tr>
<td>18</td>
<td>50:22.1</td>
<td>21</td>
<td>183</td>
<td>49.45</td>
</tr>
<tr>
<td>19</td>
<td>50:28.7</td>
<td>14</td>
<td>100</td>
<td>43.6</td>
</tr>
<tr>
<td>20</td>
<td>50:26.6</td>
<td>7</td>
<td>70</td>
<td>38.7</td>
</tr>
<tr>
<td>21</td>
<td>50:22.1</td>
<td>7</td>
<td>56</td>
<td>43.9</td>
</tr>
<tr>
<td>22</td>
<td>50:20.6</td>
<td>12</td>
<td>110</td>
<td>45.1</td>
</tr>
<tr>
<td>23</td>
<td>50:26.7</td>
<td>26</td>
<td>191</td>
<td>50.5</td>
</tr>
<tr>
<td>24</td>
<td>50:26.7</td>
<td>24</td>
<td>165</td>
<td>43.1</td>
</tr>
<tr>
<td>25</td>
<td>50:22.2</td>
<td>18</td>
<td>97</td>
<td>45.7</td>
</tr>
<tr>
<td>26</td>
<td>50:29.3</td>
<td>11</td>
<td>82</td>
<td>37.75</td>
</tr>
<tr>
<td>27</td>
<td>50:32.4</td>
<td>11</td>
<td>61</td>
<td>Collapsed</td>
</tr>
<tr>
<td>28</td>
<td>50:32.4</td>
<td>8</td>
<td>67</td>
<td>45.5</td>
</tr>
<tr>
<td>29</td>
<td>50:20.5</td>
<td>9</td>
<td>61</td>
<td>42.45</td>
</tr>
<tr>
<td>30</td>
<td>50:28.6</td>
<td>17</td>
<td>122</td>
<td>39.9</td>
</tr>
<tr>
<td>31</td>
<td>50:32.4</td>
<td>18</td>
<td>101</td>
<td>44.6</td>
</tr>
<tr>
<td>32</td>
<td>50:32.4</td>
<td>23</td>
<td>144</td>
<td>55.85</td>
</tr>
<tr>
<td>33</td>
<td>50:28.7</td>
<td>12</td>
<td>62</td>
<td>40.9</td>
</tr>
<tr>
<td>34</td>
<td>50:26.7</td>
<td>11</td>
<td>82</td>
<td>42.2</td>
</tr>
</tbody>
</table>
Table 6-6. Measured Foam Physical Properties. Average of three values reported

<table>
<thead>
<tr>
<th>Run</th>
<th>Hardness [N]</th>
<th>Density [kg/m³]</th>
<th>Tensile Strength [kPa]</th>
<th>Elongation [%]</th>
<th>Compression Set [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>445</td>
<td>54.1</td>
<td>134</td>
<td>73</td>
<td>10.1</td>
</tr>
<tr>
<td>2</td>
<td>240</td>
<td>48.2</td>
<td>86</td>
<td>83</td>
<td>7.3</td>
</tr>
<tr>
<td>3</td>
<td>384</td>
<td>67.6</td>
<td>137</td>
<td>108</td>
<td>12.6</td>
</tr>
<tr>
<td>4</td>
<td>177</td>
<td>47.6</td>
<td>74</td>
<td>83</td>
<td>5.8</td>
</tr>
<tr>
<td>5</td>
<td>141</td>
<td>59.3</td>
<td>62</td>
<td>106</td>
<td>7.7</td>
</tr>
<tr>
<td>6</td>
<td>454</td>
<td>53.1</td>
<td>148</td>
<td>79</td>
<td>8.4</td>
</tr>
<tr>
<td>7</td>
<td>152</td>
<td>47.1</td>
<td>67</td>
<td>89</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>235</td>
<td>51.2</td>
<td>64</td>
<td>81</td>
<td>6.8</td>
</tr>
<tr>
<td>9</td>
<td>192</td>
<td>50.8</td>
<td>50</td>
<td>62</td>
<td>4.9</td>
</tr>
<tr>
<td>10</td>
<td>168</td>
<td>48.9</td>
<td>56</td>
<td>75</td>
<td>6</td>
</tr>
<tr>
<td>11</td>
<td>431</td>
<td>53.5</td>
<td>90</td>
<td>59</td>
<td>7.7</td>
</tr>
<tr>
<td>12</td>
<td>154</td>
<td>48.9</td>
<td>61</td>
<td>80</td>
<td>7.5</td>
</tr>
<tr>
<td>13</td>
<td>149</td>
<td>48.2</td>
<td>60</td>
<td>83</td>
<td>8.6</td>
</tr>
<tr>
<td>14</td>
<td>230</td>
<td>51.1</td>
<td>86</td>
<td>81</td>
<td>3.9</td>
</tr>
<tr>
<td>15</td>
<td>171</td>
<td>61.2</td>
<td>65</td>
<td>86</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>354</td>
<td>55.0</td>
<td>114</td>
<td>70</td>
<td>4.6</td>
</tr>
<tr>
<td>17</td>
<td>161</td>
<td>59.1</td>
<td>55</td>
<td>87</td>
<td>4.6</td>
</tr>
<tr>
<td>18</td>
<td>175</td>
<td>60.1</td>
<td>74</td>
<td>92</td>
<td>5.4</td>
</tr>
<tr>
<td>19</td>
<td>210</td>
<td>51.3</td>
<td>93</td>
<td>87</td>
<td>6.9</td>
</tr>
<tr>
<td>20</td>
<td>153</td>
<td>48.3</td>
<td>66</td>
<td>81</td>
<td>9.1</td>
</tr>
<tr>
<td>21</td>
<td>159</td>
<td>61.1</td>
<td>76</td>
<td>101</td>
<td>6.1</td>
</tr>
<tr>
<td>22</td>
<td>129</td>
<td>58.6</td>
<td>51</td>
<td>95</td>
<td>5.6</td>
</tr>
<tr>
<td>23</td>
<td>182</td>
<td>50.3</td>
<td>67</td>
<td>87</td>
<td>4.9</td>
</tr>
<tr>
<td>24</td>
<td>301</td>
<td>58.6</td>
<td>82</td>
<td>75</td>
<td>7.2</td>
</tr>
<tr>
<td>25</td>
<td>156</td>
<td>60.7</td>
<td>57</td>
<td>85</td>
<td>5.3</td>
</tr>
<tr>
<td>26</td>
<td>289</td>
<td>51.5</td>
<td>104</td>
<td>88</td>
<td>5.3</td>
</tr>
<tr>
<td>27</td>
<td>469</td>
<td>53.6</td>
<td>140</td>
<td>82</td>
<td>7.6</td>
</tr>
<tr>
<td>28</td>
<td>477</td>
<td>51.8</td>
<td>143</td>
<td>81</td>
<td>8.9</td>
</tr>
<tr>
<td>29</td>
<td>144</td>
<td>58.4</td>
<td>70</td>
<td>115</td>
<td>5.2</td>
</tr>
<tr>
<td>30</td>
<td>275</td>
<td>50.5</td>
<td>113</td>
<td>90</td>
<td>4.4</td>
</tr>
<tr>
<td>31</td>
<td>471</td>
<td>52.7</td>
<td>158</td>
<td>86</td>
<td>4.1</td>
</tr>
<tr>
<td>32</td>
<td>708</td>
<td>62.7</td>
<td>217</td>
<td>96</td>
<td>9.2</td>
</tr>
<tr>
<td>33</td>
<td>273</td>
<td>49.8</td>
<td>85</td>
<td>79</td>
<td>9.7</td>
</tr>
<tr>
<td>34</td>
<td>292</td>
<td>50.4</td>
<td>105</td>
<td>83</td>
<td>8.6</td>
</tr>
</tbody>
</table>
### Table 6-7. Measured Foam Physical Properties. Average of three values reported

<table>
<thead>
<tr>
<th>Run</th>
<th>Hardness [N]</th>
<th>Density [kg/m³]</th>
<th>Tensile Strength [kPa]</th>
<th>Elongation [%]</th>
<th>Compression Set [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>445</td>
<td>54.1</td>
<td>134</td>
<td>73</td>
<td>10.1</td>
</tr>
<tr>
<td>2</td>
<td>240</td>
<td>48.2</td>
<td>86</td>
<td>83</td>
<td>7.3</td>
</tr>
<tr>
<td>3</td>
<td>384</td>
<td>67.6</td>
<td>137</td>
<td>108</td>
<td>12.6</td>
</tr>
<tr>
<td>4</td>
<td>177</td>
<td>47.6</td>
<td>74</td>
<td>83</td>
<td>5.8</td>
</tr>
<tr>
<td>5</td>
<td>141</td>
<td>59.3</td>
<td>62</td>
<td>106</td>
<td>7.7</td>
</tr>
<tr>
<td>6</td>
<td>454</td>
<td>53.1</td>
<td>148</td>
<td>79</td>
<td>8.4</td>
</tr>
<tr>
<td>7</td>
<td>152</td>
<td>47.1</td>
<td>67</td>
<td>89</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>235</td>
<td>51.2</td>
<td>64</td>
<td>81</td>
<td>6.8</td>
</tr>
<tr>
<td>9</td>
<td>192</td>
<td>50.8</td>
<td>50</td>
<td>62</td>
<td>4.9</td>
</tr>
<tr>
<td>10</td>
<td>168</td>
<td>48.9</td>
<td>56</td>
<td>75</td>
<td>6</td>
</tr>
<tr>
<td>11</td>
<td>431</td>
<td>53.5</td>
<td>90</td>
<td>59</td>
<td>7.7</td>
</tr>
<tr>
<td>12</td>
<td>154</td>
<td>48.9</td>
<td>61</td>
<td>80</td>
<td>7.5</td>
</tr>
<tr>
<td>13</td>
<td>149</td>
<td>48.2</td>
<td>60</td>
<td>83</td>
<td>8.6</td>
</tr>
<tr>
<td>14</td>
<td>230</td>
<td>51.1</td>
<td>86</td>
<td>81</td>
<td>3.9</td>
</tr>
<tr>
<td>15</td>
<td>171</td>
<td>61.2</td>
<td>65</td>
<td>86</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>354</td>
<td>55.0</td>
<td>114</td>
<td>70</td>
<td>4.6</td>
</tr>
<tr>
<td>17</td>
<td>161</td>
<td>59.1</td>
<td>55</td>
<td>87</td>
<td>4.6</td>
</tr>
<tr>
<td>18</td>
<td>175</td>
<td>60.1</td>
<td>74</td>
<td>92</td>
<td>5.4</td>
</tr>
<tr>
<td>19</td>
<td>210</td>
<td>51.3</td>
<td>93</td>
<td>87</td>
<td>6.9</td>
</tr>
<tr>
<td>20</td>
<td>153</td>
<td>48.3</td>
<td>66</td>
<td>81</td>
<td>9.1</td>
</tr>
<tr>
<td>21</td>
<td>159</td>
<td>61.1</td>
<td>76</td>
<td>101</td>
<td>6.1</td>
</tr>
<tr>
<td>22</td>
<td>129</td>
<td>58.6</td>
<td>51</td>
<td>95</td>
<td>5.6</td>
</tr>
<tr>
<td>23</td>
<td>182</td>
<td>50.3</td>
<td>67</td>
<td>87</td>
<td>4.9</td>
</tr>
<tr>
<td>24</td>
<td>301</td>
<td>58.6</td>
<td>82</td>
<td>75</td>
<td>7.2</td>
</tr>
<tr>
<td>25</td>
<td>156</td>
<td>60.7</td>
<td>57</td>
<td>85</td>
<td>5.3</td>
</tr>
<tr>
<td>26</td>
<td>289</td>
<td>51.5</td>
<td>104</td>
<td>88</td>
<td>5.3</td>
</tr>
<tr>
<td>27</td>
<td>469</td>
<td>53.6</td>
<td>140</td>
<td>82</td>
<td>7.6</td>
</tr>
<tr>
<td>28</td>
<td>477</td>
<td>51.8</td>
<td>143</td>
<td>81</td>
<td>8.9</td>
</tr>
<tr>
<td>29</td>
<td>144</td>
<td>58.4</td>
<td>70</td>
<td>115</td>
<td>5.2</td>
</tr>
<tr>
<td>30</td>
<td>275</td>
<td>50.5</td>
<td>113</td>
<td>90</td>
<td>4.4</td>
</tr>
<tr>
<td>31</td>
<td>471</td>
<td>52.7</td>
<td>158</td>
<td>86</td>
<td>4.1</td>
</tr>
<tr>
<td>32</td>
<td>708</td>
<td>62.7</td>
<td>217</td>
<td>96</td>
<td>9.2</td>
</tr>
<tr>
<td>33</td>
<td>273</td>
<td>49.8</td>
<td>85</td>
<td>79</td>
<td>9.7</td>
</tr>
<tr>
<td>34</td>
<td>292</td>
<td>50.4</td>
<td>105</td>
<td>83</td>
<td>8.6</td>
</tr>
</tbody>
</table>