A multi-agent systems approach to construction claims negotiation

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/2134/6813

Publisher: © Zhaomin Ren

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.

![Creative Commons Licence](https://creativecommons.org/licenses/by-nc-nd/2.5/)

**Attribution-NonCommercial-NoDerivs 2.5**

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

**Attribution.** You must attribute the work in the manner specified by the author or licensor.

**Noncommercial.** You may not use this work for commercial purposes.

**No Derivative Works.** You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code (the full licence)](http://creativecommons.org/licenses/by-nc-nd/2.5/).

Disclaimer

For the full text of this licence, please go to: [http://creativecommons.org/licenses/by-nc-nd/2.5/](http://creativecommons.org/licenses/by-nc-nd/2.5/)
A MULTI-AGENT SYSTEMS APPROACH TO CONSTRUCTION CLAIMS NEGOTIATION

ZHAOMIN REN

A thesis submitted in partial fulfilment of the requirements
of Loughborough University
for degree of doctor of philosophy

March 2002
ABSTRACT

Claims negotiation plays an important role in construction claims settlement and disputes resolution. However, claims negotiations are normally conducted inefficiently. Although many research projects have been undertaken on human behaviours in negotiation (e.g. negotiation planning, documentation and negotiation strategies) and computer-aided negotiation, there is not an effective approach to solving such problem. The development of multi-agent systems provides an innovative approach to facilitating claims negotiation, where intelligent agents can negotiate with each other for the real world parties that they represent. The significance of multi-agent systems lies in the fact that they match the fragmented nature of the construction industry.

This thesis describes the work of developing a multi-agent system for construction claims negotiation (MASCOT). The objectives are to create an architecture for the agent system, and develop a negotiation mechanism for agent interaction. A conceptual MASCOT model is designed based on a thorough analysis of the nature, characteristics and problems of construction claims negotiation, multi-agent systems negotiation mechanisms, and negotiation theories. A modified Monotonic Concession protocol and the related negotiation strategies which are based on the integration of Zeuthen’s risk evaluation model and Bayesian learning model were developed. A prototype was built using the ZEUS agent building toolkit and Java.

The system was then assessed in terms of the quality of the negotiation mechanism and prototype using critical evaluation criteria and prototype evaluation. The result revealed that the MASCOT system could significantly enhance the efficiency of construction claims negotiation. Furthermore, it is recognised that multi-agent systems have a great potential to solve the fragmentation problem in other construction areas such as scheduling, concurrent engineering; and collaborative design, particularly when the project team members are geographically distributed. This research not only contributes to the improvement of construction claims negotiation, but also provides an effective approach for the development of multi-agent system negotiation mechanism.
ACKNOWLEDGEMENTS

I am most grateful to my supervisor, Professor C. J. Anumba, for his patient guidance, invaluable suggestions, and constant encouragement throughout this research project. Without his help in obtaining funding, this research and my Ph.D. would not be possible.

I wish to express my thanks to my colleagues, Dr. O. O. Ugwu, Dr. L. Newnham and Dr. L. Lee for their support in this research project.

I would also like to thank Professor A. Thorpe for his encouragement and friendship.

I am very much indebted to my parents, my wife and son, for their love, support and patience.

Many friends, either from China, Sri Lanka, Thailand or the UK who have helped me through all these years. I would also like to express my appreciation to them.
TABLE OF CONTENTS

TITLE PAGE .......................................................................................................................i
ABSTRACT ...................................................................................................................... ii
ACKNOWLEDGEMENTS .............................................................................................. iii
TABLE OF CONTENTS ................................................................................................. iv
LIST OF FIGURES .......................................................................................................... xiv
LIST OF TABLES ........................................................................................................... xvii

CHAPTER 1 - INTRODUCTION ................................................................................... 1

1.1 INTRODUCTION .................................................................................................. 1
1.2 BACKGROUND ..................................................................................................... 1
1.3 JUSTIFICATION AND HYPOTHESIS .................................................................. 4
1.4 AIM AND OBJECTIVES ..................................................................................... 5
1.5 METHODOLOGY ................................................................................................ 6
  1.5.1 Literature Review ........................................................................................ 6
  1.5.2 Interviews .................................................................................................... 8
  1.5.3 Conceptual Model Development ............................................................... 10
  1.5.4 Model Implementation and Evaluation ...................................................... 11
  1.5.5 System Operation ...................................................................................... 11
  1.5.5 System Evaluation ...................................................................................... 11
1.5 SCOPE OF RESEARCH ....................................................................................... 12
CHAPTER 2 - CONSTRUCTION CLAIMS MANAGEMENT AND NEGOTIATION}

2.1 INTRODUCTION

2.2 DEFINITION

2.2.1 Claims

2.2.2 Claims Management

2.3 HISTORICAL PERSPECTIVE

2.3.1 Basis of Claims Management: Starting Right

2.3.2 Claims Management Process: Staying Right

2.3.3 Computer Supported Claims Management

2.3.3.1 Management Principles

2.3.3.2 Engineering Software

2.4 PROBLEM

2.4.1 The Awareness and Interpretation of Contract Terms and Provisions

2.4.2 Inadequate Information and Documentation

2.4.3 Lack of Effective Claims Management Tool

2.4.4 Inefficient Claims Negotiation

2.5 CLAIMS NEGOTIATION

2.5.1 Generic Model of Claims Negotiation

2.5.1.1 Claims Negotiation Elements

2.5.1.2 Individual Roles in Negotiation
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5.1.3 Negotiation Process</td>
<td>48</td>
</tr>
<tr>
<td>2.5.2 Problems</td>
<td>54</td>
</tr>
<tr>
<td>2.5.3 Possible Solutions</td>
<td>57</td>
</tr>
<tr>
<td>2.6 SUMMARY</td>
<td>59</td>
</tr>
<tr>
<td><strong>CHAPTER 3 - MULTI-AGENT SYSTEMS</strong></td>
<td>61</td>
</tr>
<tr>
<td>3.1 INTRODUCTION</td>
<td>61</td>
</tr>
<tr>
<td>3.2 INTELLIGENT AGENTS</td>
<td>62</td>
</tr>
<tr>
<td>3.2.1 Definition</td>
<td>62</td>
</tr>
<tr>
<td>3.2.2 Agent Development</td>
<td>66</td>
</tr>
<tr>
<td>3.2.3 Agent Taxonomy</td>
<td>68</td>
</tr>
<tr>
<td>3.2.3.1 Activity</td>
<td>70</td>
</tr>
<tr>
<td>3.2.3.2 Motivation</td>
<td>72</td>
</tr>
<tr>
<td>3.2.4 Agent Co-ordination</td>
<td>74</td>
</tr>
<tr>
<td>3.3 MULTI-AGENT SYSTEMS</td>
<td>76</td>
</tr>
<tr>
<td>3.3.1 Distributed Problem Solving</td>
<td>77</td>
</tr>
<tr>
<td>3.3.2 Multi-Agent Systems (MAS)</td>
<td>78</td>
</tr>
<tr>
<td>3.4 NEGOTIATION IN MULTI-AGENT SYSTEMS</td>
<td>81</td>
</tr>
<tr>
<td>3.4.1 Overview</td>
<td>82</td>
</tr>
<tr>
<td>3.4.2 Agent Negotiation Mechanism</td>
<td>84</td>
</tr>
<tr>
<td>3.4.3 Negotiation Models</td>
<td>88</td>
</tr>
<tr>
<td>3.4.3.1 Contract-Based Negotiation</td>
<td>88</td>
</tr>
<tr>
<td>3.4.3.2 Plan-Based Negotiation</td>
<td>90</td>
</tr>
</tbody>
</table>
3.4.3.3 Market-Based Negotiation ........................................................... 81
3.4.3.4 Game Theory-Based Negotiation ................................................. 92
3.4.3.5 AI-Based Negotiation ................................................................. 95
3.4.3.6 Psychology-Based Negotiation .................................................... 96
3.4.3.7 Other Approaches ........................................................................ 97
3.4.4 Negotiations in DPS and MAS ........................................................ 97

3.5 LEARNING IN MULTI-AGENT SYSTEMS .................................................. 98
3.5.1 Overview ........................................................................................... 99
3.5.2 Issues of Learning ............................................................................. 100
3.5.2.1 Objective of Learning ................................................................ 100
3.5.2.2 Key Elements in Learning ......................................................... 103
3.5.3 Methods of Learning ........................................................................ 105
3.5.4 Challenging Research Issues ........................................................... 109

3.6 MAS IN THE CONSTRUCTION INDUSTRY ............................................. 110
3.6.1 Why MAS? ....................................................................................... 110
3.6.2 Some Applications ........................................................................... 112
3.6.2.1 Agent-Based Design .................................................................. 112
3.6.2.2 Negotiation ................................................................................ 114
3.6.3 Problems and Challenges ............................................................... 116

3.7 SUMMARY ............................................................................................. 118

CHAPTER 4 - NEGOTIATION THEORIES ....................................................... 120
4.1 INTRODUCTION ..................................................................................... 120
4.2 MECHANICAL THEORY ................................................................. 123

4.2.1 The Classical Game Theoretical Approach ......................... 124
4.2.2 The Economic Theoretical Approach .................................. 128

4.3 Behaviour Theory ................................................................. 131

4.3.1 Psychological Model .......................................................... 131
4.3.2 The Learning Model ........................................................... 132
4.3.3 The Dual Responsiveness Model ....................................... 134
4.3.4 The Joint Decision Making Model ..................................... 134

4.4 Summary .................................................................................. 135

CHAPTER 5 - DEVELOPMENT OF THE MASCOT CONCEPTUAL MODEL . 137

5.1 INTRODUCTION ........................................................................... 137

5.2 GENERAL CHARACTERISTICS OF CLAIMS NEGOTIATION ............ 138

5.2.1 Contractually Obliged Self-Interested Relationship ................ 138
5.2.2 Participant-Dependent Information ..................................... 139
5.2.3 Strategy-Influenced Process ................................................. 140
5.2.4 Time - An Important Factor ............................................... 141
5.2.5 Tradeoffs between Negotiation Items ................................. 142

5.3 NEGOTIATION MODEL IN MASCOT .............................................. 142

5.3.1 Game and Economic Theory Models ................................. 142
5.3.2 Behaviour Theory Model .................................................... 145

5.4 MASCOT MODEL ...................................................................... 146

5.4.1 Negotiation Domain ............................................................ 146
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4.2</td>
<td>Assumptions and Requirements</td>
<td>147</td>
</tr>
<tr>
<td>5.4.3</td>
<td>Roles of Players in MASCOT</td>
<td>150</td>
</tr>
<tr>
<td>5.4.3.1</td>
<td>Task Agents</td>
<td>150</td>
</tr>
<tr>
<td>5.4.3.2</td>
<td>Utility Agents</td>
<td>153</td>
</tr>
<tr>
<td>5.4.4</td>
<td>Negotiation Protocol and Strategies</td>
<td>154</td>
</tr>
<tr>
<td>5.4.4.1</td>
<td>Negotiation Protocol</td>
<td>155</td>
</tr>
<tr>
<td>5.4.4.2</td>
<td>Negotiation Strategies</td>
<td>160</td>
</tr>
<tr>
<td>5.4.5</td>
<td>Process MODEL</td>
<td>174</td>
</tr>
<tr>
<td>5.4.5.1</td>
<td>Making an Initial Offer</td>
<td>175</td>
</tr>
<tr>
<td>5.4.5.2</td>
<td>Evaluating the Offer and Making a Counteroffer</td>
<td>175</td>
</tr>
<tr>
<td>5.4.5.3</td>
<td>Conducting Expanded Solution Searching</td>
<td>176</td>
</tr>
<tr>
<td>5.5</td>
<td>SUMMARY</td>
<td>184</td>
</tr>
</tbody>
</table>

CHAPTER 6 - IMPLEMENTATION OF THE MASCOT MODEL

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>INTRODUCTION</td>
<td>185</td>
</tr>
<tr>
<td>6.2</td>
<td>IMPLEMENTATION ENVIRONMENT</td>
<td>186</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Hardware</td>
<td>186</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Software</td>
<td>186</td>
</tr>
<tr>
<td>6.3</td>
<td>THE ZEUS AGENT BUILDING TOOLKIT</td>
<td>187</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Issues for Building of Collaborating Agent Systems</td>
<td>188</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Essential Functions of ZEUS</td>
<td>189</td>
</tr>
<tr>
<td>6.3.3</td>
<td>The ZEUS Toolkit Architecture</td>
<td>191</td>
</tr>
<tr>
<td>6.3.3.1</td>
<td>The Agent Component Library</td>
<td>191</td>
</tr>
</tbody>
</table>
6.3.3.2 The Generic ZEUS Agent .......................................................... 192
6.3.3.3 The ZEUS Agent Building Software ......................................... 194
6.3.3.4 The ZEUS Utility Agents ........................................................... 195
6.3.3.5 Integrating ZEUS Agents with External Programs ................. 195

6.4 MASCOT MODEL IMPLEMENTATION ............................................. 196

6.4.1 Role Modelling ............................................................................. 198
   6.4.1.1 The Components of a Role Model ............................................. 198
   6.4.1.2 MASCOT Role Modelling ....................................................... 200

6.4.2 Application Design ...................................................................... 205
   6.4.2.1 Negotiation Protocol and Strategies ........................................... 206
   6.4.2.2 Input Information ................................................................. 207
   6.4.2.3 Determination of the Probability Distribution ....................... 213

6.4.3 Model Realisation ........................................................................ 218
   6.4.3.1 Ontology Creation ................................................................. 218
   6.4.3.2 Agent Creation ...................................................................... 223
   6.4.3.3 Agent Organisation ............................................................... 226
   6.4.3.4 Agent Co-ordination ............................................................. 226
   6.4.3.5 Rulebase Definition ............................................................... 227
   6.4.3.6 Primitive Task Definition ...................................................... 231
   6.4.3.7 The Utility Agent Configuration .......................................... 236
   6.4.3.8 The Task Agent Configuration Stage .................................... 238
   6.4.3.9 Primitive Task Implementation and External Classes ........... 240
   6.4.3.10 Display Windows ................................................................. 245
8.3.2 Prototype Evaluation ..................................................................................293

  8.3.2.1 Evaluation Techniques .........................................................................293

  8.3.2.2 Choice of Evaluators ..........................................................................294

  8.3.2.3 Evaluation Questionnaire ....................................................................295

  8.3.2.4 Discussions .......................................................................................296

8.4 EVALUATION OF THE MASCOT NEGOTIATION MECHANISM .............296

  8.4.1 Criteria for Effective Assessment System ............................................296

  8.4.2 Analysis ..................................................................................................298

8.5 EVALUATION OF THE MASCOT PROTOTYPE ...........................................300

  8.5.1 Evaluation Process ..................................................................................300

  8.5.2 Analysis of Results ................................................................................301

     8.5.2.1 Evaluators’ Background ..................................................................301

     8.5.2.2 Responses to Questions ...................................................................302

  8.5.3 Discussion ................................................................................................315

     8.5.3.1 Results ..........................................................................................315

     8.5.3.2 Effectiveness of the Evaluation Approach .........................................316

8.6 SUMMARY .....................................................................................................317

CHAPTER 9 - CONCLUSIONS AND FURTHER DEVELOPMENTS ..........319

  9.1 INTRODUCTION ...........................................................................................319

  9.2 SUMMARY ..................................................................................................319

  9.3 CONCLUSIONS ............................................................................................325

  9.4 RECOMMENDATIONS FOR FURTHER DEVELOPMENT .......................329
9.4.1 Further Development of the MASCOT Prototype ................................... 329
9.4.2 Development of MAS for Construction Claims Management .................. 331
9.4.3 Application and Development of MAS in Construction .......................... 331
9.5 CLOSING REMARKS ....................................................................................... 333

REFERENCES .............................................................................................................. 335

APPENDICES

A: Publications arising from This Research ......................................................... 359
B: Questionnaire for Prototype Evaluation ........................................................... 361
C: Presentation for Prototype Evaluation .............................................................. 367
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1.1</td>
<td>Thesis structure</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2.1</td>
<td>Claims management procedure</td>
<td>29</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Generic claims negotiation model</td>
<td>43</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Negotiation planning process</td>
<td>51</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Nwana's requirements for agenthood</td>
<td>65</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>An agent architecture</td>
<td>65</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>The classification of learning methods for MAS</td>
<td>105</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>A general learning mechanism</td>
<td>133</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>The dual responsive model</td>
<td>134</td>
</tr>
<tr>
<td>Figure 5.1a</td>
<td>The contractual relationships between project participants</td>
<td>151</td>
</tr>
<tr>
<td>Figure 5.1b</td>
<td>The common claims negotiation relationships between project participants</td>
<td>151</td>
</tr>
<tr>
<td>Figure 5.2</td>
<td>The negotiation relationships in the MASCOT model</td>
<td>152</td>
</tr>
<tr>
<td>Figure 5.3</td>
<td>Tripartite negotiation</td>
<td>160</td>
</tr>
<tr>
<td>Figure 5.4</td>
<td>The general utility consideration of agents</td>
<td>162</td>
</tr>
<tr>
<td>Figure 5.5a</td>
<td>An example of the contractor's negotiation zone</td>
<td>166</td>
</tr>
<tr>
<td>Figure 5.5b</td>
<td>An example of the agreement zone between agents</td>
<td>166</td>
</tr>
<tr>
<td>Figure 5.6</td>
<td>Bayesian updating</td>
<td>169</td>
</tr>
<tr>
<td>Figure 5.7</td>
<td>The involvement of the client agent</td>
<td>173</td>
</tr>
</tbody>
</table>
Figure 6.15 Ontology editor window ................................................................. 223
Figure 6.16 Agent definition window in ZEUS .................................................. 224
Figure 6.17 Rulebase editor window ................................................................. 228
Figure 6.18 Primitive task editor window .......................................................... 233
Figure 6.19 The co-operation between rulebases and primitive tasks ............... 235
Figure 6.20 The utility agents configuration panel ............................................. 236
Figure 6.21 The task agent configuration panel .................................................. 239
Figure 6.22 External programs and their related external classes ....................... 240
Figure 6.23 Flow of information between agent, task and external class .......... 241
Figure 6.24 Display window of a party's estimate of the opponent's information ... 246
Figure 6.25 Display window for the contractor or the engineer in negotiation .... 247
Figure 6.26 Display window for the client during negotiation ......................... 247
Figure 7.1a The Contractor agent's utility function ........................................... 254
Figure 7.1b The Engineer agent's utility function ............................................. 254
Figure 7.1c The correlation between the two agents' utility functions ................. 254
Figure 7.2 The negotiation process of Example 1 .............................................. 278
Figure 7.3 The negotiation with time penalty (Example 2) ................................ 279
Figure 7.4 The negotiation reaches a conflict deal (Example 3) ......................... 280
Figure 7.5 The negotiation with random conditional probability (Example 4) .... 283
Figure 7.6 The negotiation with random conditional probability (Example 5) .... 284
LIST OF TABLES

Table 2.1  Likely use of documents in claim presentation ........................................... 32
Table 2.2  Number of claims for direct and indirect costs ........................................... 33
Table 2.3  Factors influencing negotiations between project participants ................. 48
Table 3.1  Agents in historical perspective ................................................................. 66
Table 3.2  Agent qualities ......................................................................................... 69
Table 3.3  Categories for agent negotiation ............................................................... 83
Table 5.1  Some of the important protocols .............................................................. 156
Table 6.1  The components of generic ZEUS agent and their functions .................... 193
Table 6.2  Interaction summary .............................................................................. 201
Table 6.3  Contractor role description ..................................................................... 202
Table 6.4  Engineer role description ........................................................................ 202
Table 6.5  Negotiator role description ..................................................................... 203
Table 6.6  Client role description ............................................................................ 203
Table 6.7  Mfacilitator role description ................................................................... 203
Table 6.8  Agents created in MASCOT model ......................................................... 205
Table 6.9  Attribute assumptions of normal distribution .......................................... 210
Table 6.10  The attribute assumptions of lognormal distribution ............................... 213
Table 6.11  Reservation distribution ........................................................................ 214
| Table 6.12 | Lognormal distribution ................................................. 214 |
| Table 6.13 | Conditional probability of the engineer's offer given the reservation value ................................................. 216 |
| Table 6.14 | Modified conditional probability of the Engineer's offer given the reservation value ................................................. 217 |
| Table 6.15 | MASCOT ontology ........................................................................ 220 |
| Table 6.16 | Tasks, rulebases and initial resources related to agents in MASCOT ................................................................. 225 |
| Table 6.17 | Rulebases of MASCOT model ......................................................... 231 |
| Table 6.18 | Modifiers in primitive task preconditions ......................................................... 232 |
| Table 6.19 | Primitive tasks in MASCOT model ......................................................... 234 |
| Table 6.20 | Primitive tasks and external classes ......................................................... 242 |
| Table 7.1 | The contractor's major negotiation figures ......................................................... 255 |
| Table 7.2 | Contractor's prior knowledge about the probability distribution of the engineer's reservation value ................................................. 255 |
| Table 7.3 | The contractor's prior knowledge about the conditional probabilities of the engineer's offer ......................................................... 256 |
| Table 7.4 | The engineer's major negotiation figures ......................................................... 256 |
| Table 7.5 | The engineer's prior knowledge about the probability distribution of the contractor's reservation value ................................................. 256 |
| Table 7.6 | The Engineer's prior knowledge about the conditional probabilities of the contractor's offer ......................................................... 257 |
| Table 7.7 | The contractor agent's estimate about the probability of the engineer agent's reservation value at the beginning of 3rd iteration ................................................. 263 |
Table 7.8  The engineer agent's estimate about the probability of the contractor agent's reservation value at the beginning of 3rd iteration......266

Table 7.9  The contractor agent's estimate about the probability of the engineer agent's reservation value at the beginning of 4th iteration ......268

Table 7.10 The engineer agent's estimate about the probability of the contractor agent's reservation value at the beginning of 4th iteration......270

Table 7.11 The contractor agent's estimate about the probability of the engineer agent's reservation value at the beginning of 5th iteration ......273

Table 7.12 The engineer agent's estimate about the probability of the contractor agent's reservation value at the beginning of 5th iteration......275

Table 7.13  The negotiation process (Example 1)...............................277

Table 7.14 The negotiation process with time penalty (Example 2)........279

Table 7.15 The negotiation process with a conflict outcome (Example 3).....280

Table 7.16 The contractor's prior knowledge about the conditional probabilities of the engineer's offer (Example 4).................................................................281

Table 7.17 The contractor's prior knowledge about the conditional probabilities of the engineer's offer (Example 5).................................................................281

Table 7.18 The engineer's prior knowledge about the conditional probabilities of the contractor's offer (Example 4, 5).................................................................282

Table 7.19 The negotiation process with random conditional probability (Example 4).................................................................282

Table 7.20 The negotiation process with random conditional probability (Example 5).................................................................284

Table 7.21 The contractor agent's major negotiation figures (Example 6) ....285

Table 7.22 The engineer agent's major negotiation figures (Example 6)........285

Table 7.23 The negotiation process with same concession rate (Example 6)....286
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.24</td>
<td>The negotiation process with same concession amount (Example 6)</td>
<td>287</td>
</tr>
<tr>
<td>8.1</td>
<td>The evaluators’ background</td>
<td>300</td>
</tr>
<tr>
<td>8.2</td>
<td>Responses to construction claims negotiation</td>
<td>303</td>
</tr>
<tr>
<td>8.3</td>
<td>Responses to Multi-agent systems</td>
<td>305</td>
</tr>
<tr>
<td>8.4</td>
<td>Responses to the MASCOT model</td>
<td>306</td>
</tr>
<tr>
<td>8.5</td>
<td>Responses to general questions (1)</td>
<td>309</td>
</tr>
<tr>
<td>8.6</td>
<td>Responses to general questions (2)</td>
<td>313</td>
</tr>
<tr>
<td>8.7</td>
<td>Responses to general questions (3)</td>
<td>314</td>
</tr>
<tr>
<td>6.1</td>
<td>Integration of primitive tasks and external Java codes</td>
<td>245</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

This chapter presents a general overview of the thesis consisting of a brief introduction and description of the subject matter of the research as well as the specific problems being studied. It also sets out the hypothesis, aims and objectives, the manner in which the research was carried out, a summary of its achievements and the structure of the thesis.

1.2 BACKGROUND

Claims appear, indeed, to be an integral part of almost all civil engineering and building contracts, and arise from the many uncertainties which are an inherent part of the industry (Bramble and Callahan, 1992; Sykes, 1999). Very few, if any, construction contracts can ever be completed without claims being made for additional payment, or for extra time in which to complete the work. The reason for this is the extremely complex, dynamic and fragmented nature of the construction industry. It is genuinely difficult to recognise fully, at the commencement of a project, all the uncertainties of one kind or another in spite of all efforts to the contrary or project management. This leads to an increasing number and amount of claims in the industry. For example, Keane (1994) reported that £1.2 billion could be the subject of construction claims or disputes at any one time, and that more than 83% of contractors in the UK claimed for one or more time extensions during 1992-1994, whilst 84% of them were dissatisfied with the result.

Claims management therefore has gained more attention in construction project management. Various efforts have been geared towards reducing claims, facilitating
claims settlements, and controlling the incidence of claims, which includes the
development of contractual/legal issues, the introduction of risk management in contract
analysis, the adoption of new procurement systems, the improvement of the claims
management process (particularly project documentation systems), the application of
information technologies, and the adoption of alternative dispute resolution (ADR)
mechanisms. On the other hand, industrial evidence shows that there is a tendency for
construction claims and disputes to increase at an alarming rate (Keane, 1994). Therefore,
more studies on various aspects of claims need to be explored further.

Negotiation is an important stage in construction claims management process. It plays a
vital role in claims settlement, disputes prevention and resolution. However, claims
negotiations are normally conducted inefficiently (Hu, 1997). Although this problem has
been widely recognised, claims negotiation has seldom been thoroughly analysed. Proper
solutions for the problem are rare. This is because claims negotiation often involves
problems like diversity of intellectual background, many variables, complex interactions,
and inadequate negotiation expertise of construction participants. The improvements for
claims negotiation, thus far, normally focus on the human aspects such as: planning
negotiation, preparing adequate negotiation documents, or adopting proper negotiation
strategies (Smith, 1992; Zack, 1994).

The revolutionary development of computer science, in the form of information
technology (IT), in the last two decades has not only changed the concept of engineering
design, but has also made vital improvements in project construction and management
with the developments of new communication tools, planning and scheduling methods,
documentation systems, expert and decision systems, and knowledge management. These
systems also play an important role in facilitating claims management. However, none of
the existing tools have essentially improved the claims negotiation process, although
expert systems and decision support systems have been adopted. The reason is that these
systems can only provide suggestions to the human negotiators.

Multi-agent systems (MAS) is a fast developing information technology, where a number
of intelligent agents, representing the real world parties, co-operate or compete to reach
their desired objectives. There is no central controlling agent within the group. Each agent is characterised as having the following attributes (Nwana, 1996):

- **Autonomy**: Agents can operate on their own without the need for human guidance. Agents have individual internal states and goals, and act in such a manner as to meet their goals.

- **Co-operation**: Co-operation with other agents is paramount and is the reason for having multiple agents in the first place. In order to co-operate, agents need to possess the ability to interact with other agents and possibly humans.

- **Learning**: For agent systems to be truly 'smart', they would have to learn as they react and/or interact with their external environment. A key attribute of any intelligent being is its ability to learn.

The distributed-collaborative characteristics of multi-agent systems make it an ideal tool to simulate and resolve many industrial problems such as: traffic control, collaborative design, business negotiation, e-commerce, and supply chain management. Meanwhile, the significance of multi-agent systems is likely to increase with the developing communication language, and the successful applications in different industries. Since negotiation is one of the major interests of multi-agent systems, this research aims to adopt multi-agent systems to resolve the existing problems in construction claims negotiation.

Despite its advantages, the introduction of information technology into the industry does not naturally lead to an improvement to the industrial problems, especially for those involving complex human and technical issues. For example, with many powerful documentation tools available, documentation is still a major problem in claims management. On the other hand, with the application of the Critical Path Method (CPM) and related software, construction planning and scheduling has improved greatly. The key point is that the application of information technology should be based on a thorough analysis of the application scenarios, a full understanding of the IT system, and a properly
developed system. This research will follow such an approach in developing a multi-agent system for construction claims negotiation (MASCOT).

1.3 JUSTIFICATION

Compared with the studies on other aspects of claims management such as: contractual/legal issues, procurement systems and contract administration, the studies on claims negotiation are much fewer. Nevertheless, the importance of claims negotiation in claims settlement, project relationship facilitation and disputes resolution has been recognised. Smith (1992) points out that negotiation has been labelled as a basic survival skill for today's project manager.

On the other hand, many key technical, professional and academic literature on claims negotiation reveal that current claims negotiations are conducted in a heuristic and inefficient way (Fenn et al., 1998). According to Hu's survey (1997), most project managers respond that claims negotiation is the most time and energy-consuming activity which they are involved in during the claims management process. The reasons are as follows (Ren et al., 2001a):

- Claims negotiation items are often arguable. Some items such as loss of productivity and overheads, by their nature, are vague and difficult to decide even with much evidence;
- The claims negotiation preparation and negotiation process is time-consuming;
- Certain problems, such as the engineer's conflicting roles in claims management caused by the construction project organisation structure sometimes become major obstacles for claims negotiation; and
- Most construction participants lack adequate negotiation expertise. Since claims negotiation involves complex human factors in addition to pure technical issues, a negotiator's personal abilities and attitudes often determine the result of a negotiation. Lack of the necessary negotiation expertise, unnecessary concession, and stubbornness are common mistakes in claims negotiation.
These problems may cause claims, which could be settled otherwise, to develop into disputes, or in the worst case, into arbitration or litigation. Moreover, unsuccessful claims negotiation often causes an adversarial project environment, which may lead to loss of productivity, time extension, and increase in the client’s costs (Filef et al., 1993).

The motivation for this research is twofold:

- Essentially, the research is motivated from the need to improve construction claims negotiation;

- It also derives from the development of multi-agent systems as a flexible and powerful tool to simulate and resolve real world co-operation problems. The distributed and collaborative characteristics of multi-agent systems constitute a promising approach to solve complicated negotiation problems in a natural way (Ferber, 1998). The successful applications of multi-agent systems in other disciplines, especially in negotiation fields, provide valuable experience for this application.

Meanwhile, the research also plans to draw some experiences for the application of multi-agent systems in other fields in the construction industry.

1.4 AIM AND OBJECTIVES

The aim of this research is to investigate the applicability of a multi-agent system to the construction claims negotiation process, such that the specific problems in current claims negotiation can be minimised. To achieve this aim, the following specific objectives were set:

- Explore the role, process, and influence factors of claims negotiation to build a conceptual model of claims negotiation; and further identify the nature and characteristics of claims negotiation to build the reasoning model on which the multi-agent system should be based;
• Examine the major principles of multi-agent systems, explore its applications in other related disciplines, and identify the key issues to be addressed for application. Particular attention is given to agent architectures and negotiation mechanisms;

• Examine the key negotiation theories and the major theoretic models, and identify their possible applications in multi-agent systems;

• Develop a multi-agent system for the claims negotiation based on the identified reasoning model and characteristics of claims negotiation, and the selected negotiation theoretic models; and

• Implement the developed MASCOT model in an agent development environment; execute the system and the prototype; evaluate the system and prototype; and identify problems and possible solutions.

### 1.5 METHODOLOGY

To achieve the above objectives, various research methods were adopted. These include: literature review, interviews, conceptual modelling, rapid prototyping, and system evaluation. Details are discussed below.

#### 1.5.1 Literature Review

An extensive range of literature (books, journal papers, industrial and academic reports, and Internet information and discussion) was reviewed in this research. The review initiated the research; and contributed to three major aspects of the research:

• It established the general background to the research;

• It allowed the research issues to be identified; and

• It not only provided the historical knowledge of the research topic, but also ensured that the most current developments and thinking were considered and included.
The critical review of literature mainly focused on three main areas:

- Various aspects of construction claims management including: current industry practice, legal and management principles, risk management, contract administration, and IT applications in claims management and dispute resolution. The main focus was on claims negotiation including elements involved in negotiation, negotiation process, problems, and possible solutions.

- Key aspects of multi-agent systems including intelligent agent definition, classification, development, agent architecture and co-ordination mechanism, the essential characteristics of multi-agent systems, working domains, negotiation mechanisms, learning approaches, and applications in the construction industry. Particular attention was paid to the analysis of the various aspects of negotiation mechanisms and learning approaches of multi-agent systems.

- The major negotiation theories covering game theory, economic theory, and behaviour theory. The application scope, assumptions, and major models were discussed. Particular attention was paid to the application of these models to the development of multi-agent system negotiation mechanisms.

The review addressed the first three research objectives. It was an ongoing process by adding and removing items as new literature highlighted the studies that might be applicable. This was particularly important for the literature regarding multi-agent systems. There were more than one hundred new papers published on the Internet from 1999 to 2001. The list of literature was constantly updated and reviewed to ensure that the most current, peer reviewed publications were traced and considered in this research.

1.5.2 Semi-Structured Interviews

Due to the very limited number of publications on claims negotiation, a qualitative study by means of semi-structured interviews was conducted to gain better understanding on construction claims negotiation. A formal interview may be conducted as opportunity
permits to the formal, pre-arranged interview; whilst an informal interview demands a more probing, spontaneous and unstructured technique governed by the situation rather than by the topic of interest. The semi-structured interviews fall between these two categories (Gummesson, 2000).

This method is often used for gathering opinions about a subject. However, the data obtained through this research method is not suitable for generalisation to any population. Data is obtained from too small a sample and, because of certain attributes, a sample is drawn to include specially selected research participants (Babbie, 1986). The aim with this kind of research is to gather insight into a problem from an informed group of people rather than quantitative data from a sample of research participants. The results of a qualitative study can indicate the scope of possible matters on a subject, but does not indicate how many people in a population will agree with a certain statement. As the objective is to obtain information on the nature of claims negotiation, this qualitative method was considered suitable.

Interviews, based on a scheme of questions, were conducted to obtain information from industry participants. Four experienced international project managers (representing the contractor) and two quantity surveyors (working for the client) took part in the interview. The interviews with the international project managers were undertaken through electronic means while the two quantity surveyors were interviewed face to face. The questions were based on knowledge obtained through the literature review and the author's industrial experience, which indicated the areas where problems could be expected in claims negotiation and the application of IT to facilitate claims negotiation. The topics covered were:

- the current status and major characteristics of claims negotiation including: application situations, negotiation procedures, documentation required, considering factors, and decision criteria (e.g. risk attitude);
- major problems and the causes of the problems;
- IT tools currently used in claims management and the perspectives of using IT technologies to resolve the problems identified; and
factors needed to be considered in this research and the potential barriers for the application of the proposed system.

The interviews were intended to obtain the participants' opinions regarding the various aspects of claims negotiation. Although some participants did not answer all the planned questions, they gave more detailed opinions on other aspects that also benefited this research. All the responses from the interviews were categorised to obtain a body of opinions and remarks. The responses were also compared with the literature review and the author's expected results. Finally, the responses were further analysed to make meaningful deductions from the data. No statistical analysis was made to avoid the temptation to generalise the information to a broader population.

Also, since there were limited formal publications about multi-agent systems (particularly at the beginning of this project), interviews, discussions and correspondence with domain experts were also a main source to get a deep insight of multi-agent systems.

Through these exercises and the literature reviews, it was possible to get a very good picture of claims negotiation, build a conceptual model of claims negotiation, address the problems, and figure out possible methods for the improvement of claims negotiation. Also, a deeper understanding of the essential characteristics of multi-agent systems, and how multi-agent systems can be applied in resolving industrial problems was also gained. More importantly, the key concepts of negotiation theories were established. Theoretical background established at this stage allowed the remaining objectives to be fulfilled.

1.5.3 Conceptual Modelling

Based on the knowledge gained at the previous stages, the MASCOT conceptual model was developed. The development of the conceptual model followed an interactive process, which involved the following:
Thorough analyses of the nature and characteristics of construction claims negotiation in terms of the application of multi-agent systems to build the reasoning model and identify the problems that need to be resolved by multi-agent systems;

A comparative analysis of the principles, application domains, conditions, benefits and limitations of the negotiation theories with respect to the specific research problem. This study was the breakthrough point of the model development process;

Analysis and development of the negotiation mechanism and learning approach; and

Development of a process model for the agent-based claims negotiation system using the IDEF0 methodology;

The developed conceptual model was further improved by reviewing the following sources: discussions with academic staffs involved in the research, suggestions and comments from the industry participants, peer reviews from researchers at conferences and workshops, and anonymous reviews by referees from academic journals. The iterative development of the model resulted in a series of papers at different development stages of the project, which are listed in Appendix A.

1.5.4 Rapid Prototyping

As part of the research objectives, the development of a prototype for the MASCOT model was required to demonstrate the implementation of the model in a multi-agent system development environment. Since the development of multi-agent systems is relatively new, there were very few mature software development environments to be adopted for the implementation of the system. This research adopted the ZEUS agent building toolkit developed by BT Plc. Based on the agent generating framework provided by the ZEUS toolkit, the negotiation protocol and strategies were fulfilled through the external JAVA program. The system input information and its transformation into the computing data were particularly addressed.
The overall approach to system development involved rapid prototyping, which entails rapid development of subsystems, obtaining feedback from academics and practitioners, and refining the system until it works well.

1.5.5 System Evaluations

Two different approaches were adopted to evaluate the MASCOT system in terms of its negotiation mechanism and prototype.

- Firstly, a theoretical analysis was conducted to evaluate the major aspects of the MASCOT negotiation mechanism, where a number of key criteria were adopted which are widely accepted in the multi-agent system community. Three major aspects of the system (i.e. efficiency, stability and simplicity) were analysed to ensure that the negotiation mechanism is theoretically sound. The suitability of the system was also highlighted.

- Secondly, a prototype evaluation was undertaken to allow the potential end-users to assess the system. A group of construction professionals were invited to take part in a demonstration of the prototype, where the MASCOT system was first presented to the evaluators, then the prototype was executed with an example. Following the demonstration, evaluators were requested to assess the effectiveness of the prototype by completing a questionnaire. General discussions were also undertaken during the demonstration to allow the evaluators to raise any questions which were not included in the questionnaire.

A detailed discussion about the system evaluation methodology is reported in Chapter 8.

1.6 SCOPE OF RESEARCH

Since this research mainly focused on the development of information technology to facilitate the claims negotiation, many complex human factors and technical issues were not considered. There are two closely linked negotiation stages in construction claims
management: negotiations about the entitlement of claims, and negotiations about the compensation amount of claims. At the first stage, negotiations are mainly concerned with the causes, evidence, responsibilities and the related clauses of the Conditions of Contract. Negotiations are essentially qualitative. Once a claim is entitled, negotiations are about the amount of the claim (i.e. the focus is on quantitative issues). This research is mainly concerned with the negotiations at the second stage, i.e., the quantitative negotiations. Those strategies involving strong human factors in quantitative negotiations, such as: use of negotiation power, threatening, and relationship considerations were not considered. Hence, the MASCOT system provides a relatively neutral negotiation environment.

1.7 CONTRIBUTIONS

Many construction problems are associated with the fragmented nature of the industry. Although different information technologies have been applied, none of them, thus far, fully addresses this problem. The development of multi-agent systems provides an innovative approach to addressing the fragmented problem. By adopting this approach, problems such as the inefficiency of claims negotiation, which is difficult to improve through other approaches, is possibly resolved or relieved. However, the performance of a multi-agent system is highly dependent on the negotiation protocol and strategies adopted, which, in turn, should be developed based on the individual application scenario. The poor performance of multi-agent systems has been a major obstacle for widespread acceptance of the systems (Krothapalli and Deshmukh, 1999). Thus, application of a multi-agent system for claims negotiation needs a thorough understanding in both areas. Much creative work has been done regarding these aspects within this thesis. The specific contributions of this research can be concluded as:

- First, this research made a thorough analysis of the nature, process and characteristics of construction claims negotiation. Problems underlying the claims negotiation were pointed out. Possible solutions for those problems were suggested.
Second, this research examined the major negotiation theories and their possible applications in multi-agent systems. This approach goes far beyond any former analyses of the theoretic basis of multi-agent systems, where AI researchers focus more on the game theory. By following the theoretical analysis in this research, more flexible and general theoretic models can be easily identified and developed for the further application of multi-agent systems in different application domains.

Third, a specific negotiation protocol has been developed which particularly addresses the problems of the engineer's conflicting roles and the low involvement of the client in claims negotiation. Also, the time issue and the extensive resolution-searching process in claims negotiation are also highlighted.

Fourth, a new negotiation mechanism has been developed which integrates an economic theoretical model: Zeuthen's bargaining model, and a learning approach, Bayesian inference approach. The negotiation mechanism particularly addresses the contractually obliged self-interested nature of claims negotiation. Meanwhile, it ensures the developed multi-agent system is stable and efficient.

Fifth, this research contributes a fully functioning multi-agent system implemented by using the ZEUS agent developing toolkit. Detailed algorithmic descriptions of the agents' behaviour and some of source code are included in the thesis.

1.8 THESIS LAYOUT AND CONTENTS

The thesis contains eight chapters illustrated in Figure 1.1.

Chapter 1 provides an overview to the research undertaken. It first introduces the background and incentives of this thesis; then discusses the objectives, methodology, scope and contributions of this research; finally it provides a guide to the structure of the thesis.
Chapter 2 explores the development of claims management and identifies the procedure, influential factors, conceptual models, problems and suggestions for improvement of claims negotiation.

Chapter 3 studies the fundamental principles of multi-agent systems and their application in other related disciplines. Particular focus is on individual negotiation mechanisms in different application areas, and various learning approaches adopted in multi-agent systems.

Chapter 4 investigates the fundamental principles of negotiation theories and various theoretical models to establish the foundations for the selection of an appropriate theoretical model for the MASCOT model.

Chapter 5 describes the MASCOT model development process. It first identifies the key characteristics of the claims negotiation (especially the reasoning model) on which the MASCOT model should be built. It then selects the negotiation theoretical models for the MASCOT model based on the identified reasoning model. Finally, the negotiation protocol and strategies are developed.

Chapter 6 presents the implementation of the developed MASCOT model in the ZEUS agent developing toolkit. The implementation includes two major stages: the implementation design stage and implementation stage. At the first stage, role modelling is analysed; roles for each agent are defined; and ontology is defined. At the second stage, the MASCOT negotiation protocol and strategies are developed through rulebase, primitive task and external Java program.

Chapter 7 illustrates the execution process of the MASCOT model in a real application scenario involving a claim for loss of productivity. A number of the key points of system operation are described including: the utility functions, the negotiation preparation, the negotiation process, and the comparisons between the MASCOT negotiation mechanism with other negotiation mechanisms.

Chapter 8 describes the system evaluation, and reports on the prototype assessment process and results. Firstly, the MASCOT system is evaluated according to several
important criteria, which are specifically defined for evaluating multi-agent system negotiation mechanism. Secondly, the developed prototype is presented, and evaluated by a group of industry experts through the use of a questionnaire.

Chapter 9 provides a summary of the thesis, draws conclusions from the research, and makes recommendations for future research.
Figure 1.1 Thesis structure
CHAPTER 2
CONSTRUCTION CLAIMS MANAGEMENT AND NEGOTIATION

2.1 INTRODUCTION

Over the past three decades, the construction industry has experienced an increase in claims, liability exposures and disputes, along with an increasing difficulty in reaching reasonable settlements in an effective, economical and timely manner (Barrie and Paulson, 1992). The dynamic, complex and fragmented nature of the industry inevitably leads to a situation where conflicts are bound to arise, and claims are inevitable. As a result, claims are considered as a way of life for the construction industry (Bradley and Langford, 1987; Sykes, 1999), as shown by the following:

- World Bank (1990) figures show that for 1627 projects completed between 1974 and 1988 the cost overrun varied between 50% and 80%;
- Onyango (1993) found that 52% of all UK construction projects ended up with a claim of some type;
- Semple et al. (1994) identified that more than half of claims constituted an additional cost of at least 30% of the original contract value based on their survey of construction projects in Canada. In addition, about a third of claims amounted to at least 60% of the original contract value. In a few cases, the claim values were almost as high as the original contract value;
- HMSO (1995) reports that the average time overrun for UK government construction projects for the period 1993-1994 was 23.2%; and
- Chung (1998) reports that there are 20,923 claims being received for the 152 Hong Kong airport contracts. Of these, 6,047 claims have been resolved at a cost of HK $2.87 billion, compared with the claimed value HK $10.8 billion up to the end of December, 1997.
Although detailed statistical data is difficult to find in the area due to the obvious reluctance of claimants and respondents to publicise such, many researchers (Adrian, 1988; Bramble and Callahan, 1992; Powell-Smith and Stephenson, 1993; Jergeas and Hartman, 1994; Hu, 1997; Vidogah and Ndekugri, 1998) have identified the fact that both the number and amount of claims in the industry have increased steadily over the past decade. Meanwhile, disputes arise when a claim or assertion made by one party is rejected by the other party, and that rejection is not accepted.

- Bramble and Callahan (1988) reported that one third of the clients of major new construction projects are involved in arbitration or litigation of construction claims;

- The construction industry dispute avoidance and resolution task force (DART, 1995) estimates that more than $60 billion dollars is spent annually on change orders in the United States, which is a major cause of construction claims and disputes. More indirect costs arise from this epidemic (Ibbs, 1997); and

- Hartman (1995) also reported the high costs of litigation caused by construction claims in the United States, which had reportedly reached as high as 20% of the cost of building in the 1980s.

The reasons for the increasing construction claims and disputes are very complex; they can be analysed from social, industrial and project aspects (Ren et al., 2001a).

- **Social factors**: the construction industry, as a whole, is under increasing pressure from society, which requires the industry to be more competitive in terms of cost, time, quality and environmental issues. For example, the pre-tax profit for the constructors of heavy utility has dropped from 7-10% to 2-4% in the last two decades since the early 1970s (Zack, 1993). The industry is becoming more risky than ever.

- **Industrial factors**: the wide range of participants often with competitive and adversarial attitudes, separation of design from construction, lack of integration, the increasing size of projects, enhanced competitive tendering, lack of effective communication, increasing technological complexity, uncertainty in construction environments, unbalanced risk allocation, and complex and confused
interdependent relationships brought about by some project procurement systems, also contribute to construction claims and disputes.

- **Project factors**: unforeseeable site conditions, unrealistic planning & specification, incomplete or ambiguous project contract document, changes by the client, acceleration, different understanding and interoperation of project drawings, specification and contract, unfulfilled duties by project participants, poor documentation system, a deficiency in contracting staff and ‘force majeure’ are the direct causes for claims.

These complex issues contribute to the arising of various kinds of claims in construction project which leads to many projects ending up in major disputes. As a result, claims management has been developed for claims avoidance and claims settlement. The following sections first undertake a systematic analysis of the developments of claims management, and highlights the problems in current claims management procedures. Then, it investigates the basic principles and status of the claims negotiation domain. Finally it makes suggestions for the improvement of claims negotiation.

### 2.2 DEFINITION

Two essential terms: claims and claims management are defined here.

#### 2.2.1 Claims

The very word “claim”, in the context of construction, is not easy to define with precision in absolute terms.

- Arditi and Patel (1989) defines construction claim as “a request by a contractor for final compensation for additional work over and above the original agreed upon contract sum, or damages supposedly resulting from events not included/envisaged in the initial contract.”

- Powell-Smith and Stephenson (1993) and Jergeas and Hartman (1994) define construction claim, in a similar way, as “any application by the contractor for
additional payment or time, or both, that arise other than under the ordinary contract provisions.”

- Hughes and Barber (1992) define construction claim in a more specific way as “a request, demand or application for payment or notification of presumed entitlement to which a contractor, rightly or wrongly at that stage, considers himself entitled and in respect of which agreement has not yet been reached”.

These definitions capture the common nature of various construction claims. Expressed in general terms, a claim is a request for something “extra” for which no provision had been included in the original agreement. There are five general types of claims that a contractor can make against a client:

- **Claims under the contract**: arising because some provisions in the contract entitle the contractor to get payment for ‘loss’ or ‘expense’ and are made under the mechanism provided by the contract itself;

- **Claims for breach of contract**: arising out of the damages for breach of contract under Common Law;

- **Claims for the breach of Common Law in tort**: arising out of the damages caused by breaching the civil duty, mainly referred to as negligence;

- **Quasi-contractual claims**: claim for the value of services rendered or work performed where there is no contractual entitlement to payment;

- **Ex gratia claims**: arising out of the kindness of the client, sometimes referred to as a sympathetic claim (Powell-Smith and Stephenson, 1993).

Of these, the greatest number of contractors’ claims fall under the first kind of claim mainly caused by the client’s change orders, design error and different site conditions (Diekmann and Nelson, 1985; Bradley and Langford, 1987; Bordoli and Baldwin, 1998). These claims may be subdivided into categories which individually or collectively combined form the subject matter of most contractor’s claims (Hughes and Barber, 1992):

- Claims concerning the **existence or applicability of the contract**, 
- Claims concerning **critical time delay**,
• Claims concerning disruption of the work,
• Claims concerning payment,
• Claims concerning contract documentation,
• Claims concerning default, determination, forfeiture etc.

There are three general reasons for the institution of claims: default by the client and his representative, unforeseeable circumstance and contractor's underestimate. In practice, claims are often caused by the problems from the first two areas. For example,

• In the US, Diekmann and Nelson (1985) find that the most common causes of a contract claim are design changes and errors. A comprehensive analysis indicates that 46% resulted from design errors. An additional 26% are due to either discretionary or mandatory changes. The more volatile issues such as delay, different site conditions and maladministrations account for only 28% of the claims.

• In the UK, Bradley and Langford (1987) have identified the major reason as variations, complexity of project, poor contractual relations, deficiency of contracting staff, general recession, weather, and misfortune. This is consistent with the findings of Heath et al. (1994);

• In Canada, Bristow and Vasilopoulous (1995) address the primary causes of claims as unrealistic expectations, ambiguous contract documents, and poor communications. Semple et al. (1995) identify acceleration, restricted access, weather, and increase in scope as the major reason of claims. Bartsch and Jergeas (1997) think misunderstand the contract and client's misguided desire to reduce costs are the major causes of claims.

2.2.2 Claims Management

Construction claims are inevitable in any major construction project (Wilson, 1982). Yet, the undesirable consequences of claims can be minimised on projects if proper claims management steps are taken throughout the entire construction process. Claims management, as defined by Keane (1994), is “the process of employing and
co-ordinating resources to progress a claim from identification and analysis through preparation, and presentation, to negotiation and settlement.”

The principle of claims management is to ensure that the client pays a fair price for interfering with the contract in the execution of the work, and the contractor gets a fair pay for the extra work he has done (Bramble and Callahan, 1992). The essence of good claims management should always ensure that the claimant’s fullest entitlements are identified on a regular basis, and with adequate details to ensure that appropriate sums are paid through the interim payment mechanism (Brewer, 1993). Meanwhile, a contractor’s attempt to make profits from claims should be discouraged.

To achieve such objectives, an adequate management setup should be built up, which can prepare, evaluate, and settle claims effectively. The key points of the management setup are (Scott, 1992):

- The contractual basis for claims;
- Claim documentation and quantification;
- Presentation of claims;
- Negotiations and settlements.

Claims management is important for three main reasons:

- Firstly, it complements the contract to ensure a fair benefit share between parties to the contract. Almost every construction project contains uncertainties more or less, which may cause a change in the work. Claims management is the major approach to ensuring that the contractor gets compensation for the extra work he has done.

- Secondly, it facilitates collaboration between project participants and keeps a harmonious relationship between project participants, thus, it prevents construction productivity dropping or construction work being halted due to unsettled claims.

- Thirdly, it helps to prevent disputes. Any unresolved claim, through the claims management process, will become a dispute, often a potentially expensive one, and one of uncertain outcome. Most disputes in construction projects are the
The consequence of claims which have been rejected by the client following assessment and appraisal by the engineer. The consequence of this may well lead to a hardening of attitudes and to increasing acrimony which may end in arbitration or litigation. The unresolved disputes, through negotiation or other alternative dispute resolution methods, will finally lead to arbitration or litigation. In such a situation, either party may either give up his original position, or spend vast amounts of time and money on a legal tussle.

The effective management of claims is critical for the success of a project. Thus, research projects on claim management are of considerable interest to both academics and industry participants. To get a whole view of the field, it is necessary to investigate the former researches systematically.

2.3 HISTORICAL PERSPECTIVE

To seek answers to the problem, numerous research projects, courses and publications on various aspects of claims (e.g. Diekmann and Nelson, 1985; Scott, 1992; Levin, 1998; Chung et al, 2000) have been undertaken to investigate industrial practices and to explore the basic principles and procedures of claim settlements and dispute avoidance. Basically, these efforts are of two kinds: those that seek answers from principles and contractual legal issues at the pre-construction phase and those that attempt to solve the problems through claims management procedures at the construction phase (Vidogah and Ndekugri, 1997; Diekmann and Girard, 1995).

2.3.1 Basis of Claims Management: Starting Right

Regardless of the background circumstances, construction claims can only arise in the context of an existing contract. It is only the contract documents which can become the framework within which claims will be made. Thus, many efforts have been made on the development of contractual legal principles and other management theories at the pre-construction phase, which mainly include standard construction contract forms and conditions, risk management, and project procurement systems. The studies of these principles and theories are essential for avoiding construction claims and
disputes in the first place, and ensuring that claims management starts right if claims cannot be avoided. To fully understand claims management, it is necessary to explore the development of these theories and principles:

1) The Standard Contract Forms and Conditions

The paper, entitled 'The Condition of Engineering Contracts', which Rimmer published in 1939, led to the publication of the first edition of the ICE Conditions of Contract in 1945 (Cottam and Hawker, 1992). The main forms and conditions of the contract set up provisions for increasing the time period, and payment of loss and expense sums caused by unexpected events, both of which have to be claimed by the contractor. Similarly, the other popular standard forms of contracts such as JCT, FIDIC, CCDC and AIA have also been published with the aim of setting up the legal basis and principal claim provisions for claims management.

Since the end of the 1970's, an obvious change in the construction industry has been increasing project size, technical complexity and high risks, which have led to a large number of change orders. Moreover, the increasing time value of money, makes time extension claims more critical than ever. The old standard forms of contract have not changed in line with the development of the industry. According to Zack (1997), 84% of the respondents to a survey indicated that specifications of projects often required modification; 65% said that documents frequently required modification; and 55% stated that contract documents often needed significant changes. Many claims and disputes were caused by the ambiguous, incomplete and inadequate understanding of the terms of contract forms and conditions. As a result, improving the legal framework for claims has been a major focus of the industry for years. The overall objective is to (Vidogah and Ndekugri, 1998):

- get the contractual language right first time in new contracts;
- increase awareness of the likely construction terms by the courts and arbitrators;
- encourage the amendment of the standard terms of contract; and

---

1 ICE: The Institution of Civil Engineering  
JCT: Standard Form of Building Contract  
FIDIC: Fédération Internationale des Ingénieurs-Conseils  
AIA: American Institute of Architecture  
CCDC: Canadian Construction Document Committee
• expose popular misunderstanding.

The current standard contract forms and general conditions have been modified or rewritten several times based on the industry's experience and academic research; for example, the current FIDIC is the fifth edition and ICE is the sixth edition. Compared with the earlier editions, major improvements in the new contracts have mainly been on:

• legal issues in contractual relationships;
• the obligations of participants;
• project scheduling requirements;
• claim procedure;
• risk clauses; and
• dispute resolution mechanisms.

The major clauses of common standard forms of contract include (Wilson, 1982; Cyriax, 1982; Trickey, 1990; Powell-Smith and Stephenson, 1993; Jergeas and Hartman, 1994):

• Project scheduling requirements;
• Shop drawing submissions;
• Payment terms;
• Rates for time and material work;
• Variation in quantities;
• Authority and obligations of parties;
• Differing site conditions;
• Notification of delay;
• Notification of extra work;
• Time limit to correct default;
• Disclaimer;
• Contractor's status report; and
• Arbitration.
2) Risk Management

Many researchers (Levitt et al., 1980; Murdoch and Hughes, 1996; Cox, 1997; Hartman et al., 1998; Kumaraswamy, 1998; Sykes, 1999) have also identified risk allocation in standard contract forms and project contracts as an important factor in the occurrence of claims and disputes on projects. In construction projects, both parties take many risks, resulting in human error and the unexpected, such as design error, different geology condition, and acts of God. Any risk can result in the need for a change to the original contract and the contract must make provision for such changes to be made. If it does not do so, adequately and fairly, the changes will result in claims. Construction contracts are supposed to assign such risks to the parties who have entered into the contracts.

Currently, the contractor, under the pressure of competition, generally prefers to assume less risk while the client appears willing to push more risks to contractors during the tendering process, which is the major source of claims (Zack, 1997). Hartman et al. (1998) point out that the management of changes and claims is the management of risks. Project participants, especially, the client team should have a fair attitude to risk allocation in selecting the contractor and contract forms, estimating, scheduling and making detailed contract provisions. Equity in risk allocation in construction contracts and procurement systems will reduce the root causes of claims, thereby avoiding construction claims and disputes.

3) Project Delivery Systems

To avoid construction claims and to facilitate claims management, new project procurement systems, such as partnering, design-build, management contracting and construction management (CM), have been adopted in the construction industry since early 1990s with aims to build collaborative working relationships between project participants. Such approaches were also emphasised in the Egan report (1998) where an integrated project process is suggested.

Generally, it is believed that the design-build procurement system can reduce claims and facilitate claims management since it reduces conflicts and engenders closer
collaboration between designer and contractor. Similarly, the adoption of partnering or partnership also reduces the possibility of claims and disputes because risks are shared between project participants. On the other hand, the benefits of CM are often doubtful. For example, although theoretically the CM system is believed to be able to facilitate claims management, industry practice shows that in most cases construction managers are not willing to make decisions regarding claims so as to protect themselves against all possible litigation. The claims management process becomes more confused and complex under the CM system (Barrie and Paulson, 1992). Brewer (1983) notes that "the imported 'management' styles of contracting have done little to place in the hands of these firms the means to price and control their risks". Keane (1994) even states that the evolution of new project delivery systems, including the construction management process has done little to reduce the number and amount of claims. In fact, the existence of a construction manager has, on occasion, complicated the liability and damage issues that accompany a claim.

2.3.2 Claims Management Process: Staying Right

In spite of the extensive literature highlighting the above problems, there is little evidence of significant improvements in claims management. The continuous escalation in claims and disputes prove that the solutions offered by the research at initial stages of construction (i.e. standard contracts, risk management and project delivery systems) are inadequate in spite of its importance. There is, therefore, a need to look for approaches other than the above pre-construction theories and principles. The management setup for dealing with claims should also be studied.

Vidogah and Ndekugri (1997) point out that claims management and 'people' issues may be, at least, as important as having a clear understanding of contractual terms and equitable risk allocation, therefore, there is a need for complementary research into the claims management process. Diekmann and Girard (1995) also report, after studies of completed projects in the USA, that people and management issues may be more influential on the incidence of disputes than risk allocation and project characteristics.
However, most of the available literature on claims management, by way of defining remedial measures, do not go beyond general exhortations to the contractor to clearly identify the causes of claims and maintain adequate information to support claims (Vidogah and Ndekugri, 1998). Very few research projects are reported, which aim at auditing the whole claims management process in terms of precise deficiencies, their severity, and specific remedial measures.

The current framework for construction claims management is based on the industry practice of the past few decades. Figure 2.1 shows a typical claims management lifecycle based on the requirements of FIDIC (4th Edition). Levin (1998) summarises the management process as comprising:

- recognition and identification of changes or the causes of claims;
- notification to the Engineer and the Owner;
- systematic and accurate documentation;
- analysis of time and cost impacts;
- pricing;
- negotiation; and
- dispute resolution and settlement.

The current industry practice shows that the main challenge of effective claims management is not caused by the management process, but by the ineffectiveness of management activity at each step. The key aspects include how to justify a proposed claim, how to quantify and present the claim with full and detailed particulars, and how to negotiate the claim with the client and his agent successfully (Levin, 1998). These are discussed further below:

1) Claims Justification

Two factors are crucial to justify a claim. The first is the identification of the cause(s) of the claim; the second is the completed documents to support the claim. Identification of a claim situation is the first and the most important phase of the entire claim process.
Construction Contracts (Basis of Claims)
(Condition of contracts, special provisions, BOQ, drawings, specifications, overheads, rate of labour, material and equipment, etc.)

Events which cause claims
1) additional work not specified in the contract documents; 2) work different from that specified
3) work in a particular manner different from that originally anticipated; 4) work resulting from
changed, amended, or clarified contract drawings or specification; ...
21) strikes; 22) force of nature: ...

With a copy to the Client

Contractor
Continue the work, documenting all the events

Engineer/Architect
Examine the evidence and costs

Client

Concrete the possible claim

Reject, and explain reasons

Request detailed evidence and calculations

Agree
Inform

Submit the requested evidence and costing

Disagree, explain the reasons

Re-quantify the claim and re-submit

Arbitration or litigation

Dispute
Negotiation

Join negotiation if necessary

Reach agreement and inform the client for the payment

Accept, inform client for payment

Figure 2.1 Claims management procedure
Many claim situations arise out of subtle differences in field conditions, from job site delays or as a result of differences in contract interpretation. More often than not, disputes relate to nothing more significant than the argument about the liability of claims and interpretation of contract clauses (Sykes, 1999). Moreover, lack of solid supporting documents makes the justification more difficult. The following list shows the possible causes of claims identified and emphasised by many researchers (Diekmann and Nelson, 1985; Semple et. al, 1994; Cox, 1997; Levin, 1998):

- additional work not specified in the contract documents;
- work different from that specified;
- work in a particular manner or method that is carried out differently from that originally anticipated;
- work resulting from contract drawings or specifications that have been changed, amended, revised, amplified, or clarified;
- unanticipated work resulting from insufficient details in the plans and specifications;
- work required to be performed in one particular method when specifications allow two or more methods;
- work out of sequence;
- terminated, disrupted, or interrupted work, wholly or partially, directly or indirectly;
- joint occupancy;
- owner furnishing equipment late, in poor condition, or not suitable for the intended use;
- accelerated performance in any way, to regain schedule, to add men or materials, or to work overtime or extra shifts;
- following a new, different, or shorter schedule;
- relocating existing work because of lack of co-ordination or information;
- different site conditions;
- differences in contract interpretation;
- defective specifications;
- delays from the owner's acts or failure to act;
- unwarranted work rejection;
- increased inspection requirements, tests or quality control program;
• owner's failure to disclose information;
• strikes; and
• forces of nature.

Obviously, there are many events other than those listed above that may cause added performance costs or time. The key is to establish a methodology and database for systematic identification and tracking of all potential changes and claims from inception to resolution. Meanwhile, project personnel, especially site management personnel should be familiar with these clauses, contract documents and legal concepts so that when unanticipated events occur for which the contractor may not be responsible, the personnel will recognise and inform the engineer of the existence of the events as well as the causes of the events. Also, it is to the client's best advantage to recognise those causes of valid construction claims, and then do whatever possible to avoid the cause of the problem. Early identification of issues is important to permit timely resolution and to prevent small issues from growing into major project problems.

Documentation is extremely important for the justification of claims as well as the whole claims management process although it is time-consuming, and is rarely as rewarding as designing the project or supervising the construction (Wilson, 1982). Proper records are crucial for justifying the identified claims, analysing the project scope change and addressing the cost of the identified claim. This, in turn, will allow all parties to negotiate a fair settlement to the claim and dispute. Badger and Gay (1996) show that documentation is the first of the top ten lessons learned in construction contracting. Table 2.1 lists some of the regularly used documents to support claims.

2) Claims Quantification

The challenge of quantification in current claims management is not only to value the direct costs and delays caused by unanticipated events, but also to price the cumulative effects of these events. Ideally, the quantification of direct costs is relatively straightforward. With the agreed rates of labour, material and equipment,
quantity of impacts and general formulas, the costs of compensation and time extensions can be worked out through detailed breakdown analyses. However, arguments often arise regarding the rates of claim items and the quantity of the impacts of unanticipated events. For example, which rate should be used if several are listed in the contract documents, or how long the contractor should be entitled if both parties are responsible for a delay? In such cases, the value proposed by the contractor and the engineer can be dramatically different.

Table 2.1 Likely use of documents in claim presentation (Vidogah and Ndekgri, 1997)

<table>
<thead>
<tr>
<th>Document</th>
<th>Rank order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correspondence</td>
<td>1</td>
</tr>
<tr>
<td>Conditions of contracts</td>
<td>2</td>
</tr>
<tr>
<td>Schedules</td>
<td>3</td>
</tr>
<tr>
<td>Claim documentation</td>
<td>4</td>
</tr>
<tr>
<td>Site diaries</td>
<td>5</td>
</tr>
<tr>
<td>Timesheets</td>
<td>6</td>
</tr>
<tr>
<td>Photo</td>
<td>7</td>
</tr>
<tr>
<td>Specification</td>
<td>8</td>
</tr>
<tr>
<td>Minutes of site meeting</td>
<td>9</td>
</tr>
<tr>
<td>Records of delay and disturbance</td>
<td>10</td>
</tr>
<tr>
<td>Day work records</td>
<td>11</td>
</tr>
<tr>
<td>Revised drawing</td>
<td>12</td>
</tr>
<tr>
<td>BOQ</td>
<td>13</td>
</tr>
<tr>
<td>Analysis of tender</td>
<td>14</td>
</tr>
<tr>
<td>Level records</td>
<td>15</td>
</tr>
</tbody>
</table>

The most difficult task in claims quantification is to evaluate the cumulative effects of the events which contractors think they also deserve, such as loss of productivity, disruption and indirect costs. These items, by nature, are ambiguous and sensitive. Some of them are impossible to quantify with precision even with the best information available. Therefore, it is very difficult to reach a satisfactory solution between project participants. Part A of Table 2.2 shows that the number of claims for indirect costs is higher than for direct costs, whilst Part B indicates that claims for
such items are more likely to be disputed (i.e. the quantification of such items is highly disputed). The final amounts are often the result of negotiations between the contractor and the engineer.

Table 2.2 Number of claims for direct and indirect costs

<table>
<thead>
<tr>
<th>PART A</th>
<th>PART B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of claims</td>
<td>Rank</td>
</tr>
<tr>
<td>Site overhead</td>
<td>1</td>
</tr>
<tr>
<td>Loss of productivity</td>
<td>2</td>
</tr>
<tr>
<td>Loss of revenue</td>
<td>3</td>
</tr>
<tr>
<td>Financing costs</td>
<td>4</td>
</tr>
<tr>
<td>Equipment costs</td>
<td>5</td>
</tr>
<tr>
<td>Premium time</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Heads of claim likely to be disputed</td>
</tr>
<tr>
<td>Cost of preparing claims</td>
<td>1</td>
</tr>
<tr>
<td>Loss of profit</td>
<td>2</td>
</tr>
<tr>
<td>Cost of disruption</td>
<td>3</td>
</tr>
<tr>
<td>Head-office overheads</td>
<td>4</td>
</tr>
<tr>
<td>Interest and finance charges</td>
<td>5</td>
</tr>
<tr>
<td>Others</td>
<td>6</td>
</tr>
</tbody>
</table>


3) Claims Negotiation

Negotiations go on throughout the entire claims management process from the justification to the settlement of a claim either formally or informally. Although standard contracts like ICE do not suppose that claims will be settled by negotiation, in practice, many claims, especially those with greater uncertainty, such as material pricing, cost of disruption, and indirect costs are settled through negotiation between the contractor and the engineer (Powell-smith and Stephenson, 1993).

Resolution by negotiation retains the highest level of control because it is the parties themselves who determine the terms of the agreement. They decide how the matter will be resolved, not some third party. This is the most common and most cost-effective method to reach an agreement. Moving from negotiation to mediation to arbitration or litigation results in a reduction of control along with an increase in costs at each step. Field et al. (1993) summarise the importance of claims negotiation as follows:
While change order negotiation is sometimes approached in a haphazard manner, an organised negotiation process and consciously selected strategies can minimise costs to the owner. From case to case, claims like back charges and change orders have a trend of growing larger as time goes by. The early settlement has the benefit of eliminating the claim creep.

In addition to minimising costs, a successful negotiation program can also provide a non-adversarial project environment in which disputes are resolved quickly, fairly, and to the satisfaction of all parties, so construction can continue unimpacted by the issues that invariably arise.

Successful negotiation is also the first line of defence in avoiding more costly and time consuming dispute resolution procedures, such as arbitration and litigation. Vidogah and Ndekugri (1997) emphasise that negotiation is of paramount importance in preventing disputes. In fact, even other alternative dispute resolution (ADR) methods all involve some form of negotiation. The intent of most ADR is to resolve disputes in forums that allow for negotiation between parties most qualified to resolve the technical portions of the dispute.

Basically, there are two major kinds of claim negotiations: qualitative negotiation and quantitative negotiations.²

- **Qualitative negotiations** are about the entitlement of the contractor to a claim which is mainly regarding the justification of claims based on the identified causes, liabilities and contract conditions. Such negotiations are essential and highly evidence-oriented with limited room to bargain.

- **Quantitative negotiations** are about the amount of compensation for the entitled claims, which commonly deal with the amount or the quantity of claim items, rates of material or labour, and indirect costs. Quantitative negotiations go into much detail and are time-consuming since there is much more room to bargain, and both parties try to maximise their benefits through this negotiation process. Thus, the current studies on claims negotiation are mainly interested in quantitative negotiations.

²A third kind negotiation involves the items claimed on an “ex-gratia” basis. Such negotiations will be introduced for discussion at the final settlement.
Spittler and Jentzen (1992) suggest that claims and disputes are best resolved by participants who can gather, exchange, and discuss all relevant information objectively, make persuasive arguments that influence other participants to change positions, and have the budget latitude to make binding agreements. This ideally should occur at the organisational level as close to the claims and dispute as circumstances will allow. In practice, negotiations are normally started at a lower level between the contractor and the engineer's quantity surveyors. The project manager and the engineer will step into the negotiation if problems cannot be solved between the quantity surveyors. If agreement still cannot be reached, the contractor may contact the client in the hope that the client joins the negotiation directly and solves the problems.

The difficulty of claims negotiation lies in the fact that negotiations are not only determined by the work conducted at previous stages of claims management: identification of claims causes, interpretation of the clauses of the contract or conditions, documentation, and quantification, but also decided by each party's expectation, interest and positions. Furthermore, the expertise and domain knowledge of the negotiators' personnel also influence the outcome of negotiation.

According the result of industry survey undertaken in this study, the following general guidelines to claims negotiation are emphasised by project managers, which also have been discussed in some other researchers' work, such as Smith (1992) and Levin (1998):

- **Get enough evidence to support negotiation:** negotiation provides a good chance for both parties to examine the evidence provided by each party. Sound evidence, in turn, will play a unique and crucial role in supporting the argument.

- **Define areas of negotiation:** both parties should recognise the issues where there is the possibility of an agreement. This sets a tone for the negotiation - the purpose is to get a resolution. It also has the benefit of limiting the issues that can be brought up as negotiation progresses.

- **Have a plan:** both parties should focus on achieving a reasonable financial goal and preserving the relationship with the opponent. They should have a settlement
range in mind, but be prepared to react to what actually happens in the negotiation.

- *Listen to the other side and be ready to make some concessions:* although both parties' different perceptions on the negotiation items may be supported by their own evidence, they should be prepared to compromise. They should also be prepared to back off positions that are taken only for the purpose of gaining leverage. Furthermore, an appropriate negotiation style is important for both parties in negotiation.

- *Have a proper negotiation team:* an experienced and reasonable negotiation team with clear authority is always important for successful claims negotiations.

A detailed analysis of these aspects of the claims negotiation will be conducted in the next section.

### 2.3.3 Computer Supported Claims Management

The speed of the development of information technology over the last two decades has brought about major changes to the construction industry. The applications of information technology also provide opportunities for the improvement of construction management principles. The developed management principles and information technologies strongly support the claims management process.

#### 2.3.3.1 Management Principles

The improvement of planning and scheduling technologies in project management, especially the application of Critical Path Method (CPM) strongly supports delay claims by sorting out the overall impact of unanticipated events on the schedule. Delays resulting from unanticipated events can be addressed clearly by analysing their effects on the critical path while the effects on non-critical paths are excluded.

In this approach, an "as-planned" CPM schedule is first built based upon the approved project schedule; then a network model known as the "impact schedule" is built to superimpose various schedule impacts upon the as-planned schedule including
excusable, compensable, non-compensable and concurrent items. Finally, delays in the critical path can be identified and the revised overall completion date for the project can be calculated. Input to the schedule will involve identifying and quantifying the individual delays and determining the responsibility and entitlement of the parties. This schedule adjusts the As-Planned schedule to reflect the overall effect of the impacts and delays (Barrie and Paulson, 1992).

With the development of information technology and the knowledge transformation from other industries, many new management concepts and principles, such as concurrent engineering and the application of neural network to support decision-making are being adopted in construction management. These applications are helpful for claims avoidance and claims management, directly or indirectly.

2.3.3.2 Engineering Software

The wide application of information technology in engineering design and project management provide more opportunities for the improvement of claims management.

- The development of various technology software in architecture, structure, geology, and building service largely improves the quality of design. Meanwhile, the applications of AutoCAD change design from paper to electronic drawing. All of these help to reduce design mistake;

- The application of MS. Word Processing, Excel Spreadsheet and Database systems transfer all the old paper documents into electronic documents. This strongly supports record-keeping at both site and head office, which help to justify and present claims. Keane (1994) has developed a computer-aided systematic approach using these essential tools to facilitate time extension claims;

- MS. Project, Primavera and other management software developed based on CPM, are used to plan, monitor and control project progress. Moreover, they are also adopted to predict, simulate and quantify the delays and costs of unanticipated events;
• The Intranet and Electronic Data Interchange (EDI) had major impact in facilitating communication between project participants in a timely and efficient way. They also facilitated record-keeping;

• Expert systems and decision supporting systems, such as SuperChanges and SuperDelay (Diekmann and Kim, 1992a) aim to work as advisors to advise field engineers and project managers about the legal consequences of certain contract conflict situations. Site engineers have to make many quick decisions, some of which are legal, mainly based on their experience. An imperfect decision can lead to costly claims and disputes. Without adequate knowledge of contract laws, field engineers might be unaware of the accepted legal basis or consequence of their decisions. A well-designed expert system can help site engineers improve the efficiency and accuracy of decision making. Moreover, a sophisticated expert system can also be a good help to project managers in the claims management process. Further examples of computer-aided claims negotiation can be found in Levitt (1987), Adams (1988), Bubbers and Christian (1992), Diekmann and Gjertsen (1992b), and Alkass et al. (1995).

2.4 PROBLEM

Despite the extensive studies on claims management theories and practices, the increasing incidence of claims and disputes imply that the current claims management principles and process are ineffective in meeting industry requirements (Diekmann and Nelson, 1985; Vidogah and Ndekugri, 1997, 1998). Due to the volatile nature of construction projects, it is also unrealistic to expect that claims can be avoided or resolved by a single principle or method. The improvement of construction claims management will be a long strategic task for the industry. The major deficiencies of current claims management practice are in the following areas:

2.4.1 The Awareness and Interpretation of Contract Terms and Provisions

Vidogah and Ndekugri (1998) identify inadequate understanding of the terms of standard forms as one of the two reasons leading to the non-entitlement of contractors' claims which in turn is the top reason for the failure of claims. Keane
(1994) also explains that differences in awareness and interpretation of contract terms is still a major contributor to problems in claims management. Contractors, engineers and clients regularly fail to agree on the basis of a claim and how it should be settled. The reasons are:

- The understanding of the terms of the contract is still inadequate in spite of the large volume of information dedicated to interpreting them. Understanding is usually based on experience and common sense. Sometimes, what is meant to be said may be the determining factor instead of what is actually written in the contract. These interpretations may lead to disagreements, disputes, and occasionally, litigation (Thomas et al., 1994);

- Contracting parties frequently interpret contract terms from their own perspectives, in an ad hoc manner and for their own benefit. The same contract clauses are regularly interpreted differently; and

- Contracts themselves contain ambiguous or unfair provisions. Contract terms alone are insufficient to clearly apportion risks or responsibilities between contracting parties. This holds true not only for what is stated explicitly in the contract documents, but also for what is implied.

2.4.2 Inadequate Information and Documentation

Many researchers have identified inadequate information and poor documentation to support claims as major problems in current claims management practice. The earlier Wood report (1975) made this very point when it emphasised the lack of factual evidence as a prime cause of delayed payment and protracted disputes. Scott (1992) points out that perhaps the biggest failing on the part of contractors when dealing with claims is the lack of sufficient recorded data on the effects of delaying and disrupting events. Wilson (1982) and Badger and Gay (1996) all stress the lack of initial records, such as minutes of meetings, correspondence, progress reports, status logs, photographs, records of delay and disturbance, and revised drawings, as the major reason for the failure of claims. Many contractors' management information systems are ill-designed to support claims. Records are kept either in an inaccessible way, particularly after project completion, or may be incomplete or designed for other
purposes even if available. Vidogah and Ndekgri (1997, 1998) conclude that the reasons for these problems are:

- a culture of bias against paperwork on site operations;
- poor design of recording systems;
- the paper-based nature of most of the relevant information; and
- poor resourcing of the claims management role in contracting organisations.

2.4.3 Lack of Effective Claims Management Tools

Compared with the number of research projects on the legal bases and principles of claims, the efforts seeking to develop effective tools to support claims management activities are few. As a result, claims management activities rely on general project management techniques. Very few project management tools, especially systematically designed, are appropriate for claims management. For example, inadequate record keeping has been recognised for a long time as a major problem area in claims management. There is virtually no chance of improvement without the application of information technology. On the other hand, industry practice shows that claim documentation is still perceived to be woefully inadequate in many cases even with the support of IT. The increasing power and affordability of information technologies do not naturally lead to an improvement of the claims management process. There is a lack of systematic analysis on how best to apply these technologies; that is more crucial for claims management.

2.4.4 Inefficient Claims Negotiation

Although the cases are quite often diverse and project-specific, claims negotiations are normally hard and time-consuming. Most project managers consider claims negotiation as the most time and energy-consuming activity in claims management (Hu, 1997). The reasons for this include the diversity of intellectual background, the many variables involved, complex interactions, and inefficiencies in the negotiation process (Zack, 1994). For example,
• the consequences of negotiations are directly related to financial gain or loss. The engineer and the client typically respond in a tough and unyielding manner in claims negotiation due to the negative perception of claims;

• negotiations are influenced by many internal and external factors (see Section 2.5);

• the negotiation items, such as loss of productivity and disruptions, are generally ambiguous and sensitive

• negotiation is more a human-oriented process than a pure technical process. A negotiator’s personal abilities, attitudes and negotiation strategies, in many cases, determine the result of a negotiation. Unfortunately, most construction personnel do not have enough negotiation expertise. Emotion, irritation, and relationship considerations often strongly influence negotiations.

Besides the above factors, problems such as inadequate negotiating expertise, improper negotiation strategies and complex human factors also contribute to the inefficiency of claims negotiation.

Although all the problems identified above are harmful for claims management, some of them like the awareness and interpretation of contracts and documentation have generated considerable attention in the industry. In contrast, others such as the claims negotiation process are seldom being studied. This research, unlike most previous research projects, will focus on the study of claims negotiation, and aims to develop a system which can improve its efficiency, and resolve other essential problems in claims negotiation.

2.5 CLAIMS NEGOTIATION

Despite the importance of claims negotiation in claims management, very few studies have been done in this field. Most project participants believe that negotiation is only a kind of game where people lose and gain through bargaining. The following sections provide a detailed analysis of the essential principles of claims negotiation,
and cover the generic model of claims negotiation, existing problems and suggestions for improvement.

2.5.1 Generic Model of Claims Negotiation

The generic model shown in Figure 2.2 illustrates the major characteristics and elements of claims negotiations, which human negotiators have to analyse before negotiation, to provide necessary background information. The model represents negotiation parties, structures, relationships, and attributes involved in claim negotiation. It consists of three major aspects, i.e. negotiation elements, negotiation process, and negotiation outcome.

2.5.1.1 Claims Negotiation Elements

Claims negotiation contains three major elements, i.e. construction project, claim items and negotiation participants. In addition to these, some other background information also influence the outcome of negotiations. For example, arbitration or court decisions related to construction claims will provide perspectives on the likely outcome should negotiation fail and the issues ultimately have to be resolved through litigation. Such information is useful for project participants if they prefer to end the negotiation with a conflict.

1) Project elements

The project data provides a background and environment for claims and external constraints for the resolution of negotiations. Two kinds of elements are considered essential:

- social-economic and natural environment in which the project is;
- project contract and procurement system, such as: the standard conditions of contract adopted and the organisational structure of the project; and
- project information, such as: the nature of the project, design quality, and each party's expectation, status and requirements for money, time and quality of the project.
Negotiation Process

Negotiation Planning

Negotiation Process

Negotiation Outcome

Negotiation Elements

Project
- Nature of work
- Contract form
- Procurement system
- Design
- Project status
  (Cost estimate via status
  Schedule via progress)
- Socio-economic condition
  (loan, labour market, material
  & equipment supplier)
- Natural environment

Claims
- Causes of claim
- Nature of claim
- Claim items (time or money)
- Stage of claim (entitlement or
  quantity)
- Evidence
- Possibility of future claims

Client

Interests
- Quality
- Value
- Progress
- Safety
- Environment

Expectations
- Functions
- Cost
- Time
- Quality
- Less trouble

Attitudes
- Co-operative
- Competitive
- Punitive

Engineer

Interests
- Reputation
- Quality
- Relationship
- Progress
- Safety

Expectations
- Low responsibility
- Reputation
- New project

Attitudes
- Competitive
- Co-operative
- Combative
- Individualistic

Contractor

Interests
- Profit
- Time
- Relationship
- Reputation

Expectations
- Cost
- Time
- Extension
- Long term
- Benefit

Attitudes
- Competitive
- Co-operative
- Individualistic

Figure 2.2 Generic claims negotiation model (Developed from Pena-Mora and Wang, 1998)
These factors decide the possible nature, frequency and future aspects of claims; influence project participants' negotiation interests and attitude; and finally affect the outcome of negotiation. For example,

- if labour cost is high in the local labour market, the contractor may assign a high weight to the labour cost claims;
- if the project adopts different contract conditions or procurement systems, the claims procedure may be different;
- if the project is under high time pressure, more weight may be given to time issues by the client; or
- if a party finds that a certain kind of claim may frequently occur in the future, he will put more weight to the claim even though it is minor at the current stage.

2) Claims elements

Claim factors reveal the nature, causes, stage and scope of claims. They are the hard variables of the negotiation, as they are quantifiable and expressible. Examples could be monetary compensation, time extension or loss of productivity caused by the variation of the thickness of a concrete wall. Claim items are the content of negotiation and are typically represented by the offers and counteroffers made by the participants throughout the negotiation. These claim items can be classified into two kinds:

- In practice, a significant portion of the difference between the contractor's quotation and the engineer's estimate relates directly to factual issues, such as the scope of the change, the quantities involved, labour wage rates, material pricing, equipment costs, and contractual requirements such as mark-up percentages, unit prices, etc. Since these issues are factual in nature, it is relatively easy for willing parties to examine the facts together and reach agreement.
- After agreement is reached on factual issues, the negotiation should proceed to issues with greater uncertainty, such as labour productivity, loss of revenue, and indirect costs. These items are often the argued points. Moreover, it also reflects both parties' expectations of the claims. For example, the contractor may exaggerate his demands on these items while the engineer is not willing to accept
any reasonable recognition of risk. In practice, the productivity data from standard estimating guides often fails to address the real situation, and may need to be adjusted when the underlying assumptions do not fit the change. A compromise is often necessary as a result of the uncertainties involved.

Besides the nature of claims items, the supporting evidence for the claims (such as: the contract clauses, site records and reasonable cost estimating and schedule review) is another crucial determinant of the outcome of negotiation.

3) Participant elements

Participant elements represent different interests, expectations and attitudes of project participants in the claims due to their different roles and positions in the project.

- **Interests**: In claims negotiation, each participant carries a set of basic interests which are simply the underlying need or preference that a negotiator carries with him to the negotiating table. These interests are determined by project and claim factors, which can be project, professional, or person-specific. Interests are the soft variables of negotiations, and represent the more qualitative aspects of the participant’s overall descriptions. These may or may not be represented by any particular set of visible positions. The interests of the negotiator determine his or her expectations and attitudes.

- **Expectations**: Closely related to interests, expectations are also soft variables of the negotiation. Expectations determine which kind of strategy a negotiator will adopt in negotiation. The negotiators’ expectation can also be identified by the utilities which negotiators assign to different negotiating items.

- **Attitudes**: Attitudes represent the willingness of the participant to negotiate or willingness to reach an agreement. Although such a variable may have no real bearing on the facts of negotiation, (i.e. a time extension or change order value), they may have a profound effect on the way in which the negotiation is conducted. The attitude of the participant may also reflect which kind of negotiator the
individual is. The style of a negotiation can be categorised depending on what kind of strategies and tactics are used.

Besides each party's characteristics, a negotiator's personal factors also influence the outcome of the negotiation.

- **Negotiator's personal skill**: Negotiations, by nature, are highly human-oriented activities. Besides the factors discussed above, a negotiator's personal ability, information and expertise influences the outcome of negotiation. Ideally, a negotiator should have (Hughes and Barber, 1992):
  a) personal capabilities and clear authority;
  b) knowledge of the industry and the project;
  c) necessary information about the specific claims item;
  d) detailed preparation of the negotiation;
  e) a clear objective and negotiation strategies; and
  f) perceptions about his/her opponent's objective.

- **Negotiator authority**: Another issue that needs to be addressed for successful claim negotiation is that every participant defines his negotiator's authority. It is not possible for the contractor and the engineer to participate in all the negotiations. Quantity surveyors are often appointed to negotiate on behalf of both parties. The empowerment may range from 'settle your own way and at any cost' through a series of gradations to 'do not settle anything but report back'. Two points need to be emphasised in this respect:
  a) First, the awarded authority should match the negotiator's position, capability, specialist negotiation items, and negotiation circumstances;
  b) Second, while each negotiator should have clear authority and limit, it is also highly desirable that those engaged opposite each other in a single phase or segment of negotiation should have matching authority (Turner, 1989). It will be frustrating for one to have the authority to settle when the other party has to report back to a higher authority.
2.5.1.2 Individual Roles in Negotiation

Besides the above factors, each party's negotiation strategies are also determined by factors such as responsibility, bargaining power and time issues. These factors, in turn, are determined by each participant's role, position and interests specified in the contract.

1) The Client: In most cases, the client does not directly get involved in the negotiations. However, when he joins the negotiation, the client, as the employer, will normally have more bargaining power than the contractor and the engineer. This is especially true in cases where the contractor expects to keep a long-term business relationship with the client. The client is generally tough in claiming money issues. However, he may make concessions if the unsettled claim delays the whole project, or in some other cases, he may just want to keep the contractor working more faithfully.

2) The Contractor: Motivated by the objective to make as much profit as possible, the contractor generally will catch all the possible opportunities for a claim. This puts him regularly in a passive position in negotiations because of his exaggeration of the claim without adequate evidence to support it, or his imperfect work with regard to the claim items. On the other hand, if the claim is sound, the contractor has a tremendous asset in setting the pace and direction of the negotiations since he has a larger degree of freedom in making his proposal and accepting a settlement (Levin, 1998).

3) The Engineer: The roles of the engineer, both as an agent of the client and an independent professional expert, puts him in a conflicting position in claims management and claims negotiation. As a professional expert, the engineer should implement various terms of the contract, and act impartially between the client and the contractor in the negotiation. On the other hand, as an agent of the client, he has to negotiate for the client and therefore is biased against the contractor. Moreover, the engineer's attitude is also influenced by his own benefits. One example is that the engineer tends to discourage any claims caused by him.
Table 2.3 lists some of the major influential factors for each party summarised from the work of the following researchers: Hughes and Barber, 1992; Smith, 1992; Spittler and Jentzen, 1992; Scott, 1992; Just and Torone, 1997; and Levin, 1998.

### Table 2.3 Factors influencing negotiations between project participants

<table>
<thead>
<tr>
<th></th>
<th>Factors that Hamper Negotiation</th>
<th>Factors that Facilitate Negotiation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Contractor</strong></td>
<td>• high ambitions;</td>
<td>• willingness to maintain a good relationship;</td>
</tr>
<tr>
<td></td>
<td>• global claims;</td>
<td>• willingness to compromise due to the weakness in claims management;</td>
</tr>
<tr>
<td></td>
<td>• lack of evidence;</td>
<td>• concentrating on future prospects of work or compensation (new claims or items)</td>
</tr>
<tr>
<td></td>
<td>• wrong calculation; or</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• poor presentation</td>
<td></td>
</tr>
<tr>
<td><strong>Engineer</strong></td>
<td>• willingness to protect himself against any harm;</td>
<td>• high expertise in construction management;</td>
</tr>
<tr>
<td></td>
<td>• discouraging any claim caused by the failure or act of the engineer</td>
<td>• consideration of reputation</td>
</tr>
<tr>
<td><strong>Client</strong></td>
<td>• willingness to pay less for more work;</td>
<td>• consideration of the entire project progress;</td>
</tr>
<tr>
<td></td>
<td>• discouraging claims</td>
<td>• involvement of client</td>
</tr>
</tbody>
</table>

Note: besides the factors listed above, some common factors also influence the results of negotiations, (e.g. the involvement of senior management in the negotiation, or the pressure from senior management) may force the early settlement of negotiations.

2.5.1.3 Negotiation Process

The claims negotiation process consists of three stages: negotiation planning, negotiation process and negotiation outcome.

1) Negotiation Planning

Negotiation preparation is crucial for the success of claims negotiation. Many of the problems of current claims negotiation process are related to poor planning and
preparation of negotiation (Smith, 1992; Field et al., 1993). Both parties need to spend time and effort on collecting data, gathering pertinent prices, establishing objectives, identifying the negotiation zone, evaluating the proposal, anticipating and analysing the opponent, formulating a definitive and defensible negotiation position, and building flexible strategies. According to Field et al., (1993), Levin (1998) and the result of the industry survey in this research, the key points include:

- **Make an independent estimate of scope and cost:** if it is determined that entitlement exists, an independent engineering cost estimate should then be prepared by both parties. This will ensure that all issues are identified, including those omitted from the contractor’s quote. To whatever extent possible, the engineer should use the contractor’s actual productivity rates for similar project work. After the estimation, all significant differences in scope, productivity, and pricing are listed for negotiation.

- **Evaluate strengths and weaknesses of both parties:** an honest evaluation of the strengths and weaknesses of both parties’ position is critical before negotiations begin. It is advantageous for each party to evaluate the anticipated positions of both parties, develop responses, agree upon the responsibilities, presentation methods, positions, and tactics which will be utilised, and identify any fallback position which may be available should negotiations not lead to agreement. The availability of a satisfactory fallback position for each party will ultimately help to determine the size of the concessions that the party must be willing to make to reach settlement. Two major issues are decided at this stage:
  
a) **Establish objectives** (what is the negotiation zone?)

  - which objectives cannot be compromised under any circumstances?
  - which can be compromised and to what extent? and
  - which objectives are expected to be compromised or dropped totally?

b) **Anticipate position of the opponent:**

  - how bad is the need for the work?
  - is there time pressure for an agreed price? and
  - is there competition for forward priced changes (can this work be transferred to another contractor?)
• **Consider the social psychological aspects:** since negotiations are conducted by human beings, thorough preparation for negotiation should include optimisation of the myriad of psychological factors involved. In order to prevent intimidation of either party, it is desirable to provide an approximate match between the sizes of the negotiation teams representing each party at the negotiation and between the authority levels of the representatives of the parties. It may also be desirable to choose representatives who can work well with the representatives chosen by the other party. Also, it is helpful for a party to estimate the other's exaggerated amount for certain claim items.

• **Build flexible strategies:** based on the above analysis, each party will plan his negotiation strategies and alternate strategies in case the primary strategies have to be abandoned.

• **Decide on the negotiation form:** although periodic negotiation meetings are necessary to ensure that issues are resolved in a timely manner, these meetings are not always the best forum for negotiating every issue. Before negotiating, it is desirable to select an optimal forum for negotiation and decide whether informal discussions with one or more members of both parties would be likely to be more productive than formal negotiations. The result of any informal negotiations can then be confirmed during a formal negotiation meeting. It may also be desirable to manage the timing of negotiations to optimise the productivity of negotiations.

Figure 2.3 illustrates such a general process and the major factors which negotiators need to consider before claims negotiation. It needs to be mentioned that the contractor and the engineer may have different interests. The engineer is normally more interested in analysing the contractor's claim amount and supporting evidence as well as the related contract documents, whilst the contractor focuses more on how much he should ask from the engineer. Unfortunately, both Smith (1992) and the results of the industry survey undertaken in this research show that project participants seldom conduct negotiation preparation systematically.
Figure 2.3 Negotiation planning process (improved from Smith, 1992; Levin, 1998 according to the industry survey in this research)
2) Negotiation Process

During negotiation, the contractor looks for the greatest sum possible whilst the engineer, despite his contractual position, will be looking to reduce the amount being claimed. Moreover, their negotiation styles, strategies and tactics also depend on the project situation, claim items, participant’s attributes and the personal approach of the negotiators. Both the contractor and the engineer try to influence, persuade or press the other side to accept his proposal by providing backup evidence, communicating information, adopting proper tactics, and making necessary concessions.

A typical negotiation iteration is: after receiving the contractor’s quotation and the explanations, the engineer analyses the quotation, explanation and supporting evidence. He then tries to point out the problem of the quotation, and re-evaluates his own estimate in a reasonable manner as necessary to correct errors, omissions, or otherwise adjust the estimate wherever required, or vice versa. Two kinds of reasons will push a party to make concessions:

- a party is persuaded by another, by showing sound evidence, that his original estimate about the claim is wrong. This is always easy to handle; or
- no obvious mistake is found in both parties’ estimate, especially in the contractor’s. The difference lies in both parties’ different perspectives on interpretation or quantification. This is the common case in claims negotiation. Both parties may make concessions for the purpose of reaching an agreement. The concession amount is decided by both the key negotiation features determined at planning stage and the situation in the negotiation.

Scott (1992) describes the contractor’s negotiation tactics for large claims as:

- Initially, the contractor should ascertain the engineer’s position, without making commitments, to get an idea where he stands on an overall basis. He should then study the differences to see where more work can be accomplished, and how much work might be needed to get the settlement closer to an acceptable amount.
• After strengthening his position with additional data, the contractor is ready for a second negotiating session. This time the contractor should make an all-out effort to achieve his goal and begin to make compromises on individual items.

• If a satisfactory compromise cannot be reached, the negotiation sequence must be repeated until either an agreement or impasse is reached.

• Leave some bargaining room and expect some give and take.

In practice, the success of negotiation is largely dependent on the ability and willingness of the parties to work together creatively and constructively to resolve claims, even if the initial negotiation has reached a stalemate. If both parties do not want to make any concession on a single item, both parties may try to find other possible approaches to resolve the problem. Compromises can also be achieved through (Ren et al., 2001a):

• trade-off between different items of the same claim (e.g., the trade-off between labour cost and loss of productivity for the same claim);

• trade-off between different compensation methods (e.g., the trade-off between money and time);

• trade-off between different claims (e.g., the trade-off between claim A and claim B);

• relieve negotiation constraints (e.g., conditionally agree some items which were not accepted at the beginning; or permit a higher level of management, empowered with greater authority, from both parties to join the negotiation, thus providing a different perspective and perhaps a broader view of the issues); or

• consider creative solution alternatives (e.g., construction contracts often include requirements that, for one reason or another, are found not really necessary during construction, but add costs for the contractor. Under specific situations, the engineer may give away such items).

3) Negotiation Outcome (Post-negotiation stage)

The outcome represents the final settlement determined by the project environment, claim elements and each negotiator's individual factors through the negotiation process. The outcome can be either an agreement or a conflict deal. For each party,
the outcome must be acceptable and reasonable, although it may not be the best one. If an agreement is reached, it is compulsory for both parties. Meanwhile, the high frequency of claims negotiation allows the contractor and the engineer to learn each other's negotiation habits towards different claim items.

2.5.2 Problems

Although, ideally, a successful negotiated settlement of construction claims should be characterised by fairness, efficiency, wisdom, and stability (Spittler et al., 1992), in practice, it is far from true. From case to case, claims negotiation often becomes a very long and drawn out process (Botha, 2001). This is caused by a number of problems associated with project and claims management, and the negotiation process, such as: the client team's negative attitudes towards the contractor's claims and the contractor's high ambitious to claim, the complex and ambiguous nature of some claim items, improper or unclear contract clauses, and poor documentation systems amongst others. This research focuses more on the key problems that directly contribute to the inefficiency of claims negotiation, as discussed below.

1) Problems in the Claims Management Procedure

The claims management procedures contribute to the inefficiency of claims negotiation. For example, the involvement of the client is currently very low in claims management (Vidogah and Ndekugri, 1997). The late involvement of the client and the engineer's conflicting role as an independent professional and client's representative have been recognised as major factors contributing to the inefficiency by the European Construction Institute (ECI, 1992). In cases where claims are caused by the engineer's failure or inaction, the engineer is likely to discourage the claims, and to deal with the claims in a partial manner by taking advantage of the low client intervention. This may finally increase the difficulty of negotiation which may, in turn, increase the likelihood of disputes. Therefore, ECI (1992) recommends earlier and greater client involvement in claims management. Furthermore, the problems from the contractor, such as lack of supporting records, unclear causation and consequence analysis, and exaggeration of claims also provide opportunities for the engineer to behave improperly.
2) The Preparation and Negotiation Process

Currently, both the preparation and negotiation process are rather time-consuming. This is caused by the necessary negotiation preparation (evidence-collection, opponent position anticipation, and negotiation objective and strategy analysis) and bargaining process (offer, evaluation, and counteroffer), as well as many other activities, which sometimes take more time than the negotiation itself. For example:

Before negotiation, claim documents have to be specially presented for negotiation; there is quite a substantial delay between document submission and the discussion of the claim. Negotiations are seldom held immediately for one claim after the documents are submitted. In most cases, the claim will be discussed at a progress meeting or special meeting where many claims are discussed; and negotiators need to be appointed and brought together for the negotiation. In cases where project managers or other key project personnel act as negotiators, negotiations are often delayed due to the absence of these people.

During negotiations, unrelated arguments such as site management, quality or site safety are often used to pressurise an opponent; a party may expect to benefit by waiting or delaying until the other side is emotionally exhausted; or neither side wants to make a concession first or easily.

Moreover, Smith (1992) identifies improper negotiation styles and lack of planning as the main reasons for inefficient construction negotiation. Field et al. (1993), Just (1993) and Zack (1994) outline some of the improper negotiation styles as:

- portraying the other party as unreasonable;
- adding emotion to meetings and correspondence;
- negotiation without being in command of the facts;
- relying on the recollection of people who were involved;
- psychological intimidation;
- either win or lose strategy, lack of necessary flexibility;
stubborn negotiation style;
- making concessions for the sake of a relationship;
- bargaining instead of negotiating;
- establishing a fixed objective instead of a range;
- failing to choose the right negotiation team members;
- failing to plan negotiation;
- unclear authority;
- failing to document negotiation;
- negotiation team members playing good guy/bad guy game; and
- requesting sympathy.

3) Human Factors

Negotiation involves many human factors in addition to the pure technical issues. A negotiator's personal abilities, attitudes, expertise, power and position in negotiation, in many cases, determine the result of a negotiation. Unfortunately, most construction participants are not very good negotiators for a variety of reasons (lack of expertise, lack of time, dislike of the negotiation process, etc.). They often realise that their negotiation skills are not as sharp as they could be (Botha, 2001). By investigating claims and disputes resolution in eight countries and areas, Fenn et al. (1998) reports that most negotiators get their negotiation knowledge only through the claims negotiation practice. Most claim negotiations are conducted in a heuristic way. Unnecessary concession and stubbornness are common mistakes, especially when negotiations become emotionally charged, which makes the negotiations harder and inefficient.

As a result, there are many individuals and firms working as professional 'claims consultants' that offer claims and disputes negotiation service in one form or another. Unlike a mediator, the claims consultant's duty is to advise project participants how they should seek to make money out of the alleged mistakes or shortcoming of other participants (Botha, 2001). The typical idea is "with my expertise in your corner, you would gain a significant edge in your negotiations to tip the balance of power
dramatically in your favour.” On the other hand, project managers, according to the industry survey in this research, responded that the involvement of a claims consultant could make the negotiation more complex and fierce.

Zack (1994) points out that “the inefficiencies in negotiation make claims resolution much more difficult and adversarial, and may delay resolution or, in the worst case, lead to expensive litigation.” Spittler et al. (1992) also warn that if incentives and mechanisms that can encourage efficient and effective job site negotiation are not made an integral part of the claims management process, disputes will inevitably escalate into serious confrontations. Thus, it is necessary to develop a methodology to facilitate claims negotiation so as to reduce the tremendous time and human resources invested.

2.5.3 Possible Solutions

Essentially, since many problems of project management and claims management contribute to the inefficiency of claims negotiation, the improvement of project management and claims management will be helpful for claims negotiation. Examples of these aspects can be:

- adopting an appropriate contract form, and implementing contract language which clearly establishes the parameters under which changes and claims will be identified, documented, and negotiated and providing a means by which work may proceed;
- adopting partnering or design-build procurement system to relieve the negative attitude of the engineer and the client towards the contractor’s claims;
- improving the documentation system to provide more sound evidence to support claims and negotiation;
- implementing cost control and change order approval procedures, as well as cost estimating and schedule review programmes to support claims avoidance and negotiations; and
- developing an issue resolution process, including regularly scheduled weekly or bi-weekly identification and issue resolution/ negotiation sessions, in order to
prevent an accumulation of unresolved issues and to identify problems that experience suggests might become late, surprise issues.

Besides the improvement of the project and claims management, another important approach is to improve the claims negotiation elements, such as:

- improving negotiation planning;
- adopting a proper negotiation style;
- organising a suitable negotiation team with proper empowerment;
- establishing new negotiation procedures such as step by step negotiation;
- agreeing on common issues in advance, and establishing and utilising precedents to reduce the need for repetition in negotiation;
- establishing mutually agreed upon fairness; and
- educating the industry participants.

Most of the current claims negotiation researchers (Smith, 1992; Filed, et al., 1993; Zack, 1994) focus on one or more of the approaches discussed above. However, the improvement of these approaches, especially those related to general project management and claims management involves complex issues and is hard to achieve in practice. On the other hand, the development of information technology provides the industry with a simple and effective tool for the improvement of many problems, which are difficult to resolve through traditional approaches. This research intends to overcome the inefficiency problem of the claims negotiation process through the adoption of new information technology.

Although the reasons may be different, low efficiency has been recognised as a common problem of various negotiations. The attempts to introduce information technology, such as: expert systems and decision supporting systems, to facilitate negotiation have been tried in different areas (Anson and Jelassi, 1990; Samuelson, 1995; Shell, 1995; Saunders and Lewicki, 1995). However, the functions of these systems are limited because they cannot match some of the key characteristics of
negotiation, such as communications between negotiators. Instead, these isolated systems can only provide advice to human negotiators.

A technology that has the potential to improve the efficiency and effectiveness of claims negotiations involves the use of a multi-agent system (MAS) to facilitate negotiations between project participants. Multi-agent systems are network systems composed of individual agents which can negotiate for their own benefits. These individual agents have the following capabilities and characterises: autonomy, facilitating and filtering information, communication, learning, and facilitating collaboration. This makes multi-agent systems ideal for supporting claims negotiation (for details, see Chapter 3). Unlike any existing negotiation supporting system, agents in a MAS, on behalf of the different project participants, can directly negotiate with each other regarding a claim item, and reach a desirable solution within a specified time frame. This constitutes a promising approach to solving complicated negotiation problems in a natural way (Ferber, 1998). Based on a pre-designed negotiation protocol, each agent negotiates with others by adopting different negotiation strategies, and finally reaches a desirable solution within a specified time frame.

This research aims to develop a multi-agent system for construction claims negotiation (MASCOT) so that the low efficiency problem of claims negotiation can be relieved. Importantly, the distributed and collaborative characteristics of multi-agent systems also provides an opportunity to resolve some other problems caused by the fragmentation of the industry, which are difficult to resolve with other information technologies. The resolution of these problems will, in turn, improve both the effectiveness and efficiency of claims negotiation. Furthermore, the system also provides an environment where other alternative dispute resolution approaches, such as the use of a mediator can be involved early in the claims negotiation process.

2.6 SUMMARY

This chapter has investigated the development of claims management and claims negotiation through a review of technical, professional and academic literature, industry survey (interviews) and analysis of this information. The key features and
current status of claims management have been addressed. The applications of
information technology in claims management were explored. The major problems
and their causes of construction claims management such as inadequate information
and documentation, lack of effective claims management tool, and inefficient claims
negotiation have also been addressed.

Construction claims negotiation was particularly highlighted. A conceptual model of
claims negotiation was established where the backgrounds, planning and process of
claims negotiation were analysed. The problematic issues and difficulties associated
with the activities of claims negotiations were addressed and possible solutions for the
problem have been suggested. Such a thorough analysis on various aspects of
construction claims negotiation set a firm foundation for the further development of
the reasoning model of construction claims negotiation which is the basis for the
MASCOT system. Chapter 5 will make a further analysis on the nature and key
characteristics of claims negotiation.

The next chapter will provide a detailed analysis on multi-agent systems.
CHAPTER 3
MULTI-AGENT SYSTEMS

3.1 INTRODUCTION

The concept of distributed artificial intelligence (DAI) originates from the real world where many cases are inherently distributed in space, function, knowledge, expertise or information (Durfee et al., 1989). The notion of DAI provides a natural metaphor to match such distribution. It represents a new way of analysing, designing, and implementing complex software systems. The key advantage of DAI is its responsibility for enacting various components of the business process which is delegated to a number of autonomous agents. These agents act collectively as a society and collaborate to achieve their own individual goals as well as the common goal of the society to which they belong (Ugwu et al., 1999). Agents are inherently modular and can be constructed locally for each resource, provided they satisfy some high level protocol of interaction.

The agent-based view in DAI offers a powerful repertoire of tools, techniques, and metaphors that have the potential to considerably improve the way in which people conceptualise and implement their systems (Jennings et al., 1999). DAI applications such as: information access, information filtering, electronic commerce, workflow management, intelligent management, and various negotiations are becoming ever more prevalent. The common point in these different types of system is the key abstraction used – (distributed) agents. The significance of agents is that they provide a natural means for performing the above tasks over a distributed environment.

Multi-agent systems emanate from the traditional field of DAI. In MAS, there is no central controlling party. Agents often co-operate to achieve their own individual goals, rather than to solve a common problem. MAS are suitable for domains that involve interactions between different people or organisations with different (possibly conflicting) goals and proprietary information. With the lack of centralised control, agents in MAS have to solve the problem of the relationship between each agent’s
behaviour and goals, and those of the global system or MAS community. Negotiation, thus, often plays a central role in agent co-operation. Moreover, agents have reasoning abilities to infer the other agents’ key features and changes in the environment.

This chapter explores the common threads that together make up the agent and multi-agent systems. The purpose is to provide an in-depth analysis of multi-agent systems, and indicate the key issues in the field. Particular focus is on the nature of autonomous agents and multi-agent systems, negotiations, and learning issues in multi-agent systems.

3.2 INTELLIGENT AGENTS

Intelligent agents are the basic cells of multi-agent systems. To understand MAS, the starting point is to define and understand the intelligent agent.

3.2.1 Definition

There is little agreement on the definition of the terms 'agent' and 'intelligent agent'. They should be clearly more than just a program, but where the boundaries lie is not clear at all. This is a manifestation of the general problem in AI of defining ‘intelligence’ that has led to much discussion. The result is that there are as many agent definitions as there are researchers, leading to the term being substantially overused. Some examples are:

- Brustoloni (1991): "Autonomous agents are systems capable of autonomous, purposeful action in the real world."
- KidSim (1994): "Let us define an agent as a persistent software entity dedicated to a specific purpose. 'Persistent' distinguishes agents from subroutines; agents have their own ideas about how to accomplish tasks, their own agendas. 'Special purpose' distinguishes them from entire multifunction applications; agents are typically much smaller."
Maes (1995): "Autonomous agents are computational systems that inhabit some complex dynamic environment, sense and act autonomously in this environment, and by doing so realise a set of goals or tasks for which they are designed."

Coen (1995): “Software agents are programs that engage in dialogs and negotiation and coordinate transfer of information.”

Hayes-Roth (1995): “Intelligent agents continuously perform three functions: perception of dynamic conditions in the environment; action to affect conditions in the environment; and reasoning to interpret perceptions; solve problems, draw inferences, and determine actions.”

IBM white paper (1994) “Intelligent agents are software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user’s goals or desires.”

Although different researchers emphasise different aspects of agency, their definitions of agents suffer from one or more of these three problems (Ugwu et al., 1999):

- Too broad a definition that will include things such as Unix demons;
- Too narrow a definition that prescribes a particular AI technique; or
- Forming a definition in terms of equally vague terms.

Despite the different definitions, there are several broad qualities that have some measure of general agreement. Wooldridge and Jennings (1995) define an agent as “a computer system, situated in some environment that is capable of flexible autonomous action in order to meet its design objective”. There are thus four key concepts in the definition:

- **Autonomy**: agents should operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state.
- **Social ability**: agents need to be able to interact with other agents (and possibly humans) via some kind of agent-communication language.
- **Reactivity**: agents should be able to perceive their environment and respond in a timely fashion to changes that occur in it. This environment may be the physical
world, a user via a graphical user interface, a collection of other agents, the
Internet, or perhaps all of these combined.

- **Pro-activeness**: agents should not simply act in response to their environment,
  they should be able to exhibit goal-directed behaviour by taking the initiative.

From this and other definitions, the key feature would appear to be 'autonomy'.
However, this is usually loosely defined; containing words like 'control over their own
actions' or 'formulate their own goals'. An agent with freewill is a high aspiration
indeed. For learning agents that deal with different situations in different ways as they
learn, something that appeared like autonomous behaviour would be possible.
However, few definitions actually include learning. Non-learning agents ultimately
only follow the same set of instructions and/or rules at all times during their existence,
hence for these agents, autonomy must take on a somewhat weaker meaning. A more
meaningful definition may be to say an agent is autonomous if it operates without the
need for the direct intervention of humans.

Nwana (1996) takes Wooldridge and Jenning's definition and reduces it to three
behavioural attributes, any two of which must be possessed by a software agent.
These are:

- **Autonomy**: this refers to the principle that agents can operate on their own
  without the need for human guidance. Agents have individual internal states and
  goals, and act in such a manner as to meet their goals. A key element of their
  autonomy is their pro-activeness, i.e. the ability to 'take the initiative' rather than
  acting simply in response to their environment.

- **Co-operation**: co-operation with other agents is paramount, and is the reason for
  having multiple agents in the first place. In order to co-operate, agents need to
  possess a social ability, i.e. the ability to interact with other agents and possibly
  humans via some communication language.

- **Learning**: for agent systems to be truly 'smart', they would have to learn as they
  react and/or interact with their external environment. A key attribute of any
  intelligent being is its ability to learn. The learning may also take the form of
  increased performance over time.
Nwana’s requirements for agenthood may be neatly shown as a Venn diagram in Figure 3.1.

![Venn Diagram](image)

**Figure 3.1: Nwana’s requirements for agenthood**

The inclusion of learning is at least an aspiration, it recognises a core quality of the intelligent behaviour. The diagram neatly provides a framework into which all software agents in this thesis can be currently defined. A number of other attributes for agenthood have been cited. Some of these need to be considered for any classification of agents; others are better under the category of generally desirable qualities. In the next section Nwana’s definition will be extended to include some of these attributes. Figure 3.2 shows an example of an agent architecture (Sen, 1997).

![Agent Architecture Diagram](image)

**Figure 3.2 An agent architecture** (shaded modules represent components particular to agents in a MAS)
3.2.2 Agent Development

Agents can be understood as an incremental extension of previous software technologies (Table 3.1). In the beginning was the program, a monolithic deck of machine instructions and data tied together with tangled ‘goto’ statements that took over the complete resources of the computer when the user fed it into the card reader. The structured programming movement modularised program code through constructs such as subroutines and structured loops. These constructs localised the definition of how the program would function, but relied on an external definition of the data on which the code would operate and external innovation. The next major development, objects, gave software modules local state as well as local code, but innovation was still determined externally, by sending a message. Agents add two things to (passive) objects: a local thread of control, and local initiative (usually expressed as local goals). Together, these enable the agents to monitor and respond to their environment autonomously (Parunak et al. 1997).

Two requirements make agents an attractive technology for a modern design environment.

- First, agents are intrinsically distributed. While their local threads can be supported on a single processor, it is also natural to distribute them across a network, supporting the distribution requirements.
- Second, the modularity of agents makes it natural to encapsulate humans as peer agents to computer processes using common language and protocols to integrate people and machines. In nature, this integration requires people to reduce the bandwidth of their communication to a level that computerised agents can handle.

| Table 3.1 Agents in historical perspective (Parunak et al. 1997) |
|-----------------------------|-----------------------------|-----------------------------|-----------------------------|
| How does a unit behave?     | Monolithic program          | Structured programming      | Object-oriented programming (OO) | Agent-oriented programming |
|                             | External                    | Local                       | local                       | local                      |
| What does a unit do when it runs? | External                    | External                    | local                       | local                      |
| When does a unit run?      | External                    | External (called)           | External (message)          | Local (thread; goals)      |
Many researchers appear to confuse the following: agents and objects; distributed computing and agent-based computing; object-oriented systems (OO), expert systems and agent-based systems. OO computing, and distributed computing do not in themselves offer solutions to agent-based problems because distributed computing modules are usually passive and dumb. Also, their communications are usually low level while agent-based systems require high-level messages (Newell, 1982).

OO techniques are good in general, but are rather low-level for intelligent applications. They can be used, for instance, to implement knowledge representations, but they do not themselves provide a knowledge representation. OO development methodologies can, however, be seen as a low-level underpinning for a multi-agent methodology. The same might be said of distributed computing methodologies and indeed, many agent-based systems are built on top of distributed platforms. However, it can be argued that if OO approaches are still relatively new, agent-based systems are even newer and less generally accepted. Again, the knowledge level is wrong. For instance, communications protocols do not operate at the high level of Speech Acts as one might wish for an agent-based system. More importantly, agent applications require a co-operative knowledge level (Jennings and Campos, 1997), while expert systems typically operate at the symbolic and knowledge level (Newell, 1982).

The benefit of an agent-based system would be a reduction of the semantic gap between analysis on the one hand, and design and implementation on the other, leading to a reduction in the time to design and implement, with the usual trade-off between better expandability and losses in execution efficiency and design specificity. Current methodologies emphasise top-down design, but agent-based systems adopt a different approach: top-down within the agent, and bottom-up in the agent community (Wooldridge, 1997). In summary, agent-based systems research can be regarded as developing a way of looking at problems rather than a technology. Hence, agent-based systems can, and do, use OO programs, expert systems, and distributed computing technologies to implement applications and toolkits that embody this approach.
3.2.3 Agent Taxonomy

The use of a taxonomy is an important approach in understanding and designing agents. There have been many attempts to produce taxonomies or classifications of agents. None of these seem to be complete and most of them become dated quickly. For example, Brustoloni's (1991) taxonomy of software agents begins with a three-way classification into regulation agents, planning agents, or adaptive agents. A regulation agent reacts to each sensory input as it comes in, and always knows what to do. It neither plans nor learns, and so on. This yields a two-layer taxonomy.

There are many other possible classification schemes. Agents might be classified according to the tasks they perform such as: information gathering agents or email filtering agents. Agents may also be classified by the range and effectiveness of their actions, or by the degree of sophistication of their internal state such as: goal driven, non-temporal agents to those with a full Beliefs, Desires and Intentions (BDI) reasoning capability. Another possible taxonomy might involve the environment in which the agent finds itself, for example software agents as opposed to artificial life agents.

Most of these taxonomies seem cumbersome and unsuited to classifying many agent types. Any taxonomy has to divide agent-space from the top and there seems to be no natural top level divisions. Secondly, agents are often an ad hoc collection of techniques, and frequently fall into two categories of a taxonomy. Given the difficulties of a formal taxonomy, a more pragmatic approach may be to list the qualities that designers may wish their agents to have, each as a separate dimension. An agent will then be represented by a point in agent-space. Table 3.2 extends Nwana's model (1996) with other desirable or useful agent qualities. Nwana's qualities are represented as core qualities. There are four agent quality types:

- **Core**: at least two need to be present for agenthood.
- **Motivational**: this deals with the basis on which an agent interacts with other agents, the environment, etc., whether it is acting for itself, a group or a wider community. This is really a continuum and a point must be chosen along this.
<table>
<thead>
<tr>
<th>Quality Type</th>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core</td>
<td>autonomous</td>
<td>exercises control over its own actions</td>
</tr>
<tr>
<td></td>
<td>co-operative</td>
<td>communicates with other agents, perhaps including people</td>
</tr>
<tr>
<td></td>
<td>learning</td>
<td>changes its behaviour based on its previous experience</td>
</tr>
<tr>
<td>Activity</td>
<td>goal-orientated</td>
<td>is the classical symbolic AI approach and is based on the Physical Symbol System Hypothesis, where the external world can be modelled symbolically and when processes are run on this representation the agent has its own purpose, and does not simply act in response to the environment (Newell and Simon, 1976).</td>
</tr>
<tr>
<td></td>
<td>reactive</td>
<td>responds in a timely fashion to changes in the environment. In extreme, it represents a rejection of all symbolic world models because one method, i.e. formal logic, has been found wanting. Such simple systems usually involve learning to respond to certain inputs in certain ways.</td>
</tr>
<tr>
<td></td>
<td>hybrid</td>
<td>allows agents to be able to respond rapidly in a reactive and possibly learned way, while maintaining a small and consequently manageable symbolic representation of the world.</td>
</tr>
<tr>
<td>Motivation</td>
<td>altruistic</td>
<td>does not have private goals, and acts in the interest of fellow agents or a group. These agents are really only possible in closed systems.</td>
</tr>
<tr>
<td></td>
<td>self-interested</td>
<td>acts in own self-interest. Many of them are concerned with task allocation and resource sharing.</td>
</tr>
<tr>
<td></td>
<td>veracious</td>
<td>will not knowingly communicate false information.</td>
</tr>
<tr>
<td>Other</td>
<td>mobile</td>
<td>able to transport itself from one machine to another</td>
</tr>
<tr>
<td></td>
<td>personalisable</td>
<td>easily retrained to perform a task the way a particular user wants it done (Foner, 1993).</td>
</tr>
<tr>
<td></td>
<td>graceful</td>
<td>if failure occurs most of a task can still be accomplished, instead of failing to accomplish any of the task (Foner, 1993).</td>
</tr>
<tr>
<td></td>
<td>degradation</td>
<td></td>
</tr>
</tbody>
</table>
• **Activity**: this defines the level of passivity of an agent. Does it actively seek to realise its own goals, or does it wait for the environment to change around it?

• **Other**: a collection of qualities sometimes seen as necessary for particular agent applications.

Classes of agents can then be defined by some combination of the core qualities, motivation, activity and the extra properties. No agent will have all these qualities. For example, a poker-playing agent would not want to be veracious. Equally a proactive, autonomous agent within an open system (e.g. the Internet) would not necessarily want to be altruistic. Some of these qualities are discussed in more detail below.

### 3.2.3.1 Activity

Muller (1998) classifies agents according to the influential threads of agent research, i.e. reactive agents, deliberative agents, and interacting agents. Each of these threads focuses on one important property of an agent. Reactive agents focus on reactive and real-time behaviour. Deliberative agents focus on the ability to act in a goal-directed manner. Interacting agents focus on the ability of co-operative social behaviour.

• **Reactive Agents**

Reactive agents are built according to the behaviour-based paradigm, have no (or at most a very simple) internal representation of the world, and provide a tight coupling of perception and action. Such simple systems usually involve learning to respond to certain inputs in certain ways. Such agents can learn by observation of the user, but cannot formulate plans to achieve distant goals. There are two important points that the reactive approach has introduced to AI generally and agent research in particular (Wooldridge and Jennings, 1995):

a) Intelligence is not exclusively to be found in large, centralised singular systems, but can be an emergent property of the interaction of many unintelligent units.
b) Most everyday activity is really routine, once learned it can be accomplished in a routine way with little variation.

Valuable though these insights are, the requirements placed on many agents means there has to be some symbolic representation. This is because they usually have to interact with humans who will ask explicit questions such as ‘why?’ and ‘how?’ They are also often expected to interact with a great variety of other agents with vastly different requirements.

• Deliberative Agents

Deliberative agents are agents in the symbolic artificial intelligence tradition that have a symbolic representation of the world in terms of categories such as beliefs, goals or intentions, and that possess logical inference mechanisms to make decisions based on their world model. In essence this states that the external world can be modeled symbolically and when processes are run on this representation (essentially theorem proving) it is capable of general intelligent action. Some of the key problems are:

a) *Speed of execution:* A logic-based system uses formal rules of inference to deduce if some action or event is a logical consequence of its current knowledge/beliefs, etc. The theorem proving search process is an NP-complete problem (non-polynomial – its time complexity is exponential). The effect of this, even in moderately dynamic and unpredictable environments, is that the world changes faster than a plan-based agent can reason about actions to take to achieve its goals and execute those actions.

b) *Common-sense reasoning:* This is perhaps the second most important practical shortcoming of plan-based agents. In most applications, common-sense reasoning involving notions such as time, space, and causality is handled in an ad hoc, application-specific manner. Indeed, the issue of common-sense reasoning has defied formal treatment in AI, and presents perhaps the major stumbling block to the development of reasonably smart agent systems.

c) *Reasoning about other agents’ desires and intentions:* A reasoning agent in a cooperative setting, in addition to the communications overheads to ensure cooperation, agents have to reason about the goals, plans and beliefs of other agents,
in order, for example, to minimise duplication of effort. This is itself an NP-complete problem and, for practical purposes, it dramatically reduces the size of the manageable knowledge-base of each agent.

- **Interacting Agents**

Until recently, interacting agents were not classified separately. Interacting agents are able to coordinate their activities with those of other agents through communication and, in particular, negotiation. Interacting agents have been mainly investigated in distributed AI; they may have explicit representations of other agents, and may be able to reason about them. So far, the focus has been on the co-ordination process itself and on mechanisms for co-operation among autonomous agents rather than on the structure of these agents. A detailed discussion about interacting agents is contained in Section 3.4.

Besides the reactive agents and deliberative agents, Ferguson (1992) classifies hybrid agents. Hybrid systems allow the agent to be able to respond rapidly in a reactive and possibly learned way, while maintaining a small and consequently manageable symbolic representation of the world. Its reactive quality means the nuts and bolts of how it interacts with the world do not need to be represented symbolically. In this way the symbolic system can concentrate on planning, and dealing with unexpected situations. Thus, it is still capable of functioning reactively while waiting for symbolic decisions to be made. The Touring Machines hybrid (Ferguson, 1992) is a good example of this approach.

### 3.2.3.2 Motivation

Agents can be classified as self-interested and altruistic according to their motivation for action:

- **Altruistic Agents**

Some domains are inherently suited to the use of altruistic agents. For example a single factory scheduling problem where each work-cell is represented by an agent. If
the cells do not have private goals the agents will need to act in the interest of the company, i.e. altruistically. There are many multi-agent collaboration projects that use altruistic agents within the literature, some illustrative examples follow. Negotiation in the examples is generally within a small group of 'friend' agents. For example,

**Pleiades:** The complete system is designed for a limited environment (called the InfoSphere) and is a set of Internet-based heterogeneous information resources by Sycara and Zeng (1994). *Pleiades* has been used to create the Visitor Hosting System where the agents co-operate in order to manage a visitor's schedule. Agents arrange appointments and meetings with other agents and accordingly formulate plans for the visitor. Agent skills include the following: knowledge of how to gather information, knowledge of other agents it must co-ordinate with, and strategies for, conflict resolution. The architecture has no central planner and hence agents must all engage in co-ordination by communicating to others their constraints, expectations and other relevant information.

**ADEPT:** This project uses collaborating agents to help in business decision making where the information needed is often spread through several companies in different databases (Jennings et al., 1996). This system attempts to provide access to business processes and information on request. Due to the distributed nature of business information collaborative agent technology provides a good basis upon which to build a system of this kind. The requirements in providing quotations cover such things as obtaining credit references, designing the system and costing it.

- **Self-Interested Agents**

There are many examples of self-interested agents. Many of them concern task allocation and resource sharing where the agents are self-interested, some examples follow:

a) Electronic marketplaces: Agents representing different enterprises, buying and selling (Rosenschein and Zlotkin, 1994).

b) Information retrieval: Information servers can form coalitions for answering queries (Tsvetovatyy and Gini, 1996; Fischer et al., 1996).
c) Air Traffic Control: Aeroplanes belonging to different airlines need to share the limited resources at airport, the control mechanism needs to give priority to planes with less fuel on board (Sandholm and Lesser, 1997a).

d) Distribution problems: Package delivery companies may co-operate to reduce expenses (Sen and Durfee, 1997).

In these examples, the agents are self-interested and try to maximise their own benefits. They typically involve less than a dozen agents. There has been less work done with self-interested agents in open systems. One of the main reasons is that it is much harder to negotiate with other agents who might not be telling the truth. Trust during negotiation is one of the key issues. The majority of self-interested agents are based on Game Theory or Economic Theory. A further discussion of Game Theory and Economic Theory is contained in Section 5.1.

3.2.4 Agent Co-ordination

Although there are single agent systems like information retrieval agent, most agent-based systems contain more than one agent. Therefore, co-ordination is central to agent-based systems to ensure a community of individual agents acting in a coherent, harmonious and expected way. The approach to agent co-ordination reflects how people view the real world problems and model them in agent-based systems. Agent co-ordination has been studied by researchers in diverse disciplines such as: organisation theory, economic theory, social psychology, anthropology, and sociology. Several typical ways of structuring and co-ordinating agents have been developed. Here are some examples:

- **Organisational Structuring**: This is one of the most common and also the simplest interaction mechanisms. It usually exploits a hierarchical structure, implemented as client/server or master/slave, where the master could gather information from the agents of the group, create plans, and assign tasks to individual agents in order to ensure global coherence. A typical use might be for resource allocation such as Werkman (1990). A more practical application may be an assembly line. Here all agents are set up to interact only with several other agents, all sharing the same explicit end goal (i.e. the manufacture of a product).
However, these types of system allow few of the benefits of DAI as it assumes there is one agent with a global view of the full task. In real and complex situations agents are likely to possess knowledge that the central agent does not know of, and feeding this back to the central agent is difficult. Additionally it is very difficult for peer agents with different goals to resolve their difficulties as all must go before a central arbiter with imperfect knowledge.

- **Contracting**: The Contract Net protocol (Smith, 1980) is one of the most commonly used protocols. It is different from a master/slave system in that a manager agent will break a problem into the component problems and then announce each task. Contractors then table bids to the manager. The manager then reviews the bids and awards the contract. Contract nets are best used when the problem can be broken down via a well-defined hierarchical nature into a set of tasks. This means that the main planning has to be done centrally.

- **Multi-Agent Planning**: This usually involves a central arbiter who will review all potential plans of individual agents. This agent then checks for conflict and rejects or revises as appropriate. An example of this is multi-agent planning for air traffic control (Cammarata et al., 1983). It is appropriate to have a central agent with ultimate responsibility in such a safety-critical area. The system works because within the problem space there are many possible solutions that at most cause only minor inconvenience to other aircraft, hence conflict is less common. In other domains where there is more chance of a conflict of interests between agents, direct negotiation is more efficient.

- **Peer To Peer Negotiation**: All the preceding organisational models can and do involve negotiation but it generally plays a minor part, as it will be negotiation between agents of differing rank. Peer to peer negotiation effectively means there is no structure and agents must communicate directly with other agents to achieve its (or the group's) goals. Thus, it is most commonly adopted in MAS systems.

A detailed analysis of these forms of organisation is discussed in the following sections.
3.3 MULTI-AGENT SYSTEMS

Distributed artificial intelligence is designed and implemented as several interacting agents. The goal of DAI is clear enough and has been proven in many prototypes: creating a system that interconnects separately developed agents, thus enabling the ensemble to function beyond the capabilities of any singular agent in the set-up (Nwana and Ndumu, 1999). DAI systems have received considerable attention for two main reasons (Russell and Norvig, 1995).

- First, they have useful properties such as parallelism, robustness, and scalability. Therefore, they are applicable in many domains which cannot be handled by centralised AI systems. In particular, they are well suited for domains which require resolution of interest and goal conflicts, integration of multiple knowledge sources and resources, time-bounded processing of very large data sets, or on-line interpretation of data arising from different geographical locations.

- Second, they are in accordance with the insight gained in disciplines such as AI, psychology, and sociology that intelligence is tightly and inevitably coupled with interaction.

In other words, DAI is ideally suited to representing problems that have multiple problem solving methods, multiple perspectives and/or multiple problem solving entities. Such systems have the traditional advantages of distributed and concurrent problem solving, and also have the additional advantage of sophisticated patterns of interactions. Examples of common types of interactions include: co-operation (working together towards a common aim); co-ordination (organising problem solving activity so that harmful interactions are avoided or beneficial interactions are exploited); and consensus (coming to an agreement which is acceptable to all the parties involved) (Jennings et al., 1999). It is the flexibility and high-level nature of the interactions which distinguishes DAI from other forms of software and which provides the underlying power of the paradigm.
Traditionally, research into systems composed of multiple agents was carried out under the banner of DAI, and has historically been divided into two main camps: Distributed Problem Solving (DPS) and Multi-Agent Systems. More recently, the term 'multi-agent systems' has come to have a more general meaning, and is now used to refer to all types of systems composed of multiple autonomous components.

3.3.1 Distributed Problem Solving

DPS considers how the task of solving a particular problem can be divided among a number of modules (or nodes) that co-operate in dividing and sharing knowledge about the problem and about its evolving solutions. In a pure DPS system, all interaction strategies are incorporated as an integral part of the system. Agents' interactions are guided by co-operation strategies meant to improve their collective performance. Conflict among the agents in these environments may arise while each tries to achieve its own sub-task, but their overall task is the same. For example,

- **Air traffic control**: Cammarata et al. (1983) develop co-operation strategies for resolving conflicts between the plans of a group of agents. They apply these strategies to an air-traffic control domain, in which the aim is to enable each agent to construct a flight plan that will maintain a safe distance with each aircraft in its vicinity and satisfy additional constraints. Agents involved in a potentially conflicting situation choose one of the agents involved in the conflict to resolve it. The chosen agent acts as a centralised planner to develop a multi-agent plan that specifies the conflict-free flight paths that the agents will follow. The decision of which agent will do the planning is based on different criteria, for example, 'most-informationed' agent, or 'most-constrained' agents.

- **The distributed vehicle monitoring task**: In this domain, a set of agents is distributed geographically, with each being capable of sensing some portion of an overall area to be monitored. As vehicles move through its sensed area, each agent detects characterised sounds from those vehicles at discrete time intervals. By analysing the combination of sounds heard from a particular location at a specific time, an agent can develop interpretations of what vehicle might have created...
these sounds. By analysing temperate sequences of vehicle interpretations, and using knowledge about mobility constraints of different vehicles, the agent can generate tentative maps of vehicle movements in its area. By communicating tentative maps to one another, agents can obtain increased reliability and avoid redundant tracking in overlapping regions (Durfee, 1988).

- **Industrial process control**: ARCHON, a software platform for building DAI, and an associated methodology for building applications, has been developed to facilitate industrial process control. The system addresses two major problems. One is concerned with providing the necessary control and level of integration to help the sub-components of an industrial process to work together. Another is concerned with decomposing the overall application goal(s) and with distributing the constituent tasks throughout the community. The system has been applied in several process control applications such as: electricity transportation management, and particle accelerator control. It is one of the world's earliest field-tested DAI systems (Cockburn and Jennings, 1996).


3.3.2 Multi-Agent Systems (MAS)

Research in MAS is concerned with the behaviour of a collection of autonomous agents aiming to solve a given problem. There is no global control, no globally consistent knowledge, and no globally shared goals or global success criteria among these agents (Hewitt, 1985). MAS offer a way to relax the constraints of centralised, planned, sequential control, although not every MAS takes full advantage of this potential. They offer production systems that are decentralised rather than centralised, emergent rather than planned, and concurrent rather than sequential (Parunak, 1996).

Jennings et al. (1998) summarise the characteristics of MAS as:

- Each agent is individually motivated and attempts to maximise its own utility;
Each agent has incomplete information, or capabilities for solving the problem, thus each agent has a limited viewpoint;

There is no global system control;

Data is decentralised; and

Computation is asynchronous.

There is an increasing interest in multi-agent system research because of its ability to provide robustness and efficiency; the ability to allow inter-operation of existing legacy systems; and the ability to solve problems in which data, expertise, or control is distributed. Multi-agent systems have been applied in many cases where the need for interaction and negotiation exists, such as WWW searches, e-commerce, supply chain management, design/project management, knowledge management, project coordination, computer networks, operating systems, multi-enterprise manufacturing, or multi-robot systems. Here are some examples:

- **Adjusting agent autonomy in supply chain management (SCM):** A supply chain is a network of suppliers, factories, warehouses, distribution centres and retailers. SCM manages the co-operation of these system components, which correlate with each other through chain activities to implement system functionality. Software agents (Lin et al., 1998; Chen et al., 1999) are introduced as an entity with goals, action and domain knowledge situated in the environment. Each functionality of the chain is implemented by one kind of autonomous, intelligent, proactive and adaptive agent. These agents co-operate in a dynamically changing and open environment. A mixed negotiation process is developed for the specific problem where SCM's functional agents contain self-adaptive rules for their reasoning procedure, meanwhile human involvement is also considered.

- **Agent-based international crisis negotiation:** In this case, a strategic negotiation model of alternative offers is developed to facilitate the negotiations among the three parties of a hostage crisis. In this model, agents are self-motivated, rational, and autonomous, each with its own utility function. Both parties can opt out, and while one loses over time, the other gains (up to a point). Specific issues are
conflicting objectives and utility functions of parties, and the impact of each item on bargaining behaviour in a crisis. The study provides strategies for a wide range of situations, which satisfy the criteria: symmetrical distribution, simplicity, instantaneous, efficiency and stability (Kraus, 1993).

- Agent based project management (IPM): Integrated project management (IPM) means that design and project planning are interleaved with plan execution, allowing both the design and plan to be changed as necessary. This requires the right change to be propagated through plan and design. This study develops a Redux model to facilitate the IPM, where agents communicate their goals and decisions to the Redux server, uses a set of rules to maintain the consistency of goals and decisions, and propagate the effects of design changes in a collaborative design environment. The main problems addressed in the Redux server include conflict resolution when design constraints are violated, and communicating any design changes to appropriate team members whose decisions will be affected by such changes. Such timely communication will reduce the impact of making design decisions on the basis of obsolete data/information, and this level of message propagation involves some reasoning by the agent that generates such changes (Petrie et al., 1998).


With the wide range of applications, some key aspects of MAS have gained considerable attention from researchers such as: interactions between self-interested agents, negotiation protocols, and learning approaches of agents. For example, self-interested agents interact in a shared environment which represents many real world problems. Such domains include both systems where agents are adversarial to each other (e.g. bargaining parties) as well as domains where agents are indifferent to each other. In the former case, research concentrates on issues like modelling the knowledge and behavioural strategies of opponents, learning to exploit an opponent's
weakness, and developing interaction rules by which agents can arrive at equilibrium configurations (Sen, 1997). In the latter case, the key research problems are designing social laws, conventions, and protocols by which each agent can achieve its own goal without significantly affecting the chances of others achieving their goals, then the whole society exhibits desirable behaviour - in other words, locally good behaviour implies globally good behaviour (Jennings et al., 1999).

The goal is to design a mechanism for self-interested agents such that if agents follow this, the overall system behaviour will be acceptable. For example, economic-based approaches, and market mechanisms in particular, are becoming increasingly attractive to MAS researchers both because of the ready availability of underlying models, and their potential applicability in Internet-based commerce. In such approaches, agents are often assumed to be self-interested utility maximisers. The areas where economic-based approaches have been applied to MAS research to date are resource allocation; task allocation and negotiation. The next section will focus on the general negotiation mechanisms in multi-agent systems.

3.4 NEGOTIATION IN MULTI-AGENT SYSTEMS

In systems composed of multiple autonomous agents, negotiation is a key form of interaction that enables groups of agents to arrive at a mutual agreement regarding belief, goal, or plan. Particularly because the agents are autonomous and, in many cases, are self-motivated, agents must influence others to convince them to act in certain ways (Beer et al., 1999). Negotiation is used more specifically for conflict resolution and avoidance (Adler et al., 1989; Cammarata et al., 1983; Sycara, 1988; Klein, 1991), task allocation (Davis and Smith, 1983; Durfee and Montgomery, 1990; Zlotkin and Rosenschein, 1994), and resource allocation (Adler et al., 1989; Sathi and Fox, 1989; Conry et al., 1991), and hence for the coherence of the agent society. The potential benefits of agent negotiation include saving time and money, efficiency for computationally intense negotiations searching for optimal results, and the ability to incorporate multiple negotiation strategies for changing environments.
3.4.1 Overview

Although negotiation is highly important for the modelling of multi-agent systems, there is no clear and common definition of what negotiation is in the agent world, and no formal theory of agent negotiations (Muller, 1996). Jennings et al. (1999) describe negotiations between agents as: "The main characteristics of negotiation in terms of agent applications can be stated as: the presence of some form of conflicts that need to be resolved in a decentralised manner, by (self-interested) agents, under conditions of bounded rationality, and incomplete information. Furthermore, the agents communicate and actively exchange proposals and counter-proposals. Conflicts between agents may be about the limited available resources, conflict beliefs between agents, or disagreements on issues, such as price. In the first case, the negotiation becomes an optimisation problem; in the second case, at least one of the agents has to change its beliefs; and in the third case, the negotiation is a bargaining situation. It is often difficult to see what exactly one wants to achieve when resolving the situation."

However, many other researchers like Smith (1980) and Shen (2001) consider negotiation as a more general approach to agent co-ordination. It is not necessary that there is a conflict between agents as a condition of negotiation. Negotiation could be simply for the purpose of a better solution to a problem. Perhaps, a basic definition is that of Bussmann and Muller (1992): "...negotiation is the communication process of a group of agents in order to reach a mutually accepted agreement on some matter."

This section will discuss agent negotiation in a broad sense (i.e. consider negotiation mechanisms for both DPS and MAS).

Despite the ambiguous term and the wide range of negotiations - from situations involving task and resource allocation, to situations involving agent to agent bargaining - the negotiations are intended to improve the global state of affairs or to achieve individual agents' objectives such as: minimising the time to find a solution; minimising the total resource usage in doing so; maximising the quantity of the result; trying to achieve Pareto optimality (i.e. the outcome maximises the product of the agents' utilities), or trying to reach a Nash equilibrium (Shen et al., 2001).
To negotiate with one another, each individual agent should be able to make proposals, counter-proposals, accept or reject proposals, and generate arguments in support of their adopted stance on the negotiating subject. Preist (1999) extends this with some points relating specifically to competitive negotiation:

- First, all parties should be clear about the belief sets of each of the agents involved in the negotiation process. At the very least a shared ontology of the domain being negotiated over, or a way of establishing one needs to exist.

- Then they must be able to make contact with potential negotiators and recognise the type of “game” in progress. Questions, such as the number of negotiators and the bargaining power of individuals, need to be considered and an appropriate strategy based on this information selected.

- During the negotiation, it requires both a means to evaluate the relative value to the agent of different offers made during negotiation and a means to estimate the relative value to other agents of offers the agent may potentially make.

- Finally, in some situations, the ability to exchange constraints on acceptable offers and to reason with these constraints may make negotiation more efficient.

The important issues in agent negotiation include the application domain, system architectures and infrastructures, interaction protocols, and the architecture and strategies adopted by individual agents. Muller (1996) summarises both the agent and group negotiation issues into three major categories (Table 3.3):

Table 3.3 Categories for agent negotiation (Muller, 1996)

<table>
<thead>
<tr>
<th>Negotiation language category</th>
<th>Negotiation decision category</th>
<th>Negotiation process category</th>
</tr>
</thead>
<tbody>
<tr>
<td>protocols</td>
<td>utility</td>
<td>procedure</td>
</tr>
<tr>
<td>primitives</td>
<td>matching</td>
<td>behaviour</td>
</tr>
<tr>
<td>semantics</td>
<td>preferences</td>
<td></td>
</tr>
<tr>
<td>object structure</td>
<td>strategies</td>
<td></td>
</tr>
</tbody>
</table>

- Negotiation language category: research is concerned with the communication primitives for negotiation, their semantics and their usage in terms of a negotiation
protocol. This category also comprises the investigation of the structure of the negotiation topics.

- Negotiation decision category: algorithms to compare the negotiation topics and correlation functions for them are discussed. The definition of utility functions and the representation and structure of the agents’ preferences are fixed. Negotiation strategies also fall into this category.

- Negotiation process category: general models of the negotiation process are investigated and the global behaviour of the negotiation participants analysed.

Many other researchers (Jennings et al., 1999; Shen, 2001; Nwana and Ndumu, 1999) have different prospects about the major features of agent negotiation. For example, Shen (2001) outlines the three main features of agents negotiation as: language used by participating agents, protocol followed by agents as they negotiate, and the decision process each agent uses to determine its position, concessions and criteria for agreement. Nwana and Ndumu (1999) emphasise the importance of a common set of speech acts, a common service ontology, and a common set of prescriptive conversation policies (i.e. protocols for negotiating agents).

### 3.4.2 Agent Negotiation Mechanism

Many factors need to be considered prior to designing an agent negotiation mechanism (Ren et al. 2001b):

- First, autonomous agents, such as a computer system, work differently from human negotiators. Rule based (mechanical) negotiation theories are essential for the agent negotiation system, whilst some behavioural theoretical models, like learning approaches, are also applicable to the system.

- MAS negotiation mechanism should consider some highly desirable properties such as the ability to guarantee convergence, Pareto-optimality or equilibrium. The specific application scenario and objective should be reached. Meanwhile, factors like efficiency and symmetry should also be considered. For example, the designed negotiation mechanism should prevent agents from spending too much
time on negotiation, and therefore not keeping to their timetables for satisfying their goals.

- Finally, many other specific factors will also influence the design and implementation of a negotiation mechanism such as:
  a) information source (complete vs. incomplete);
  b) influence processes to change position (rationality in term of value, risk, or utility);
  c) level of shared goals among agents (all sharing the same goals vs. self-interested agents);
  d) number of agents (a few vs. very large number);
  e) type of agents (automated agents vs. system composed of people and automated agent; reactive vs. learning);
  f) communication and computation costs (the availability and costs of communication vs. the computation capability and costs); and
  g) agent organisations (peer to peer vs. superintendent to subordinate).

Different approaches have been adopted to develop agent negotiation systems and techniques, which can be categorised as either environment-centred or agent-centred. Environment-centred designers ask the question: How can the rules of the environment be designed so that the agents in it, regardless of their origin, capabilities, or intentions, will interact productively and fairly? Developers of agent-centred negotiation mechanisms focus on the question: What is the best strategy for an agent to follow, given an environment in which the agent must operate?

Given the ubiquity and importance in many different contexts, negotiation mechanism covers a broad range of phenomena, and encompasses multifarious approaches. Despite this variety, agents' negotiations are generally composed of two phases: a communication phase where information relevant to the negotiation is communicated to participating agents, and a bargaining phase where "deals" are made between individuals through relaxation of initial goals, mutual concessions, lies, or threats (Adler et al., 1989). Agent negotiation mechanisms mainly address these two aspects
which cover three main broad areas: negotiation objects, negotiation protocol and negotiation strategy.

- **Negotiation objects:** Negotiation objects are the range of issues over which agreement must be reached. Particularly, to what kinds of agreements can the agents come? The object may contain single issue, such as price, or multiple issues relating to price, timing, quality, etc. Also relevant are the allowable operations on these objects. In the simplest case, the structure and contents of the agreement are fixed, and negotiation amounts to accepting or rejecting the offer. The next level, however, offers flexibility to change the values of the issues in the negotiation object, through counter-proposal, changing the structure of the negotiation object, and so on. Finally, participant might be allowed to dynamically extend the structure of the negotiation objects (Jennings, 2000).

- **Negotiation protocols:** Given a set of possible deals, negotiation protocols set the rules by which the agents will come to a consensus. In a broad and general sense, negotiation protocols cover the permissible types of participants (e.g. the negotiators and relevant their parties), the negotiation states (e.g. accepting bids, negotiation closed), the events that cause state transitions (e.g. no more bidders, bid accepted), and the valid actions of the participants in a particular state (e.g. which can be sent by whom, to whom and at when) (Muller, 1996).

In a more specific sense, negotiation protocols set the stage for the negotiation process. They contain the basic rules for a negotiation process and the communication. Since negotiation involves exchanges of messages, protocols structure what are called conversations, define classes of dialogue. The simplest dialogues are found in contract-net approaches where they are limited to exchange involving offers, bids, and grants of contract. More complex dialogues are found in human types of negotiation, when trying to change other agents’ beliefs. Reed (1998) identifies five types of dialogue: persuasion, negotiation, inquiry, deliberation, and information seeking. For example, persuasion dialogue covers the case of conflicting beliefs. Negotiation dialogue differs procedurally from persuasion in an important respect, in that coherence between beliefs is not
demanded, and the relevant beliefs of the participants may well remain at odds after negotiation.

There are two approaches to dialogue. One is an unmediated approach which involves bilateral, or multilateral, communication between all of the agents. That mechanism may not scale well. In the mediated approach, agents submit messages to some institution implementing the mechanism. The process may be iterative, with the institution providing some feedback based on previous messages received. The process terminates under conditions prescribed by the mechanism rules (Wellman and Wurman, 1998).

- **Negotiation strategies:** Given a set of possible deals and a negotiation protocol, negotiation strategies provide the decision-making apparatus by which participants attempt to achieve their objectives. It determines which from possible alternative actions the agent will choose at each step. Each agent's strategy will strongly depend on the type of application it is involved in. For example, an agent may update its local states and perform its function based on certain predefined rules if the agent works in a collaborative way. Or it may select to use a rational strategy to maximise its utility if the agent works in a competitive way.

The relative importance of the negotiation protocols and strategies varies according to the negotiation and environmental context. In some circumstances, the negotiation protocol is the domain concern. For example, the system designer may determine that the negotiation is best organised using a particular form of action. This mechanism, design choice, constrains the type of operations that can be performed on the negotiation object and prescribes the behaviour of the agent's decision-making models. In other cases, however, the agent's strategy is the domain concern. The protocol does not prescribe an agent's behaviour, and there is scope for strategic reasoning to determine the best course of action. In such a case, the relative success of two agents is determined by the effectiveness of their individual strategies.
3.4.3 Negotiation Models

To make agent systems work in the complex environment context, different negotiation models have been developed from various aspects, either for DPS or for MAS problems. For example, negotiation has been examined under a game-theoretic approach in which every agent knows all relevant information about other agents (Genesereth et al., 1986; Zlotkin and Rosenschein, 1989; Kraus and Wilkenfeld, 1991), and under conditions where agents are hostile and completely unwilling to share private information (Sycara, 1988). Various points along the cooperation/hostility continuum are examined in (Zlotkin and Rosenschein, 1989, 1990). Negotiation can occur among peers (Cammarata et al., 1983; Rosenschein and Zlotkin, 1994), through a mediator or arbitrator (Sycara, 1988), or hierarchically through an organisation (Davis and Smith, 1983; Durfee and Montgomery, 1990).

By focusing on the design of a negotiation mechanism, the following section examines a few important negotiation protocols and related strategies in both DPS and MAS domains. Jennings et al. (1999) classify the current agent negotiation models as AI, game theory and psychology-based negotiations. Shen et al. (2001) make a more thorough classification, which includes contract, plan, market, game theory, and AI-based negotiations. With the quick development of agent negotiation systems, many models are particular for their application environments, and thus not easily classified.

3.4.3.1 Contract-Based Negotiation

Smith (1980) introduces a simple negotiation mechanism among co-operative agents in DPS environment, called Contract Net Protocol (CNP), in which an agent having some work to subcontract broadcasts an offer and waits for other agents to send bids. After some delay, the best offers are retained and contracts are allocated to one or more contractors who process their subtasks. The contact-net protocol provides for co-ordination in task allocation, with dynamic allocation and natural load balancing. This approach is quite simple, and can be efficient.
However, the CNP fails to capture many intuitive and important aspects of the negotiation process. For example, bidders cannot counter-propose better options, they cannot modify any of the service agreement parameters, and the emphasis in devising a complete specification is placed solely with the task manager. Also, when the number of nodes is large, the number of messages on the network increases, which can lead to a situation where agents spend more time processing messages than doing the actual work, or worse, the system stops through being flooded with messages.

Various improvements and extensions to the basic CNP have been proposed. In the most basic approach, the choice of a contractor is done by comparing bids corresponding to a particular offer, using whatever mechanisms are relevant to the problem. For example,

- Malone et al. (1988) developed a Distributed Scheduling Protocol (DSP) by overlaying CNP with an economic model. They introduce a motivation framework in the terms of economic theory, and provide a more theoretical term in which to discuss the task-sharing algorithm, while the autonomy agents willingly bid for tasks without explicit motivation in Smith’s (1980) original work. DSP includes two primary dimensions: a) contractors select manager’s tasks in the order of the tasks’ numerical priorities, and b) managers select contractors that satisfy the minimum requirements to perform the job.

- Sandholm (1993) presents a modified version of the CNP for competitive agents in the transportation domain. It provides a formalisation of the bidding and the decision awarding processes, based on marginal cost calculations based on local agent criteria. More importantly, an agent will submit a bid for a set of delivery tasks only if the maximum price mentioned in the tasks’ announcement is greater than what the deliveries will cost that agent.

- A more general approach proposed by Sandholm and Lesser (1997b) is to develop protocols with continuous levels of commitment based on a monetary penalty method, where commitment varies from the original bounded to breakable as a continuum by assigning a commitment breaking cost to each commitment separately. This cost can increase with time, decrease as a function of acceptance time of the offer, or be conditions on events in other negotiations or the environment.
3.4.3.2 Plan-Based Negotiation

Plan-based negotiation is based on co-operation protocols and strategies for resolving conflicts among the plans of a group of agents. As an important negotiation technique, it suffers from limitations inherent in centralised or distributed multi-agent planning. Durfee and Lesser (1991) developed a Partial Global Planning (PGP) approach, which requires agents to exchange descriptions of intermediate situations and results, enabling them to check for potential task overlaps, and decide which agent should do what work. PGP is a flexible and dynamic approach to co-ordination that does not assume any particular distribution of sub-problems, expertise, or other resources. Agent interactions take the form of communicating plans and goals at an appropriate level of abstraction. These communications enable a receiving agent to form an expectation about the future behaviour of a sending agent, adjusting its own local planning appropriately, thus improving agent predictability and network coherence.

Conry at al. (1986) proposed a negotiation protocol called multi-stage negotiation for co-operatively resolving resource allocation conflicts. They were specifically concerned with negotiation strategies for distributed constraint satisfaction problems, where a group of agents have a goal, but each agent has only limited resources. The local constraints give rise to a complex set of global and inter-dependent constraints. This investigation is done in the context of the monitoring and control of a complex communication system. Their implementation involves developing algorithms for multi-agent planning, taking the inevitable conflicts into consideration.

Kreifelt and Von Martial (1991) developed a negotiation approach, where negotiation contains two-stages: first, agents plan their activities separately, and then secondly, co-ordinate their plans. A separate co-ordination agent carries out the co-ordination of all the agents' plans. The negotiation protocol is described in terms of agent states, message types and conversation rules. The problem of this approach is that it does not actually present a negotiation model but just prescribes one, and it is really left to the agents to achieve consensus.
3.4.3.3 Market-Based Negotiation

The goal of market-based negotiation is to resolve a distributed resource allocation problem. Agents are classified as producers and consumers of goods and services. Equilibrium is reached when the price of goods is such that all resources are being used up. A particular agent wants to acquire goods, but is limited by a budget. Thus, it will make offers based on the current price of goods and its own preferences. It has an internal utility function, and its goal is to increase utility, which corresponds to the hypothesis of rational behaviour. Producers have a specific production technology, and seek to maximise their profits. Given a set of prices, the trading process involves a sequence of offers in which each consumer states how much of each resource it wants to purchase. If the demand differs from the supply, then prices will have to be adjusted by the producers (Shen et al., 2001).

Mullen and Wellman (1996) adopt a market-based negotiation for a digital library service. In this approach, an alternative information service is treated as competing with economic activities. Given a measure of priorities over the end-user services provided, the various agents effectively compete to provide the highest level of service using the minimal computational resources. One central capability of the agent is thus to be able to reach agreements on suitable compensation. The goal is to achieve an efficient overall allocation of resources towards the optimal provision of services to users.

One of the drawbacks of the market-based approach using prices as a primary controlling mechanism is that the convergence process may be slow, involving a large number of offers (and computations). A new approach focusing on resources rather than prices was introduced by Ygge and Akkermans (1998). It appears to be complex but efficient. One appealing feature of the new approach is that, in each iteration, the computed scheme is feasible although not optimal. In project KASBAH (Chavez et al., 1997), negotiation is done between one buyer and one seller, there are no globally maintained prices and the information that each agent can have is limited, which is more in line with actual web applications. Lee (1998) proposes a strategy of risk redistribution, trying to minimise the amount of computation in each agent. Other approaches based on probability theory have been proposed by Ekenberg et al. (1995).
Considering the natural link between the market-based approach and the contract net protocol, contract net protocol has been incorporated with various market-based strategies depending on the type of contract an agent can consider such as: the de-commitment of contact is discussed by Sandholm and Lesser (1995). Various types of strategies exist; some of them have been studied by Matos et al. (1998): time dependent, resource dependent, or behaviour development.

3.4.3.4 Game Theory-Based Negotiation

Although various negotiation mechanisms have been developed based on the above approaches, these negotiation models are mainly used to resolve conflicts in the DPS domain. They are far from satisfying the requirements of real world problems. Hence, many negotiation mechanisms hence have been developed for MAS. Most of them (Roth, 1979; Kraus, 1993, 1995; Rosenschein and Zlotkin, 1991, 1994; Pena-Mora and Wang, 1998; Ugwu, 1999) are based on game theory.

There is a particular match between game theory and agent-based systems. One of the major assumptions of game theory is that players are rational. When game theory is applied in analysing human negotiation, the problem regularly faced is that human beings do not always act rationally and frequently do not have consistent preferences over alternatives. On the other hand, agents, being pre-programmed in their behaviour, make concrete the notion of “strategy” which plays a central role in game theory - the idea that a player adopts rules of behaviour before starting to play a given game, and that these rules entirely control its responses during the game (Rosenschein and Zlotkin, 1994). Moreover, the game solutions, like equilibrium points, stable strategies and Pareto-optimal solutions are also the basis of the MAS negotiation solutions.

Game theory has some other essential assumptions, such as utility maximisation, complete knowledge, isolated negotiation, inter-agent comparison of utility, symmetric ability, binding commitments, and no explicit utility transfer. These

---

3 Here, the game theory-based negotiation actually covers the economic theory-based negotiation. A detailed discussion is in Chapter 4.1.
assumptions are also the basis for game theory-based negotiations. Zlotkin and Rosenschein (1991, 1994) explain that their work on MAS followed the general direction that treats negotiation in the spirit of game theory, while altering game theory assumptions which are irrelevant to MAS. The main point is that by appropriately adjusting the rules of the game by which the programs must interact, the private strategies of agents can be influenced. Certain strategies simply become the best for an agent to adopt given the rules of the game (Rosenschein and Zlotkin, 1994). A number of examples are presented in the subsequent paragraphs.

Zlotkin and Rosenschein (1991) study the problem of incomplete information in MAS negotiation where agents need to reach an agreement on task allocation. The incomplete information is either about the opponent’s goals or about the value of its goals. By adopting certain game-theoretic techniques to model communication and promises, they introduce a mechanism that they called ‘one negotiation phase’ in which agents simultaneously declare private information before beginning the negotiation. They also identify situations and protocols where agents have incentives to tell the truth in “one negotiation phase” and cases where it is beneficial for agents to lie. In their study, the process of negotiation was severally restricted, and it assumed that each agent knew the complete payoff matrix associated with the interaction.

Zlotkin and Rosenschein (1994) extend the previous work. They explore various situations in different negotiation domains and the possible negotiation protocols and strategies where they divide negotiation domains into three: task orientated domains (TODs), state orientated domains (SODs) and worth orientated domains (WODs), with each domain being a generalisation of the previous.

- TODs are the simplest case where an agent's activity is defined in terms of the set of tasks it has to achieve. It is assumed that all resources are available to the agent, the benefit of negotiation being the redistribution of tasks amongst a group of agents. There is no possibility of deadlock as all agents can proceed with their original task list and be no worse off. This is clearly limited.

- SODs deal with problems where agents wish to change their environment from an initial state to some goal state. The classic AI Blocks World problem is a good
example. There is the possibility of conflict and deadlock, as agents may have
different goals, and to satisfy all may be impossible or require more effort from
each agent than if they were alone in the world. In this situation agents must be
able to make concessions.

- WODs are domains where agents attach a worth to each potential state. This
allows much more flexible goals to be set and allows concessions to be made on
these goals. An example would be agents in an electronic marketplace where the
goal for a seller may be to obtain the highest price for $x$ within time $y$. There is
again the possibility of conflict and deadlock, but now within a more complicated
bargaining environment.

Within these domains, Zlotkin and Rosenschein (1994) explore three different types
of situation:

- Cooperative, in which the cost to both agents of a joint plan that achieves all goals
  is less than or equal to any individual plan that would achieve a single agent’s
goal;

- Compromise, in which at least one agent will have to pay more to achieve its goal
  jointly than it would to achieve them individually but, given the inevitable
  presence of another agents, a deal can be made that achieve the goals; and

- Conflict, in which at least one agent has to pay an unacceptably high cost for any
  joint plan and, therefore, no deal can be made.

Kraus (1996) studies the multi-interactions among self-motivated agents where agents
do not have complete information. By adopting different game theory techniques, she
develops a strategic model that not only takes into consideration the passage of time
during the negotiation (i.e. time constraints), but also includes belief systems. Using a
distributed mechanism, agents negotiate, and can reach efficient agreements without
delays. She also assumes that the set of possible agreements is limited and that there is
full information, but she makes fewer restrictions on the negotiation procedure. The
study provides strategies for a wide range of situations, which satisfies the criteria:
symmetrical distribution, simplicity, instantaneous, efficiently and stability.
Although game theory provides a theoretical basis for MAS negotiation mechanism, it is not a perfect solution. Problems like complete information assumption, inflexibility, and inadequacy or inappropriateness for real world situations are all barriers to the application of game theory in MAS. As a result, many researchers have attempted to incorporate other negotiation mechanisms like the bargaining theoretic approach, with game theory.

3.4.3.5 AI-Based Negotiation

Several common AI approaches have been adopted in developing agent negotiation mechanisms such as: case-based reasoning (Sycara, 1987), negotiation search (Lander and Lesser, 1991, 1993) and knowledge-based approach (Werkman, 1990).

Sycara (1987) presents a model of negotiation, called Persuader, that operates in the domain of labour negotiation. It involves three agents (i.e. a union, a company, and a mediator), and is inspired by human negotiation. It models the iterative exchange of proposals and counter-proposals in order for the parties to reach agreement. Agents can modify other agent's beliefs, behaviours and intentions via persuasion. Each agent's multi-dimensional utility model is private knowledge. Belief revision to change the agent's utilities so that an agreement can be reached via persuasive argumentation. A case-based reasoning mediator is incorporated into the model with the idea that human negotiators make decisions with reference to past negotiation experiences.

Lander and Lesser (1991, 1993) develop a negotiation system, called TEAM, that explores negotiation search for conflict resolution among heterogeneous and reusable agents in the domain of DPS. In TEAM, an extended search is carried out by an agent recognising a conflict in order to find another solution in its local search space that avoids the conflict, while relaxation is used to expand the local solution space. In the negotiated search, loosely coupled agents interleave the tasks of: local search for a solution to some sub-problem; integration of local sub-problem solutions into a shared solution; information exchange to define and refine the shared search space of the agents, and assessment and reassessment of emerging solutions. A selected solution is acceptable if it satisfies the requirements imposed by each agent.
Werkman (1990) propose a knowledge-based model of incremental negotiation in DPS domain. This scheme uses a shared knowledge representation, which allows agents to negotiate in a similar manner to co-operating experts with a common background of domain knowledge. Essentially, it explores a blackboard having partitions for requested proposals, rejected proposals, accepted proposals, communications, and shared knowledge. The negotiation in this model follows a procedure of proposal, evaluation and counter-proposal. An arbitrator agent is introduced to help agents to resolve possible deadlocks, by reviewing their negotiation dialogue and using their mutual information network to generate alternative proposals. This is done using issue relaxation techniques or some intelligent proposal generator approach. This approach may fail to achieve resolution in which case the arbitrator may set time limits or use other techniques.

3.4.3.6 Psychology-based negotiation

Behaviour negotiation theory is an important approach in analysing negotiation because it reflects the human negotiators’ psychological responses during negotiation. Soci-psychology aspect is always a central point in analysing human negotiations. Bussamnn and Muller (1992) present a cyclic negotiation model in the DPS domain based on Gulliver’s (1979) eight phases of negotiation process. This model addresses the limitation of other negotiation proposals and models such as market, game theory and AI approaches. The cyclic nature of the model also addresses the thorny issue of conflict resolution. The general strategy is that negotiation begins with one, some or every agent making a proposal. Next, the agents evaluate and check the proposals against their preferences, and criticise them by listing any of their preferences violated by the proposal. The agents then update their knowledge about the other agent’s preferences and the negotiation cycle resumes with a new proposal or proposals in the light of this newly learned information. Conflicts between agents are handled in a concurrent conflict resolution cycle.
3.4.3.7 Other Approaches

Besides the above negotiation approaches, many other negotiation mechanisms have been developed to address different aspects of negotiation in different environments. These include the heuristics approach (Kraus and Lehmann, 1995), the flowchart approach (Polat et al., 1993) and the argument-based approach (Sierra et al., 1998).

One of the more important ideas is the argument-based negotiation. Sierra et al., (1998) proposed an argument-based negotiation in the domain of business process management. The emphasis is on how agents can justify their negotiation stance, and how agents persuade one another to change their decisions. Negotiators provide arguments to support their stance. Thus, in addition to generating proposals, counter-proposals and critiques, the negotiator is seeking to make the proposal more acceptable by providing additional meta-level information in the form of arguments for its position. The nature and types of the arguments can vary enormously. However, common categories include threats, rewards, and appeals. Whatever its precise form, the role of the supporting argument is either to modify the recipient's region of acceptability or its rating function over this region. In so doing, arguments have the potential to increase the likelihood and/or the speed of agreements being reached. In the former case, this is by persuading agents to accept deals that they may previously have rejected. In the latter case, this is achieved by convincing agents to accept their opponent's position on a given issue (and to cease negotiating over it).

3.4.4 Negotiations in DPS and MAS

Although the above discussions have, explicitly or implicitly, addressed the working domain of each negotiation mechanism, it is necessary to clarify the differences between the negotiation mechanism in these two domains.

The negotiations in DPS domain are often termed as "co-operative negotiation". They are often used in situations where agents have "a global goal/single task envisioned for the system" (Smith and Davis, 1983). System designers impose an interaction protocol and a strategy for each agent. The main question is what social outcomes follow given the protocols and assuming that agents use the imposed strategies.
Systems like Contract Net, Distributed Scheduling and some market-based approaches are typical examples of negotiation in DPS.

The negotiations in the MAS domain are often regarded as "competitive negotiation". They are often used in situations where "agents of disparate interests attempt to make a group choice over well-defined alternatives" (Rosenschein & Zlotkin, 1994). The agents are provided with an interaction protocol, but each agent will choose its own strategy to maximise its own good without concern for the global good. The protocols need to be designed using a non-cooperative strategic perspective. The main question is what social outcomes follow given a protocol that guarantees that each agent's desired local strategy is best for that agent and thus the agent will use it. This approach is required in designing robust non-manipulation MAS where agents are constructed by separate designers and represent different real-world parties. Compared with negotiation in DPS, such self-interest naturally prevails in negotiations among independent businesses or individuals. In building computer support for negotiation in such settings, the issue of self-interest has to be dealt with.

To improve the quality of the MAS negotiation mechanism, the adoption of proper negotiation theoretic models is crucial. Such models should match the nature of negotiation objects, suit the application environment, and reflect the characteristics of MAS. The negotiation cannot be tackled by technological or economic methods alone. Instead, the successful solutions are likely to emerge from a deep understanding and careful integration of both.

3.5 LEARNING IN MULTI-AGENT SYSTEMS

A major problem in the development of multi-agent systems is the difficulty for a developer to foresee all potential situations an agent could encounter and specify agent behaviour optimally in advance. Therefore it is widely recognised in the agent community that one of the more important features of high level agents is their capability to adapt, to learn, and to modify their behaviours. Weiss (1993) concludes that the two important reasons for learning in multi-agent systems are: to be able to endow artificial multi-agent systems with the ability to automatically improve their
behaviour; and to get a better understanding of the learning processes in natural multi-agent systems.

3.5.1 Overview

As a group, agents work in an open, complex and dynamic environment, which results from a number of factors such as (extended from Prasad and Lesser, 1999):

- **environment uncertainty**: it is impossible to define all conditions before the systems start to work;
- **dynamic environment**: the system exists in an environment whose conditions vary over time;
- **communication constraints**: every communication link has limited parameters such as range and bandwidth and a certain noise level;
- **degree of clustering**: in the case of a larger number of agents, it is advantageous to divide them into groups according to their functions; however, this functional grouping is limited;
- **time stress**: the time for decision making is not infinite, especially in real time systems the question of quick response plays a vital role;
- **option multiplicity**: it represents the number of planning options available to each agent;
- **density of the solution space**: it represents the ratio of acceptable, conflict-free plans to the number of potential plans;
- **complexity of interactions between agents**: an agent's activities might lead to a change in the other agents' decisions; and
- **varying goals, abilities, preferences, skills and levels of knowledge of individual agents**.

Agents in such a system face uncertainties due to their partial views of the other agents and the environment. Incomplete information about the progress, characteristics, expectations, or preferences of the other agents, and generated partial results may lead to global incoherence and degradation in system performance (Weiss, 1996). To effectively utilise opportunities, agents need to learn about other agents and adapt their local behaviour based on group composition and dynamics.
As a result, there has been considerable research on MAS learning for particular applications such as: the prisoner's dilemma (Sandholm and Crites, 1995), predator/prey (Nagayuki et al., 2000), agent co-ordination (Prasad and Lesser, 1999), engineering design (Grece and Brown, 1998), and negotiation (Zeng and Sycara, 1998), with varying degree of success. The following sections explore the major characteristics of, and approaches to agent learning, with a focus on the learning during negotiations.

3.5.2 Issues of Learning

Learning in MAS can be defined operationally to mean the ability to perform new tasks that could not be performed before or to perform old tasks better as a result of changes produced by the learning process. In a stronger and more specific meaning, "multi-agent learning" refers only to situations in which several agents collectively pursue a common learning goal. In a weaker and less specific meaning, "multi-agent learning" additionally refers to situations in which an agent pursues its own learning goals, but is affected in its learning by other agents, their knowledge, beliefs, and intentions (Weiss, 1996). Agents learn in a communal way; their learning is influenced by individual goals and preferences, exchanged information, shared assumptions, commonly developed viewpoints, environment, social and cultural convention and norms which regulate and constrain their behaviours and interaction.

To conduct effective learning in MAS, several important issues need to be addressed before and during the learning process such as: the objectives and focuses of learning, the key components of learning, and the approach to learning.

3.5.2.1 Objective of Learning

The learning objective is highly dependent on the application domains and the goals of each individual agent. An agent in DPS may learn from the others and the environment, and adapt its behaviour for better co-operation or solution of a specific
problem; whilst an agent in MAS often learns for its own benefit. Therefore, the learning objectives and learning approaches in different systems are often different.

The items which an agent expects to learn from others could be the preferences, utility functions, risk attitudes, tasks, strategies, sequences of actions or plans, specific domain knowledge, prediction of decisions, types of conflicts, and so on. For example, in a MAS negotiation domain, an agent may expect to achieve the following objectives through learning:

- **Changing own beliefs and learning about others’ beliefs:**

  Agents can hold different beliefs about the same fact. As a result of the knowledge and information exchanged during the negotiation, an agent can change its beliefs, if the new belief is supported by more powerful evidence. The change of beliefs through learning can imply significant changes in the proposal generated by an agent. As beliefs are seen as directly related to preferences, learning in this direction can influence the preferred order of an agent’s decisions in a multiple solution situation. Assuming that an agent has acquired knowledge from the external environment, it can influence other agents to revise their beliefs as a result of the negotiation, and thus propagate the external influence (Grece and Brown, 1994).

  Meanwhile, an agent can make inferences about the other agents’ beliefs, and analyse their intentions and further negotiation strategies, so that it can determine its negotiation strategies accordingly. Such learning plays an extremely important role for many agents. Generally, the key negotiation features, like agents’ utility functions, risk attitudes and reservation values represent agents’ beliefs.

- **Learning negotiation strategies**

  Depending on different negotiating situations, the negotiating agents not only need to understand the opponents’ beliefs and intentions, but also are urged to know the opponents’ negotiation strategies. Or in some other cases, both parties’ objectives are relatively apparent, negotiation results depend on the negotiation strategies taken by
the negotiating agents. However, to understand the negotiation strategies is one of the most complex problems even if the agent has some knowledge about the opponent's beliefs. In multi-step negotiations, agents could change their strategy dynamically, and it is very hard to tell whether and when a change of strategy occurs (Cross, 1977). By analysing the difference between the predicted response and the actual response, an agent can adjust its perceptions about the opponents' strategies.

Grecu and Brown (1994) emphasise the importance of learning negotiation strategies for the learning agent itself. The results of negotiation could be used to evaluate the quality of specific negotiation steps. This could reinforce the agent's drive to use the same sequence of actions in a similar situation or, alternatively, to weaken that drive. More generally, a negotiation history can be used to extract and to compile particularly useful sequences of negotiation actions. Agents might recognise the applicability of a strategy at moments where the existing commitments prevent the execution of the initial steps of that strategy. Otherwise, the strategy might prove to be inefficient if used continuously during negotiation. In addition, an agent's observation of the other agents' actions, guided by its strategies, may eventually lead to the synthesis of new strategies.

- Learning conflict patterns

Based on participation in different conflict resolution processes, an agent can learn to recognise and classify different types of conflict. This implies learning about the context leading to conflicts and learning the characteristics of conflict. The first factor is important for avoiding conflicts, the second one for taking negotiation decisions. The negotiation model proposed by Klein (1991) uses a hierarchy of possible types of conflicts between agents. An inductive learning approach could be used to automate the construction of such a classification scheme and to facilitate adding new conflict types.

Although these learning cases, especially the first two, represent some of the essential learning objectives in MAS negotiation systems, the learning objectives could be various depending on different application domains such as: negotiation issues, domain knowledge or environment issues.
3.5.2.2 Key Elements in Learning

Learning in a multi-agent environment can be a very complex process, considering that agents learn mutually, and the environment continuously changes. Moreover, agents' actions and strategies are often not directly observable, the action taken by the learning agent can strongly bias the range of behaviours that are encountered (Grecu and Brown, 1998a). However, no matter how complex the learning process is, there are always three essential components in the process, on which the learning agent bases its learning. These are agents' expectations, feedback information, and credit assignment to evaluate the feedback information and decision-making.

- Expectations:

Expectations are the bases for agent learning. In a general MAS domain, expectations represent an agent's beliefs that events will occur in a pre-defined way. Expectations encode the agent's current knowledge of an event and the global environment in which it operates, and represent a basis for action in a partially observable and partially computable world (Grecu and Brown, 1998b). Expectations also reflect the criteria through which the agent relates to the environment. The expectations of an agent guide its decision-making during the operation process. In a negotiation domain, an agent's expectations determine what and how much the agent expects to get from the others, or what the others will do.

On the other hand, an agent's expectations are limited in its anticipatory power due to the constraints imposed on perceiving the other agents and the environment. During the system operation process, an agent plans to perform a task according to its expectations, but may fail to do that because an event does not occur or because other agents respond in an unexpected manner. It means the agent's expectations are violated, either in a good or a bad sense. The violation of the expectation indicates that the agent's knowledge about other agents, events, or environment has limited validity. If not noticed or taken into consideration, the agent will repeatedly fail in outlining and implementing its actions. A continuous learning process makes it
possible for an agent to modify its expectations to be more realistic during the negotiation process.

Hu and Wellman (1996) characterise an agent's belief process in terms of conjectures about the effect of their actions. A conjectural equilibrium is then defined, in which all agents' expectations are realised, and each agent responds optimally to its expectations. They present a multi-agent system where an agent builds a model of the response of others. Their experimental results show that depending on the starting point, the agent may be better or worse than had it not attempted to learn about/from the other agents.

- Feedback

The availability of information is a primary requirement for the learning process. Sound and unbiased feedback information provides a learning agent with resources about the other agents' perceptions, the properties of events, and the system working environment. Feedback can originate from direct communication with other agents, or indirectly, mediated by intermediary agents, or without communication, directly through the learning agent's observations of the effects of its decisions and other agents' actions. Feedback can be biased by the path to the receiver. Feedback may also contain different information or be from several sources, and therefore its effect depends on how these sources are used: filtered, independently or in a combined manner. Feedback can also be affected and reduced by processes such as conflicts and backtracking, or hidden through social decision making schemes.

- Evaluation Criteria

Evaluation criteria are closely related to the expectations of the learning agent. A basic problem that any learning system is confronted with is how the agent evaluates the feedback from the others as a response to the agent's last decisions or actions. In a more general sense, it is a problem of properly assigning credit for overall performance changes to each of the system activities that contributed to those changes (Weiss, 1996). The selection of performance criteria shapes the direction in which the
system will evolve. The evaluation problem can be usefully decomposed into two sub-problems: the assignment of credit for an overall performance change to external actions, and the assignment of credit for an action to the corresponding internal decisions.

3.5.3 Methods of Learning

The approach to learning is the most essential problem for any MAS learning problem. Most current approaches are extended from machine learning methods. A few researchers have made detailed analyses:

- Winston (1997) has drawn a big picture of the existing learning methods in MAS (Figure 3.3). Most of the efforts in AI focus on the first two classes of methods, with recent emphasis on the second class.

![Figure 3.3. The classification of learning methods for MAS (modified from Winston, 1997)]
Carbonell (1989) makes a summary of the major machine learning paradigms which have a high potential of being applied in MAS. They are:

a) Inductive learning: acquiring concepts from sets of positive and negative examples;

b) Analytic learning: explanation-based learning and certain forms of analogical and case-based learning methods, deductive methods, and analytic methods;

c) Genetic algorithms: classifier systems; and


Jennings et al. (1999) conclude that the previous work related to learning in MAS is limited and much of this work relied on techniques derived from reinforcement learning, genetic algorithms and classifier systems.

Grecu and Brown (1998a) list the learning approaches in the MAS engineering design domain as: explanation-based learning, concept/induction learning, knowledge compilation, multi-strategy learning, case-based learning, reinforcement learning, generic algorithm learning, and neural networks. Some of these are briefly discussed below:

a) Reinforcement learning is based on the idea that the tendency to produce an action should be reinforced if it produces favourable results, and weakened if it produces unfavourable results. It uses few little computation resources per example but requires a large number of examples.

b) The concept learning model developed by Shaw (1996) closely follows the paradigm of collective group induction. Agents develop different hypotheses of concept description by seeing a set of training instances. Agents differ in the rules they use to generate their hypotheses. Hypotheses are periodically integrated into group hypotheses that achieve a better accuracy than the individual agent hypotheses.

c) Many multi-strategy learning approaches use an initial learning phase to achieve acceptable levels of performance in agents. The agents’ knowledge is
then used to seed and evolve an agent population with refined performance. Gordon and Subramanian (1994) describe an agent learning approach in an embedded adversarial multi-agent setting. The first learning phase uses compilation to operationalise the agent's task knowledge. The second phase refines the rule sets representing the agents' task strategies through a genetic algorithm approach.

d) The genetic algorithm approach has lead to a significant amount of experimental and theoretical results. Grefenstetter and Daley (1996) describe experiments with co-evolutionary approaches, that are similar to an ecological environment where species evolve during the interactions with each other. Their goal is to design behaviour strategies for intelligent robots in MAS environments. Haynes and Sen (1997) analyse crossover operators and fitness functions that allow rapid evolution of agents with good task performance.

- Weiss (1996) has undertaken a comprehensive study about the possible learning approaches in MAS according to the learning method and learning feedback. According to the learning method or strategy by a learning entity, the following methods are usually distinguished. A major difference between these methods lies in the amount of learning effort required by them, increasing from top to bottom:

a) Rote learning: direct implantation of knowledge and skills without requiring further inference or transformation from the learner;

b) Learning from instruction and by advise taking: operationalisation-transformation into an internal representation and integration with prior knowledge and skills - of new information like an instruction or an advice that is not directly executable by the learner;

c) Learning from examples and by practice: extraction and refinement of knowledge and skills like a general concept or a standardised pattern of motion from positive and negative examples or from practical experience;

d) Learning by analogy: solution-preserving transformation of knowledge and skills from a solved to similar but unsolved problem; and
e) Learning by discovery: gathering new knowledge and skills by making observations, conducting experiments, and generating and testing hypotheses or theories on the basis of the observational and experimental results.

According to the learning feedback that is available to a learning entity, learning can be distinguished as:

a) Supervised learning: the feedback specifies the desired activity of the learner and the objective of learning is to match this desired action as closely as possible;

b) Reinforcement learning: the feedback only specifies the utility of the actual activity of the learner and the objective is to maximise this utility; and

c) Unsupervised learning: no explicit feedback is provided and the objective is to find out useful and desired activities on the basis of trial and error and self-organised processes.

In all three cases, the learning feedback is assumed to be provided by the system environment or the agents themselves. This means that the environment or an agent providing feedback acts as a teacher in the case of supervised learning and as a "critic" in the case of reinforcement learning; in the unsupervised learning, the environment and the agents just act as passive 'observers'.

Also, learning in MAS can also be analysed according to other criteria such as:

- the purpose and goal of learning (i.e. learning to improve a single agent's skills and abilities, or to improve the agent system's coherence and co-ordination as a whole);

- the categories of learning (i.e. only one of the agents gets involved in the learning process, or all available agents are involved); and

- an agent's involvement in a learning process (the involvement of an agent is essential or not for achieving the pursued learning goal).

These learning paradigms emerged from quite different scientific roots, employ different computational methods, and often rely on different ways of evaluating
success. Some of the learned elements have generality which makes them applicable to any type of negotiation partner, some will apply only to agents of a particular type, while other learned elements represent information useful only when negotiating with specific agents. Furthermore, different agents do not necessarily adopt the same learning method or the same type of learning feedback. In the course of learning, an agent may employ different learning methods and types of learning feedback. Finally, learning can occur not only during the negotiation process, but also afterwards. The negotiation history together with some evaluation techniques for the agent's past actions can be used to classify agents as more or less successful.

### 3.5.4 Challenging Research Issues

Although agent learning has gained a high interest in the MAS research community, the current learning approaches are very limited. For example, most current approaches to learning in MAS have been using purely reactive architectures where agents base their actions just on the current situation and not on the previous history, and there is no notion of deliberate planning towards an explicit goal (Kaelbling et al., 1996). Such approaches are less than optimal in complex domains where knowledge-based planning and complex co-ordination is necessary. In other words, although these learning techniques can be successful in restricted domains, they strip agents of the ability to adapt in a domain-dependent fashion, based on background knowledge of the respective situation. This ability is crucial in complex domains where background knowledge has a large impact on the quality of the agents' decision making.

Besides the learning techniques, Weiss (1996) points out some other aspects which challenge agent learning in MAS:

- requirements for learning in MAS;
- principles and concepts of learning in MAS;
- models and architectures of MAS capable of learning;
- extension and transformation of single-agent learning approaches to MAS learning approaches;
• parallel and distributed inductive learning in MAS;
• multi-strategy and multi-perspective learning in MAS;
• learning in MAS as organisational self-design; and
• theoretical analysis of learning in MAS.

3.6 MAS IN THE CONSTRUCTION INDUSTRY

This section addresses two major aspects of the application of MAS in the construction industry. One is the necessity, possibility and advantages of MAS in solving construction problems. Another is how to apply MAS in the industry. A few applications are described.

3.6.1 Why MAS?

One of the most important driving forces behind MAS research and development is the technology push of a growing standardised communication infrastructure - Internet, WWW, KQML (Knowledge Query and Manipulation Language), and XML (eXtensible Markup Language). The Internet is particularly important as it enables separately designed agents, often belonging to different organisations, to interact in an open environment in real-time and carry out transactions safely. Another reason is the strong application pull for computer support for negotiation at the operative decision-making level.

MAS are not required merely to produce modularity (though they reduce complexity), extra speed (though this may be an effect of their inherent parallelism), reliability (though they provide redundancy), flexibility or re-usability (Newell, 1982). In the same way, they are not required simply because a problem is too large for a centralised single agent due to resource limitation, nor because of the sheer risk of a centralised system, nor merely for reasons of efficiency, heterogeneous reasoning, etc. Problems requiring multi-agent solutions include the following (Sen, 1997; Nwana and Ndumu, 1999):
Problems requiring the interconnecting and inter-operation of multiple, autonomous, "self-interested" existing legacy systems, e.g. expert systems or decision-support systems.

Similarly, problems whose solutions draw from distributed autonomous experts. Agents in a multi-agent system are often designed based on functionality: such systems are modular in design and hence can be easier to develop and maintain.

Problems that are inherently distributed in nature (e.g. distributed sensor interpretation, co-ordination of self-interested agents, etc.).

Problems whose solutions require the collation and fusion of information, knowledge or data from distributed, autonomous and selfish information sources.

A key benefit for the application of MAS in the construction industry is that MAS provides a decentralised approach to modelling the fragmented construction engineering and management problems. Such fragmented problems, though widely recognised, are difficult to be solved using other approaches. The idea of incorporating MAS into the construction industry provides a novel approach to tackling distributed problems. Some other key benefits could be: effective decomposition of large-scale problems; improved collaborative and concurrent working; and easier and cheaper access to specialist information. The use of agent-based systems is expected to result in increased competitiveness of the construction industry as the decentralisation of complex, large-scale problems and the collaborative input to their resolution, will lead to better quality, more economic, safer and more optimal solutions.

3.6.2 Some Applications

Given the potential benefits that MAS could bring to the industry, several research projects have been conducted, attempting to resolve the conflicts between project participants, to facilitate decision support at various stages of a construction project, or to reach a better solution for construction engineering problems. Two main areas being addressed are engineering design and negotiations.
Much of the work in the area of automated design has been done on building design support. Focus is on co-ordinating the activities and information flow between designers who are often geographically distributed, and on modelling agents to facilitate workflow between architectures and engineering in a collaborative design environment. Some of the developed systems are described as follows:

- Fenves et al., (1994) developed an Integrated Building Design Environment (IBDE) project. Agents in IBDE are classified into two groups: generators and critics. The generators typically are knowledge-based systems that contribute to the development of the emerging design such as the ‘ARCHPLAN‘ (developing the building design concept), the ‘CORE‘ (generating layouts of the service core), and the ‘STRPES‘ (configuring the structure system). The critics do not contribute directly to the design descriptions, but evaluate the current description and make recommendations for redesign such as constructibility assessment by the ‘CONSTRUCTION CRITIC‘ and structure evaluation by the ‘STRUCTURAL CRITIC‘. Each generator and critic agent is described according to the role it plays in the overall project. As an information-processing unit, each agent is first described in terms of its principal inputs and outputs, then is further described by the problem-solving paradigm it adopts, and how it transforms inputs into solutions.

- Chiou and Logcher (1996) have implemented an agent-based system for design. Agents have areas of specialist knowledge and perform design and checking tasks. They interact directly with a user who is responsible for design changes. However, there is no provision for direct negotiation between agents during the design process, so the number of designs evaluated remains small. Hence, convergence to a near optimal design depends on the user and does not make full use of the available computational power which can allow for the evaluation of many slightly differing designs automatically.
• Heckel, et al. (1996) developed an Agent Collaboration Environment (ACE). The ACE supports collaboration amongst members of the design team by providing the infrastructure for a community of cooperative design agents that assist the users. In the framework, agents are organised into business processes to reflect the various functional tasks and workflow between different design teams in an organisational setting. Agents communicate with each other using libraries of design objects such as beams, columns, and footings. Agents are reactive, but have the capability to run in the background and advise users on design issues such as code violations. Heckel (1996) metaphorically described the ACE as “a gathering place of several ‘experts’ represented as agents. Each agent has particular expertise in a relatively narrow domain, but the groups of agents are all working on a common problem” with each agent embodying a particular expertise that must be acquired from source. The primary role of an ACE agent is as “design assistants that use heuristic rules and a powerful checklist facility to automate routine design tasks, enhancing productivity and design quality”.

• Radeke (1997) describes the GENIAL project (Global Engineering Network Intelligent Access Libraries). The objective of the project is to facilitate large-scale collaborative engineering by establishing a common semantic infrastructure that will enable heterogeneous software systems to communicate and exchange information. An example of such system-system communication is automated database transactions at enterprise-level. This will enable enterprises from different engineering sectors to combine internal knowledge with engineering knowledge accessed on-line and world-wide via the global engineering network. It is expected that the project will lead to domain-specific toolkits for acquisition, retrieval and presentation of internal and external engineering knowledge.

• The DESSYS (2000) project is part of a wider research program, Virtual Reality Design Information System. DESSYS investigates the deployment of multiple software agents to improve collaborative decision-making in multidisciplinary architectural design environment. This research covers knowledge modelling for a decision support system in geotechnical design. It also investigates some of the important issues for decision-support in construction. The first issue is on different
techniques of transferring knowledge and how the knowledge is used in the process of making decisions in structural-design. The second issue deals with formalising expert-knowledge in such a way that it can be used as decision-knowledge in the early phases of a multidisciplinary building-process. The final phases deal with developing a knowledge-based agent, and validating the prototype.

3.6.2.2 Negotiation

Negotiation is the core of many MAS in construction because it is often unavoidable between different project participants with their particular tasks and domain knowledge whilst they interact to achieve their individual objective as well as the group goals. Furthermore, the importance of negotiation in MAS is likely to increase. One reason is the growth of fast and inexpensive standardised communication infrastructures, over which separately designed agents belonging to different organisations, can interact in an open environment in real-time, and safely carry out transactions. Secondly, there is an industrial trend to be able to respond to larger and more diverse orders. Such ventures can realise the resource allocation efficiently and are easier to adapt to a dynamically changing economic environment. The following are negotiation algorithms developed in construction.

- Ugwu et al. (1999) developed an ADLIB model in which agents are used to model construction design project. Each agent needs to satisfy its minimum specification requirement, while it may give up something for the other participants to reach their minimum requirements. The relationship between agents lies between fully co-operative and fully hostile. Different design teams co-ordinate their requirements through negotiations. The monotonic concession protocol (MCP) is adopted in their negotiations, whilst each agent adopts a gradient concession strategy given the MCP protocol.

- Pena-Mora and Wang (1998) develop a CONVINCER model to facilitate the negotiation of conflict resolution in large-scale civil engineering projects. Essentially, the model is based on game theory, where a player's actions are based
on the premise that the decision of any player can affect the payoff of all players. Also, the model analyses various conflict situations and settlement solutions. The negotiation model contains two steps. First, each negotiator expresses his interests in the conflict. Once the interests have been expressed correctly, the influence of positions of conflicting interests on the overall negotiation outcome is evaluated using game theory. By following this collaborative negotiation approach, they build an agent, which facilitates or mediates the negotiation of conflicts in projects.

- Kim et al. (2000) develop a project schedule co-ordination model through compensatory negotiation - a framework wherein a project can be rescheduled dynamically by all of the concerned project participants based on their resource profiles. It is intended to lead to individually rational and globally optimal solutions in agent-based project schedule co-ordination domains. The compensatory negotiation approach allows an agent to transfer utility to other agents for compensation of the disadvantageous agreements through a multi-linked negotiation process. In their study, utility is defined as the difference between benefit (i.e. profit gain for the task from a possible option) and cost (i.e. cost incurred for succeeding task due to the possible option).

- Oliveira et al. (1997) develop a MACIV model. The project aims to design and implement a MAS, enabling decentralised management of the different resources in a construction company. In their system, negotiation follows a six-step process, i.e. announcing, task evaluation, selection phase, market manipulation and price adjustment. The last two steps are repeated until all the agents except one get out of the process, or some timeout arrives. Each agent makes decisions according to its estimate of the cost for the execution of each task in terms of the travel cost, depreciation cost, operation cost, operator cost and profits. A co-ordinator agent is introduced in the system.

A common drawback in these negotiation algorithms is that they often fail to address the complex issues of negotiations in construction. For example, although the CONVINCER model focuses on the negotiation resolution of construction conflicts,
each party's decision-making is finally determined by a simple game theory rule, which can hardly represent the real negotiation situations. A more sophisticated negotiation mechanism needs to be developed for construction claims negotiation - the focus of this thesis.

3.6.3 Problems and Challenges

Although multi-agent systems provide many potential advantages, they also face many difficult challenges such as: the information discovery problem, the communication problem, the ontology problem, the legacy software integration problem, the reasoning and co-ordination problem and monitoring problem (Nwana and Ndumu, 1999). To design and implement MAS for industrial problems, developers need to address a number of questions (Gasser and Huhns, 1989; Jennings et al. 1999):

- How to formulate, describe, decompose, and allocate problems and synthesise results among a group of intelligent agents;
- How to enable agents to communicate and interact. What communication languages and protocols to use. What and when to communicate;
- How to ensure that agents act coherently in making decisions or taking action, accommodating the non-local effects of local decisions and avoiding harmful interactions;
- How to enable individual agents to represent and reason about the actions, plans, and knowledge of other agents in order to co-ordinate with them. How to reason about the state of their co-ordinated process (e.g. initiation and completion);
- How to recognise and reconcile disparate viewpoints and conflicting intentions among a collection of agents trying to co-ordinate their actions;
- How to effectively balance local computation and communication. More generally, how to manage allocation of limited resources;
- How to avoid or mitigate harmful overall system behaviour, such as chaotic or oscillatory behaviour; and

116
• How to engineer and constrain practical multi-agent systems. How to design technology platforms and development methodologies for multi-agent systems.

All these questions are concerned with the key points of the development of the MASCOT model, and will be addressed (more or less) in this project. For example, the first four questions address the critical issues such as how to model the industrial problems in a MAS environment, and how to build the agent co-ordination mechanism based on the identified industrial problem. These questions need to be answered during the conceptual model development process. The answers of the later questions are crucial for the quality of the developed system; and ensure that the system will work properly.

Jennings et al. (1999) puts particular emphasis on two major technical impediments to the widespread adoption of agent technology:

• the lack of a systematic methodology enabling designers to clearly specify and structure their applications as multi-agent systems; and

• the lack of widely available industrial-strength toolkits for building MAS.

The former means that most existing applications have been designed in a fairly ad hoc manner - either by borrowing a methodology and trying to shoe-horn it to the multi-agent context, or by working without a methodology and designing the system based on intuition and past experience. This is the biggest and hardest problem for the widespread application of MAS. What is required is a means of analysing the problem of working out how it can be best structured as a MAS, and then determining how the individual agent can be structured. A clear, non-ad hoc methodology is vital for the further development of MAS.

The latter impediment means that most MAS projects expend significant development effort building up basic infrastructure before the main thrust of agent and inter-agent development can commence. Again, this is an unsustainable position. The position can be alleviated to a certain extent by exploiting existing technologies as, and where appropriate - rather than re-inventing the wheel as often happens at the moment. However, greater support is still needed for the process of building agent-level
features. Thus, a toolkit is required, providing facilities for: specifying an agent's problem solving behaviour specifying how and when agents should interact, and visualising and debugging the problem solving behaviour of the agents and of the entire system.

Solutions to the above problems are intertwined. For example, a different modelling scheme for an individual agent may constrain the range of effective co-ordination regimes; different procedures of communication and interaction have implications for behavioural coherence; different problem and task decompositions may yield different interactions. The application of MAS in the construction industry face all these problems, and have the additional difficulties arising form the flexible and sophisticated interactions between autonomous problem solving components required by the specific industrial problems. The solutions to these problems need to be formed in the context of solving real or quasi-real world problems.

3.7 SUMMARY

This chapter explored the main concepts and characteristics of intelligent agents and multi-agent systems. It first presented the definitions, taxonomies, development process, and organisations of intelligent agents; it then discussed the major characteristics, disciplines, classification, and applications of agent-based systems. Both DPS and MAS were introduced, particularly the latter and a number of applications were examined. Section 3.4 explored the basic principles of agent negotiation and various agent negotiation mechanisms. Section 3.5 analysed the key components of agent learning; discussed the major agent learning approaches; and pointed out the problems and challenges for developing learning approaches. Finally, this chapter discussed the potential benefits, application methods, implementation areas, and possible problems for the application of MAS in construction.

Some of the main points made in this chapter include the following:

- In general, MAS represent a melting pot of ideas orienting from such areas as distributed computing, object-oriented systems, software engineering, artificial
intelligence, economics, sociology, and organisational science. At its core is the concept of autonomous agents interacting with one another for their individual and/or collective good. This basic conceptual framework offers a natural and powerful means of analysing, modelling, and resolving many real world problems.

- Negotiation, the most important agent collaboration mechanism, plays a central role in MAS. The development of various agent negotiation mechanisms and their applications in different environments reveal the potential and possibility for the MASCOT system to facilitate construction claims negotiation. The study of negotiation mechanisms provided valuable experience for the development of MASCOT negotiation protocol and strategies, and pointed out the direction in which the MASCOT system should focus.

- An agent's learning ability increases its negotiation power in a MAS. The analysis of the components and processes of agent learning suggested what and how an agent would learn from other agents or the environments. The philosophical differences between different learning methods suggested which method was more suitable for a particular system, and which had more potential to be applied in the MASCOT system.

- The study of the research projects conducted on the applications of MAS in the construction industry to facilitate engineering designs and negotiations indicated how MAS have been framed to solve different industrial problems. This is particularly important for the development of the MASCOT system in the construction industry environment.

The next two chapters will address the above three aspects of the MASCOT system (i.e. the industrial background, agent negotiation, and learning). Chapter 4 first explores the major negotiation theories on which the MASCOT negotiation mechanism is based. Chapter 5 identifies the reasoning model of the MASCOT system by further analysing the nature and major characteristics of claims negotiation; selects the theoretical negotiation model; and develops the MASCOT negotiation mechanism (e.g. the negotiation protocol and strategies), where a particular learning approach – Bayesian learning approach is adopted.
CHAPTER 4
NEGOTIATION THEORIES

4.1 INTRODUCTION

Negotiation, as an important human co-operation approach, has been studied and defined by many researchers in different research domains, such as: economics, society, politics and AI systems. For example,

- Collins Cobuild English Dictionary: Negotiations are formal discussions between people who have different aims or intentions, especially in business or politics, during which they try to reach an agreement.

- Zartman (1977): Negotiation is a joint decision process between two or several parties or their representatives. Negotiation tends to be a matter of finding a formula encompassing the optimum combination of interests of both parties and then of working out the details that implement these principles and affect the agreement. Negotiation is a dynamic process, on-going, involving moves and countermoves.

- Hammer and Clay (1977): Negotiation is the interaction that occurs when two or more persons attempt to agree on a mutually acceptable outcome in a situation where their orders of preference for possible outcomes are negatively correlated.

- Gulliver (1979): Negotiation is one kind of problem-solving process, in which people attempt to reach a joint decision on matters of common concern in situations where they are in disagreement and conflict.

- Rosenschein and Zlotkin (1994): Negotiation is a form of decision-making process where two or more parties jointly search a space of possible solutions with the goal of reaching a consensus (deal).
• Lesser (1998): *Negotiation, the process of arriving at a state that is mutually agreeable to a set of agents, is intimately related to co-ordination.*

These definitions provide different views about negotiation objective, nature, scope, and elements involved. For example,

• From a social-psychological perspective, Bartos (1977) concludes that the nature of negotiation is to resolve the conflicts between competitive individualism and co-operative collectivism. Negotiation often involves dual and mostly conflicting motivations: the individual (competitive) desire to maximise one’s own utility and the collective (co-operative) desire to reach a fair solution. Negotiations can proceed smoothly only when they are guided by the collective desire for fairness, or when the loss of breaking negotiation is higher than that of reaching an agreement for either party even if negotiators are selfishly motivated.

• In DAI, as defined by Lesser (1998), negotiation could be a co-ordination approach for participants to gain more. There is not necessary any conflict between the participants. This is so-called co-operate negotiation.

In most cases, negotiation participants have power over each other during this process (Young, 1991). There are relatively fixed parties and flexible values in negotiation. Value and behaviour are modified to alter divergent positions toward a common convergence of values (Spector, 1977). Negotiation normally includes three stages:

• Negotiation starts from the point where each party tries to maximise his own payoff;

• By exchanging information, two parties explore the nature and extent of their differences and the possibilities open to them, and seek to induce or persuade each

---

4 The following study focuses more on this kind of negotiation. Such negotiation is also commonly termed as bargaining.

5 At this point, the focus is on the changing value of a negotiator during the negotiation. However, the participants of a negotiation could be changed (add or reduce) during the negotiation process. This is true particularly when negotiation is conducted between agents in multi-agent systems.
other to modify their expectations and requirements, and then search for an outcome that is at least satisfactory enough to both parties (Gulliver, 1979). Different strategies are adopted by negotiation participants, such as: concession making, contending, problem solving, inaction, withdrawal (Pruitt and Carnevale, 1993).

- Finally, an agreement is reached, theoretically, at an equilibrium point where the opposing interests are balanced. Practically, the final results are inevitably influenced by many factors such as: the negotiators’ personal capabilities, negotiation strategies, time constraints, expectations, and the relationship between the parties. Therefore, the outcome of a negotiation could be the victory of one party, a simple compromise, win-win result, or a conflict.

As an important social activity, negotiation involves more complex human interactions than simple technical issues. Many important theories and principles have been developed to explain various aspects of negotiation. Pruitt and Carnevale (1993) summarise three main traditions in the study of negotiation:

- The first consists of books and manuals providing advice on practical negotiation issues (e.g. Fisher and Patton, 1982; Cohen, 1991; Warham, 1993; Fowler, 1996).

- The second consists of mathematical models of rational behaviour developed by economists and game theorists (e.g. Luce and Raiffa, 1957; Young 1975). These models are ordinarily limited to a relatively narrow set of tactics, such as concession making or third-party recommendations for particular agreements. They focus on the various situations, assumptions, outcomes, and bargaining processes of negotiation. The two kinds of model are sometimes combined by theorists who use the tools of rational analysis to examine the wide range of tactics used by most negotiators and third parties (e.g. Schelling, 1960; Raiffa, 1982).

Rojot (1991), accepted by most of the negotiation theorists, terms the theories developed based on these approaches as Mechanical Theory where the distinctions
between the classical game theoretical model and the economic theoretical model are identified.

- The third, behavioural, tradition seeks to develop and test predictive and practical theory about the impact of environmental conditions on negotiator (or mediator) behaviour, human behaviours during negotiation, and the impact of these conditions and behaviours on outcomes (e.g. Walton and Mckersie, 1965; Rubin and Brown, 1975; Morley and Stephenson, 1977). The theories developed based on this approach are classified as behaviour theory.

With the aim of developing a MAS negotiation system, this study focuses on the second approach since it provides theoretical foundations for computer-based negotiation systems. Meanwhile the behaviour theoretical approach is also taken into consideration, as the concepts of behaviour models are crucial for computer negotiation systems to simulate practical negotiations.

- Section 2 first reviews classical game theory and its applications in negotiation, discusses economic theory, and compares the advantages and disadvantages of these two approaches in analysing negotiation.
- Section 3 discusses behaviour theory and a few important behaviour theoretical models.

4.2 MECHANICAL THEORY

Mechanical Theory includes two closely related approaches: the classical game theoretical approach and the economic theory approach. These are discussed below:

---

6 Some MAS researchers view the theories developed based on this approach as Game Theory, where the economic approach is treated as a branch of the game theoretical negotiation approach termed as strategic bargaining approach, whilst the classical game theoretical approach is termed as axiom bargaining approach. The problem in this classification is that a confusion is easily generated between the strategic (normal) form of game theory and the strategic bargaining approach (for details, see Osborne and Rubinstein, 1994; Rosenschein and Zlotkin, 1994; Sandholm, 1996; Kraus, 1997). To avoid such a confusion, the term classical game theoretical approach is used to distinguish the general game theoretical approach.
4.2.1 The Classical Game Theoretical Approach

Game theory seeks to get at the essentials of decision-making and the associated strategies in situations where two or more parties are interdependent, and where, therefore, the outcome of their conflict and competition must be the product of their joint requirements and the interaction of their separate choices (Bacharach and Lawler, 1981). These parties are in a situation in which there may be many possible outcomes with different values to them. Although they may have some control which will influence the outcome, they do not have complete control over others. Each party in a game faces a cross-optimisation problem.

Game theory represents the most thorough search for a determinate solution to negotiation problems (Bacharach and Lawler, 1981). All types of negotiation can be conceptualised as different kinds of games (Brams, 1990). It focuses on the logical and hypothetical conceptualisation of problems and processes wherein many of the variables of the real negotiation are ignored. For example, all game theoretical models are based on some important simplifying assumptions which determine how the theory works and its scope. According to Nash (1950), the fundamental assumptions are:

- both the number of players and their identity are assumed to be fixed and known to everyone;
- players are rational and expect others to be rational;
- players attempt to maximise their own gain or utility;
- players have complete information on the utility of alternative settlements to themselves and their opponents. The payoff function for each player is fixed and known at the outset. The utility function of each player is fixed and known.

Based on these assumptions, game theory places negotiation in the context of a general theory of individual choice. Given the mutual dependence relationship in decision-making, such choice becomes a strategic issue; that is, a party must assess how to
maximise his own gain in the context of potential interference from others (Gulliver, 1979). The game theoretical models are essentially static models which attempt to deduce what strategy a player should take, taking account of the fact that an outcome is the result of the interdependent strategies and choices of two or more players. Therefore, the major objective of the game theory approach is to offer a set of rules that describe how rational actors choose the best strategy most consistent with the bargainers’ conflicting interests (Shubik, 1975).

The basic elements of game theory are players, payoffs and rules of action. First, there must be a well-defined set of courses of action for each of a number of players. Second, there must be well-defined preferences for each player among possible outcomes of the game, or mixtures of his outcomes. Third, the relationships must exist whereby the outcome is determined by the player's courses of action (Bacharach and Lawler, 1981).

A game can be categorised as zero sum or non-zero sum, depending on whether there is a fixed set of available payoffs. In zero sum games, players try to garner as much of this fixed pot of rewards as possible. Every player assumes that his opponent will “do his worst” (Von Neumann-Morgenstern, 1947). The key point in this analysis is the proposition that a rational player in a two-person, zero-sum game can predict the preferences of his opponent accurately. There is no strategic interaction in his decision-making problem7 (Young, 1975).

In a non-zero sum game, players look to find opportunities such that mutual gains can be made, thereby increasing the size of the pot. In this situation, there are strategic interactions between players. To eliminate strategic interaction from non-zero sum situations, game theory is divided into a number of separate domains distinguished by the specific assumptions they employ:

---

7 Strategic interaction is simply the set of behaviour patterns manifested by individuals whose choices are interdependent in negotiation (Young, 1975).
Nash (1950) has sought to predict outcomes in non-zero sum interactions by searching for solutions that seem particularly stable - equilibrium points. Nash shows that under some rational behaviour and symmetry assumption, players would reach an agreement on a deal that will be individual rational, Pareto-optimal, and will maximise the product of the players' utility.

Nash (1953) also sought to circumvent the "outguessing" regress by introducing fixed decision rules, whose application yielded unique prediction concerning the distribution of payoffs between the players. These decision rules were based on specific criteria concerning the worth or power of the individual players and/or on certain desirable attributes of a solution. Games of this kind are known as co-operative games.

Harsanyi (1977) has sought to achieve determinate solutions for non-zero sum games by introducing the notion that each player may be able to assign "subjective probabilities" to the choices of the other players.

Unlike the competitive games, these models not only use the idea of a solution concept where the agents' strategies form some type of equilibrium. Instead, desirable properties for a solution, called axioms of the bargaining solution, are postulated, and then the solution concept that satisfies these axioms is sought (Osborne and Rubinstein, 1990; 1994).

For example, in the Nash bargaining solution (1950), it is assumed that the space of feasible utility vectors is compact and convex. When many deals are individually rational (i.e. pay more than the fallback) to both agents, there may be many Nash equilibrium solutions. An example is that the agents are bargaining over how to split a dollar, all splits that gives each agent more than zero are in equilibrium. If Agent 1's strategy is to offer $P$ and no more, Agent 2's best response is to take the offer as opposed to the fallback which is zero. Now, 1's best response to this is to offer $P$ and no more. Thus, a Nash equilibrium exists for any $P$ that defines a contract that is individually rational for both agents, and feasible ($0<P<1$). Due to the non-uniqueness of the equilibrium, a
stronger (axiomatic) solution concept such as the Nash bargaining solution is needed to prescribe a unique solution.

The first axiom is based on the view that the agents' numeric utility functions really only represent ordinal preferences among outcomes - the actual cardinality of the utilities do not matter. Therefore, the utility function can be transformed, and the resulting game should be equivalent to the original game. The second axiom requires symmetry: if the agents have symmetric bargaining positions, their outcome utilities should be equal. Third, independence of irrelevant is required. The fourth axiom requires Pareto efficiency. It turns out that there is a unique solution that satisfies the four axioms. This Nash bargaining solution selects the utility pair that maximises the product of the players' gains in utility over their fallback utility.

Game theory provides a useful benchmark and a fundamentally important methodological approach to the study of situations involving potential conflict. It provides a device to isolate key factors in negotiation with facility (Shubik, 1975). For example, some of the most important games such as 'the prisoner's dilemma' and 'the game of chicken' provide deep insight to negotiation. Rapoport (1983) conclude the strength of classical game theory as: "The mathematician's elucidation of problems in game theory sometimes leads not to a solution but to a clarification, namely of what it is that the problem involves, what obstacles stand in the way of solutions, what special cases of the problem can be treated by what methods".

Although classical game theory is expected to provide a scientific framework that would allow the prediction of outcomes as well as the explanation of a negotiator's behaviour and decisions in real-life negotiation, there does not appear to be any convincing formulation that offers reliable explanation or prediction (Gulliver, 1979). The assumptions of the theory and its highly abstract mathematical solutions raise general problems in regard to their potential validity and applicability (Zartman, 1977):

- the classical game theory formula presents a static representation of what is essentially a dynamic process in which components and their inter-relationships are
intrinsically subject to change, thus affecting their contribution to and the nature of the outcome;

- the classical game theory formula includes components that are not measurable nor ascertainable even in broadly acceptable approximations\(^8\); and

- the classical game theory is unsuitable for the analysis of the negotiation process because its assumptions identify and remove all the obstacles that bargainers have to confront.

4.2.2 The Economic Theoretical Approach

The economic theoretical approach, in contrast to the classical game theoretical approach, seeks to develop dynamic models of process, involving offers and counteroffers and interdependent concession making. There is no concern for the discovery of once-for-all strategies but rather an intention to examine how the bargainers should interact in terms of their expectations of each other. According to Young (1975), this approach is principally under conditions of bilateral monopoly which seeks to explain a jointly determined outcome in terms of the rational tendencies of the parties to reach an optimal point of intersection on their lists of interchangeable preferences.

Economic theoretical approaches are fundamentally convergence models. They analyse the processes through which the demands of the participants converge over time toward some specific point on the contract curve. Therefore, the key element of these models is the development of a specific concession mechanism that permits the positions of the parties to converge in the course of a series of offers and counteroffers (Bacharach and Lawler, 1981).

The best known economic theoretical approach is Zeuthen's economic welfare under condition of bilateral monopoly. In this model, the two players' bargaining problem is

\(^8\) This is also the common limitation faced by the mechanical theory (i.e. the adoption of the utility functions as the criterion to evaluate a participant's payoffs and determine his negotiation strategies).
considered as a one-player decision process under the assumption that if none of the players concedes at a particular step, they will reach a conflict.

- The individual player compares the certain value he can obtain by accepting the other side's offer. Based on this offer, his own favoured outcome and the results of conflict, he calculates the maximum probability of conflict he would be willing to accept in preference to acquiescing on the current offer of the other side.

- Concession will be made by the side willing to accept the smaller risk of conflict at any given moment in time. A player needs only reduce his own demand to the point where he is willing to accept a greater risk of conflict than the other (Young, 1975). Accordingly, each player must continue to concede until he is willing to accept a larger risk of conflict than his opponent.

Cross's model (1975) emphasises the role of time as an important factor in bargaining, and conceptualises the process of making concession in terms of the adjustment of expectations through learning. In this model, an individual bargainer starts his calculations with his preference ordering for the outcomes in the payoff possibility set; a schedule of costs arising from the time that elapses before a specific contract is agree upon; and a precise estimate of the other side's concession rate over time. Each bargainer proceeds to calculate the optimal level for his own initial demand on the assumption that the other player will make all of the concessions by taking into account the trade-off between improvements in the final settlement associated with higher initial demands and the increased costs which higher demands produce as they extend the time required to reach a specific agreement.

Based on these economic models, some bargaining alternative offering approaches in modern game theory have been discussed (e.g. Rasmusen, 1994; Osborne and Rubinstein, 1990, 1994; Kreps, 1990). An example is that one can again think about deciding how to split a dollar. With a finite number of offers and no time discount, the last offerer will get it all, and the other agent is indifferent between accepting a zero offer or getting a zero fallback payoff. With time discount, a finite game can be solved starting from the end.
The agent that is to make the first and the last offer gets a payoff that approaches \( \frac{1}{1+Q} \) as the number of negotiation rounds approaches infinity. The term \( Q \) is the discount factor. When protocol in a non-discounted setting allows an infinite number of bargaining rounds, the solution concept is powerless because any split of the dollar can be supported in sub-game perfect Nash equilibrium. On the other hand, in a discounted infinite round setting, the sub-game perfect Nash equilibrium is unique. Specifically, the first offerer gets \( \frac{1-Q^2}{1-Q_1Q_2} \), where the first offerer’s discount factor is \( Q_1 \), and the other player’s \( Q_2 \) (Rasmusen, 1989).

Another model of sequential bargaining does not use discounts, but assumes that there is a fixed bargaining cost per negotiation round. If the agents have symmetric bargaining costs, the solution concept is again powerless because any split of the dollar can be supported in sub-game perfect Nash equilibrium. On the other hand, if the first offerer’s bargaining cost is smaller than the other agent’s, the first agent gets the entire dollar. If the first agent’s bargaining cost is greater than the second’s, the first agent receives a payoff that equals the second’s bargaining cost. The second agent receives one minus this. Agreement is reached on the first round.

The economic theoretical models differ from the models of the classical game theoretical approach in several ways:

- First, the economic theoretical models deal only with certain non-zero sum or mixed-motive situations. Thus, they focus on interactions in which there is a distinct range of possible outcomes within which each of the participants would prefer to reach an agreement than to accept an outcome of non-agreement, even though they have conflicting interests concerning the precise terms of agreement. Situations involving pure conflict and pure co-operation are outside the scope of these models.

- Second, economic theoretical models treat negotiation as a process of convergence over time involving a sequence of offers and counter-offers. Consequently, the economic theoretical models are dynamic models which focus on the bargaining process as well as on the ultimate outcome of bargaining, whereas the game theoretic
models are predominantly static models which concentrate on the ultimate distribution of payoffs among the parties.

- Third, the economic theoretical models tend to emphasise the formation of expectations about the behaviour of the relevant other(s) in contrast to the models of game theory which stress either conditions that allow each player to make accurate predictions concerning the behaviour of the relevant other(s) or characteristics of a permissible solution which are sufficient to yield determinate outcomes (Young, 1975).

4.3 BEHAVIOUR THEORY

The applications of classical game theory and economic theory are limited by their assumptions which neglect the complex human factors in negotiation. On the contrary, behaviour theory attempts to analyse the negotiation processes in which negotiators influence each other's expectations, perceptions, assessments, and decisions during the search for an outcome, thereby affecting the outcome. Much attention is given to the nature of changing expectations and negotiator's tactics, and to the significance of uncertainties of information, perception, and evaluation - all matters that tend to be ignored by the mechanical theory. All this involves a closer approximation to the real world (Zartman, 1977). The major Behaviour models are discussed below:

4.3.1 Psychological Model

This approach focuses on analysing the personality or psychological responses of the decision-makers themselves rather than the negotiation process. It seeks to explain bargaining effectiveness in terms of variables such as the behavioural characteristics of the negotiators and their perceived and actual use of interpersonal strategies. It addresses the extent to which personality, perception, expectation, persuasion, and interaction of these factors within negotiator dyads can adequately describe and explain the process and outcomes of bargaining (Spector, 1977). Four factors are analysed in this approach:
- **Negotiator personality** identifies basic predisposition toward the opponent and motives for future actions and responses. It shapes one's perspective and expectations of particular objects and goals.

- **Perceptions and expectations** of the opponent's strengths, weaknesses, intentions, commitments, and goals are likely to affect negotiation responses, the tone of interpersonal communication, and the learning process.

- The use of **persuasive techniques** and their success in modifying negotiator values toward initially desired end-states should help to achieve acceptable outcomes.

- The **interaction of psychological and contextual factors** address the possibility that certain personalities become instrumental in motivating the bargaining process only under particular negotiation conditions.

The advantages of the psychological approach lie in the fact that it focuses on the fixed element of the process - the parties - and their ability or propensity to modify the variable element - the values at stake. It deals with realistic aspects of negotiation using concepts that are possible to operationalise. In many practical negotiations, psychological approaches play a very important role in the success of the negotiations. The drawback of this approach is that the analysis of the agent rather than the process. It focuses on the secondary rather than the primary element of decision-making.

### 4.3.2 The Learning Model

This approach views negotiation as a learning process in which each party is largely dependent on his experience of the results of past actions by the two parties. What has occurred previously is used as a standard of assessment by which to choose what to aim for and what to do next (Gulliver, 1979). The basis of this model is that negotiators' strategies are changeable. Negotiation strategies are contingent, contain errors, expectations will change, and this will lead in turn to a modification of each party's choice of strategy.
Negotiators' behaviours are classified as: actual payoff demands and manipulative moves such as threats or coercive actions. The latter are not directly related to the payoff, but they do affect the overall value of the negotiation because they may influence the settlement point and costs of negotiation. The learning process involves identifying the other party's real payoff demands and manipulative moves to decide the negotiator's own strategy.

The learning mechanism establishes a dynamic interaction between the two parties' behaviours. Supposing $S_a$ is Party A's strategy, $R_a$ is Party B's estimate of Party A's strategy under an uncertainty $V_b$. $S_a$ determines the current course of $r_a$, which is used by Party B in the formation of $R_a$. In response to $R_a$, Party B selects strategy $S_b$ which determines the course of $r_b$, and this in turn is the basis for Party A's estimate $R_b$. As a consequence of learning, $S_a$ and $R_b$ may be continually changing, so that what is learned in the form of $R_a$ or $R_b$ is a composite of a sequence of strategies rather than a single one (Cross, 1977). Such a learning process is expressed in Figure 4.1.

The learning model plays an important role in current negotiation theory since it reflects and simulates one of the most important characteristics of practical negotiations - the inference process of negotiators. For example, its importance has been recognised in the game research community as fundamental for understanding human behaviour as well as for developing new solution concepts (Cross, 1977; Osborne & Rubinstein 1994; Jordan, 1992). Various learning mechanisms, such as: Q-learning, collective learning and Bayesian learning have been developed for intelligent agents.
4.3.3 The Dual Responsiveness Model

Unlike the learning model, the dual responsive approach shows that negotiator’s responsiveness can be based on both his own previous concessions and the opponent’s concessions. A negotiator’s response is a function of his own previous pattern of concession making as well as the opponent’s concession rate (Figure 4.2). His responses are mediated by expectations which are adjusted through the course of the conference. This approach suggests two types of monitoring functions in negotiation: each negotiator monitors the other side for evidence of movement and monitors his own side for evidence of preferences. Unlike the learning approach, this approach calls attention to the importance of an internal dynamic in bargaining while both emphasise the importance of mutual responsiveness. Responsiveness is likely to occur in both directions.

![Figure 4.2 The dual responsive model](image)

4.3.4 The Joint Decision Making Model

Zartman (1977) points out that the above models do not correspond to the conceptual characteristics that negotiation is a mode of the decision-making process, and do not deal with the process as it is actually practised. He concludes that negotiation tends to be a matter of finding a formula encompassing the optimum combination of interests of both parties and then of working out the details that implement these principles and affect the agreement.
Negotiators first seek a general definition of the items under discussion, conceived and grouped in such a way as to be susceptible to joint agreement under a common notion of justice. Once agreement on a formula is achieved, it is possible to turn to the specifics of items and to exchange proposals, concessions, and agreements. Details are resolved most frequently in terms of the referents which justify them and give them value rather than in their own intrinsic values. This means that convergence does not take place by inching away from fixed positions toward a middle, but rather by establishing a referent principle from which the value of the detailed item will be derived (Zartman, 1977).

Different from all the others, the joint decision making model views negotiation at a macro level, which provides some special advantages.

- First, it is possible to prescribe a negotiation through the achieved formula which can be made in impractical or artificial terms;

- Second, it leaves room for analysis of power as added value while others are not able to do; and

- Third, it can be used in conjunction with the other models.

4.4 SUMMARY

This chapter first defined negotiation and explored the nature and major elements of negotiation. Then, it discussed the two major negotiation theories: mechanical theory and behaviour theory. Section 2 reviewed two major theoretical approaches of mechanical theory: classical game theory and economic theory. These approaches are mainly mathematical models based on the rational behaviour assumption. The former (e.g. Nash solutions, 1950, 1953) is mainly concerned with the predictions of outcomes under certain assumptions about the players and the outcomes themselves. The latter (e.g. Zeuthen’s model) focuses on analysing the negotiation process given the rational assumption.
Although these two theoretical approaches have been criticised as over simplistic; relying too much on the assumptions; and ignoring the social norms, relationships between negotiators, and group decision processes; these approaches uncover many ambiguous problems in negotiation. Furthermore, these approaches (especially the economic theoretical approach) provide important theoretical models for the development of the computer negotiation systems.

Section 3 discussed behaviour theory and several important theoretical models including psychological model, learning model, dual-response model, and joint decision model. Unlike mechanical theory, behaviour theoretical models aim to address the complex human responses in negotiation, such as how a negotiator analyses the opponent's expectations and psychological changes during negotiations, and the corresponding strategies that the negotiator will take to deal with such changes. Although behaviour models are normally difficult to simulate in computer-based negotiation systems, they represent some important concepts in practical negotiations, which developers of MAS negotiation systems need to address in their systems.

The next chapter will discuss the MASCOT system development process based on the previous studies on claims negotiation, multi-agent systems, and negotiation theories. It will also address the selection of the most appropriate negotiation theory for the MASCOT negotiation mechanism. Furthermore, the learning model will be considered to tackle some of the complex human issues in construction claims negotiation.
CHAPTER 5
DEVELOPMENT OF THE MASCOT CONCEPTUAL MODEL

5.1 INTRODUCTION

The previous chapters have addressed three major issues for the development of the MASCOT model: construction claims negotiation, multi-agent systems and negotiation theories. This chapter builds on these in the development of the MASCOT conceptual model. It first analyses some essential characteristics of construction claims negotiation and explores the selection of an appropriate negotiation theory for the MASCOT model. The MASCOT conceptual model is then presented and its key features outlined.

Parinak (1996) identifies three essential aspects for the design of a MAS negotiation system, i.e.

- what are the issues over which negotiation takes place?
- what reasoning model will the agents employ? and
- what negotiation protocol will be used?

Besides these issues, this study also focuses on the essential negotiation theoretical models on which the MASCOT model is based. As a whole, this chapter conducts three major studies for the development of the MASCOT model, which include:

- Identifying the nature and essential characteristics of construction claims negotiation. Based on the identified nature, an essential conceptual negotiation model (i.e. the reasoning model) for the MASCOT system is developed. The identified essential characteristics are also crucial for the development of the MASCOT negotiation protocol and strategies. These issues are addressed in Section 5.2;
Selecting an appropriate theoretical negotiation model on which the MASCOT negotiation mechanism will be based, given the nature of construction claims negotiation and the characteristics of various negotiation theories. This issue is discussed in Section 5.3;

Defining the key assumptions and requirements of the MASCOT system, developing the negotiation protocol and various negotiation strategies, and developing the process model of the system. These issues are presented in Section 5.4.

5.2 GENERAL CHARACTERISTICS OF CLAIMS NEGOTIATION

Based on the previous analysis of construction claims negotiation, this section makes a further analysis on the nature and characteristics of claims negotiation in terms of the application of multi-agent systems. Such an analysis is of paramount importance for the development of the MASCOT model. Some of the important characteristics are discussed below.

5.2.1 Contractually Obliged Self-Interested Relationship

Essentially, construction claims negotiation is different from other business negotiations in several essential ways; for example:

- In common business negotiation, one party may simply draw back from the negotiation if the party finds that the negotiation has fallen into a deadlock, or it cannot get the expected profits from the negotiation. In construction claims negotiation, nobody can easily walk away from the negotiation unless it is ready for arbitration or litigation which they can rarely afford to undergo. As a result, both sides need to have a certain flexibility and be able to work out solutions to avoid a conflict result (i.e. a negotiation ends without an agreement; the claim thus falls into a dispute), and to keep the project going. Efficient negotiations are therefore essential for construction claims management.
• Furthermore, since the participants in construction claims negotiation are legally obliged by a project contract, the negotiations are conducted within the framework of the contract. Contract documents set promises for the claims negotiation. Factors such as the causes of claims and compensations are principally addressed in the contract documents. Claim negotiations are conducted within these boundaries.

• On the other hand, construction project teams are temporary organisations. Each participant belongs to a different organisation and aims to get maximum benefits. Therefore, each participant will try to maximise its own benefit as long as it does not break the co-operative relationship in the project. Such a self-interested competition relationship is more evident in claims management and negotiation than in other construction activities since claims represent a major approach to ensuring reasonable profit sharing between the client and the contractor during the construction process.

The above can be summarised as a contractually obliged self-interested competitive relationship. It represents the basic framework on which the MASCOT system is based.

5.2.2 Participant-Dependent Information

Due to the different roles, project participants normally have different perspectives on a project. For example, the client may know well the final functional requirements, budget and the financial status of the project while the engineer understands well the client's requirements, contract documents as well as the contractor's financial situation, progress and quality of the work. On the other hand, the contractor often has more detailed information about what really happened on site (e.g. schedule, progress, the circumstances that led to a claim, and the actual cost of change orders). Besides such distributed information, each party also has different expertise. The client might know little or nothing about the construction; the engineer (or architect) should be an expert in design and project management; the contractor may be a specialist in structural work, with mechanical & electrical (M&E) knowledge from its subcontractors. Since
construction claims negotiation is evidence-oriented, information plays a critical role. Thus, each party will try to use his specific information and expertise to explain, argue and persuade the other party to accept its offer.

More importantly, each party has its own interest, expectation and estimate about a certain claim, which will determine its utility function, negotiation strategies and preferred outcome. All this information is unknown to the other parties at the beginning of negotiation. During the negotiation, each party will seek to obtain such information by observing the opponent's offers, counteroffers and supporting arguments.

5.2.3 Strategy-Influenced Process

Given the contractually bounded self-interested nature and the participant-dependent information barriers, resorting to a predefined strategy is a common occurrence. This means that a number of strategies are deployed in an attempt to move the final settlement position from the middle towards each party's own end of the opening extremes. Incomplete information and different strategies will influence the payoffs of negotiation (Kennan and Wilson, 1992; Bacharach and Lawler, 1981). For example, the contractor may exaggerate his initial claims amount in order to make the engineer's compromise point near to his expected outcome. If there is not enough information to identify the exaggeration, the contractor's strategy may not be perceived by the engineer. In practice, depending on different situations, either or both parties may (Pickavance, 1997):

- inflate the opening demands;
- misrepresent their position or interests;
- withhold sensitive or potentially damaging information;
- use threatening behaviour;
- adopt an intransigent stance until the other side is ready to move;
- secure concessions before giving concessions;
- concede little and slowly;
- make a tit-for-tat counter proposal; or
• re-open negotiation on agreed settlements as a result of “new” information.

Most of these complex strategies represent the highly psychological aspects of human negotiations, and are difficult to be simulated by computer systems. This, in turn, requires careful consideration in the MASCOT negotiation strategies to clearly address the key aspects of negotiators’ decision-making criteria.

5.2.4 Time - An Important Factor

The claims negotiation process is time-consuming. Before negotiation, documents have to be specially presented; negotiators need to be gathered; and there is often a substantial delay between the submission of documents and negotiation meetings. During negotiation, neither side wants to make a concession first or easily. A party may adopt a time-consuming strategy to expect to benefit from the opponent’s time pressure or emotional exhaustion. Unrelated issues (such as site management, quality or safety) are often brought up to pressurise the opponent. Moreover, most claims cannot be settled in one meeting. There are usually a series of meetings, either formal or informal, to allow both parties to make offers and counteroffers.

As a result, many standard contracts, such as FIDIC require the contractor to complete the revised work first, regardless of claims for extra payment, disputes, and references to arbitration. Negotiations for compensation are conducted long after the work or even the whole project has been completed to avoid the potential delays caused by long arguments. However, having done the work, the contractor has little negotiating strength and becomes a supplicant. This often results in disputes and litigation. On the other hand, if compensations are negotiated before the work starts, the client is under the risk of delay, especially when the revised work on a critical path. Thus, the time issue will be an essential factor in the design of MASCOT to ensure that a negotiation outcome can be reached within an expected time scope.
5.2.5 Tradeoffs between Negotiation Items

Construction claims are ongoing processes in a construction project; thus, negotiations for these claims are also intensive. Claim/negotiation items can be quite different from their nature, amount and importance. Different claim items often influence each other. Even for a single claim item, negotiations could be about the money compensation, time extension, or technical issues. These aspects are often linked with, and have impacts on one another. Tradeoffs between different aspects of a claim, or different claims are common in practical claims negotiation. From case to case, such tradeoffs are important for resolving claims.

5.3 NEGOTIATION MODEL IN MASCOT

The selection of negotiation models is the basis for designing the MASCOT negotiation mechanism, which should consider the assumptions, functions and limitations of each negotiation theory as well as the important characteristics of construction claim negotiation discussed above. Based on the previous studies, the following sections discuss the possibility of the application of the classical game theory approach, economic theory approach and behaviour theory approach in the MASCOT model.

5.3.1 Game and Economic Theory Models

Both the game theoretical approach and the economic (bargaining) theoretical approach analyses the technical aspect of negotiation, constructs formal models of negotiation environments, and provides clear analyses of various situations and precise results concerning the strategy a negotiator should choose. Therefore, they provide general theoretical bases for automated negotiations.

Classic game theoretical models, such as Nash (1950) and Von Neumann and Morgenstern (1964) aim to sort out solutions of game through proper assumptions and mathematical equations. Agents' negotiating manoeuvres are expressed using the Nash
equilibrium concept without specifying the negotiation process. Since this study intends to build automated negotiators, such classic game theoretical models are not suitable for the construction of the MASCOT mechanism (However, the concept and the assumptions of game theory approach will still be adopted, see Chapter 4).

In contrast, economic theoretical models, such as Zeuthen (1975) and Cross (1975) provide a useful theoretical foundation for designing the MASCOT model. Unlike the classic game theory, economic theory focuses on analysing the strategic process to reach a final equilibrium point. Genesereth et al. (1986), Rosenschein and Zlotkin (1994) and Kraus (1995, 1997) have developed their multi-agent negotiation systems based on such strategic economic models. Moreover, construction claims negotiation, in most cases, is characterised as bargaining between the contractor and the engineer with the support of relevant evidence. Thus, the application of the economic (bargaining) theory as a theoretical basis for the MASCOT model matches the nature of MAS and claims negotiation.

Although the economic theoretical approach provides a general framework for modelling agent claim negotiations, several important issues need to be addressed for the applications. Kraus (1997) summaries these as:

1) Choosing a strategic economic theoretical approach which is applicable for the specific MAS negotiation problem; and matching the MAS scenarios with the economic-theoretical definitions of the chosen model

Considering the nature and characteristics of construction claims negotiation, this study adopts the Zeuthen's model as the core negotiation models. Zeuthen's risk evaluation model specifically addresses the participants' risk perception in conflict avoidance, which matches the obliged self-interested nature of construction claims negotiation. Moreover, Harsanyi (1977) has demonstrated that Zeuthen's solution is identical to Nash's method (1950); that is, parties will settle at the point that maximises the product of the difference between what bargainers get from conflict and what each gets from the settlement point.
Meanwhile, the time accounting and learning concepts expressed in Cross's model are also very important to make Zeuthen's model work in the claims negotiation environment. The former addresses the important time issues in claims negotiation and helps to keep the negotiation stable, whilst the latter overcomes the complete information assumption on which Zeuthen's model is based. Aspects of the behaviour-oriented approach are incorporated by the provision of a learning mechanism for the MASCOT agents.

2) Identifying equilibrium strategies

Given specific assumptions about different application environments, different concepts of equilibrium and strategies that are in equilibrium have been identified, such as Nash Equilibrium\(^9\) (Nash, 1950), Perfect Equilibrium\(^10\) (Selten, 1975), Bayesian-Nash Equilibrium\(^11\) (Harsanyi, 1956) and Sequential Equilibrium\(^12\) (Kraus, 1996). If agents adopt these equilibrium strategies, the interaction among these agents is efficient and stable within a certain scope. In order to address equilibrium in the MASCOT model, this study formalises several assumptions that are appropriate for the construction claims negotiation environment. For example, all agents sustain a loss over time; agents can get information about their opponents through a learning approach; there is a finite set of agreements, and agents prefer to reach an agreement which may not bring them maximum benefits, rather than opting out of the negotiations. These assumptions are different from the assumptions in common economic theory or game theory. A detailed analysis of equilibrium strategies under these assumptions is discussed later.

---

\(^9\) See Section 5.4.2.

\(^10\) It can be said that a set of strategies is in perfect equilibrium if the agent's strategies induce an equilibrium at any stage of the interaction (Selten, 1975). Perfect equilibrium is stronger than Nash equilibrium. In each stage of the negotiation, assuming that an agent follows the perfect equilibrium strategy, the other agent has no strategy better than to follow its own perfect equilibrium strategy.

\(^11\) See Section 5.4.2.

\(^12\) When the number of agents is \(n\) and the number of possible types is \(k\) then a sequential equilibrium is a sequence of \(nk\) strategies and a system of belief with the following properties: each agent has a belief about its opponent's type. At each negotiation step, any agent's strategy will be optimal given its opponent's possible strategies, the history up to the given time periods, and the agent's current beliefs (Kraus, 1996).
3) Developing low complexity techniques for searching for appropriate strategies

Development of the negotiation strategies which are in equilibrium can rely on theorems proven in advance, but this can be done only when the set of possible agreements can be defined. In such situations, there is a need to develop relatively simple computational techniques for appropriate strategies adopted by the automated negotiators. This study adopts Zeuthen's bargaining model and Bayesian learning approach. Both approaches are based on clear concepts with simple computation methods. Also, the other MASCOT negotiation strategies such as expanded solution-searching (see Section 5.4.4) are also computationally simple.

4) Providing utility functions

Both economic theory and game theory assume that each player knows exactly the opponent's utility functions. In most cases, the designer of an automated agent is required to provide the agents with a utility function. Without doing so, it is often difficult to apply these theories to automated agents. Moreover, since the MASCOT model works in a worth-oriented domain, utility functions are built to represent agents' worth functions (see Section 5.4.1). In the MASCOT model, agents' utility functions are represented as linear functions, which are determined by two factors: reservation value and optimum value. Since the complete information assumption is not valid in construction claims negotiation, these features are expected to be obtained through the learning mechanism based on agents' prior knowledge and the opponent's continuing offers during negotiation. Thus, an agent can estimate its opponent's utility according to its updated information of reservation value and optimum value.

5.3.2 Behaviour Theory Model

Both game theory and economic theory focus more on the quantitative aspects of the negotiation process, which are suitable for the application of intelligent agents. On the
contrary, behaviour theory focuses more on the qualitative aspects of the negotiation process (Susskind and Cruikshank, 1987), which are difficult to simulate and analyse by computer technologies. However, human behaviour aspects are emphasised by most of the researchers in construction claims negotiation (Bent, 1985; Zack, 1994). The limitation is obvious if the negotiation system does not take into account the human negotiation behaviour aspects.

As discussed in Chapter 3, different learning approaches have been adopted in multi-agent systems, such as: reinforcement learning, incentive learning and generic learning. This research adopts a different learning approach the “Bayesian learning mechanism”, which allows a negotiation agent to infer its opponent’s key negotiation features during the negotiation process. This method, based on the Bayesian rule, provides an effective approach for analysing the human risk perceptions and expectations, and mutual influence process during the claims negotiation process. An agent can adjust its negotiation strategy by analysing the opponent’s expectation. Also, since this method is based on a mathematical concept, it can be applied in computer systems.

5.4 MASCOT MODEL

Based on the identified key characteristics of construction claims negotiation and selected negotiation theory models, this section makes a detailed analysis on the system working domain, assumptions and requirements, negotiation protocol and strategies, and process model of the MASCOT model.

5.4.1 Negotiation Domain

Rosenschein and Zlotkin (1994) classify agent working domains as task-oriented domains (TODs), state-oriented domains (SODs) and worth-oriented domains (WODs), with each domain being a generalisation of the previous (see Section 3.4.3.4). Different working domains will determine different negotiation approaches for agents to achieve their
objectives. The agents in the MASCOT model work in the WODs because agents in construction claims negotiation cannot always achieve their full goal. On the contrary, they often have to make a compromise with other parties. Under this domain, agents assign worth to each potential state that represents the agent's desire. The worth function allows agents to compromise or release part of their goal. Those states with the highest value of worth might be thought of as those that satisfy the full goal. The worth function also facilitates the evaluation of different alternatives such as time extension and monetary compensation through the utility functions. In this study, utility functions are adopted to represent agents' worth functions\textsuperscript{13}.

5.4.2 Assumptions and Requirements

Since the MASCOT model adopts economic theory model as its essential negotiation model, some of the key assumptions on which economic theory is based are made for the MASCOT model. Assumptions about construction claims negotiation are also made to ensure that only the major problems of the claims negotiation are addressed in the model.

1) Assumptions

Initially, a number of simplifying assumptions is made to lay down the foundations of the MASCOT model. These assumptions include two aspects: the assumptions about construction claim negotiations and multi-agent systems (based on the assumptions of economic theory).

- **Quantitative negotiation**: the contractor's entitlement to the claim being negotiated has been established. Negotiation is only concerned with the amount of compensation in the claim.
- **Isolated negotiation**: no further change or information comes to agents during the negotiation. Moreover, an agent cannot commit itself as part of the current

\textsuperscript{13} Rosenschein and Zlotkin (1994) explain that a worth captures the desire of an agent. Worth functions allow agents to compromise on their goals, sometimes increasing the overall efficiency of the agreements. For a detailed discussion of the distinctions between utility function and worth function see Rosenschein and Zlotkin (1994).
negotiation to some behaviour in a future negotiation, nor can it expect that its current behaviour will in anyway affect a future negotiation.

- **Rationality**: agents are rational, and willing to maximise their utilities, or willing to reduce the risk of loss caused by a conflict deal.

- **Fixed utility functions**: each agent’s utility function is not changed during the negotiation. In other words, agents’ risk attitudes (i.e. risk averse, risk taking or risk neutral) are consistent during the negotiation.

- **Incomplete information**: each agent has complete information about itself, but incomplete information about the others (e.g. an agent’s reservation value and utility function are private information). Other agents can estimate such values and update their estimates through a learning process, but they would not know the exact figures.

- **Agreement Preference**: agents will prefer to reach an agreement rather than have a conflict deal even if the agreement might not bring them the maximum benefits.

- **Partial-symmetry**: three agents (i.e. the contractor agent, the engineer agent and the client agent) are involved in this model. The negotiations between the contractor agent and the engineer agent are symmetrical, whilst the negotiations between the client agent and the engineer agent are not. The relationship between the client agent and the engineer agent is treated as owner-servant. Furthermore, it is assumed that there is no direct negotiation between the client agent and the contractor agent; their negotiations are conducted through the engineer agent. However, this would not stop direct communications between them.

2) Requirements

As discussed in Chapter 3, several requirements should be taken into account prior to designing a multi-agent system. In this study, these are:

- **Suitability**: the system should best reflect the characteristics of construction claim negotiations, and should generate similar or reasonable negotiation outcomes to human negotiations.
- **Simplicity**: the negotiation process should be short and consume only a reasonable amount of communication and computing resources.

- **Distributed**: the decision making process should be distributed. There should be no central unit or agent that is managing the problem.

- **Efficiency**: agents should not squander resources when they come to an agreement. There should not be wasted utility when an agreement is reached. Ideally, agents as a group are expected to arrive at the optimal solution or a Pareto Optimality, where no agent could derive more from a different agreement without some other agent deriving less from that alternate agreement. However, in the MASCOT model, given the limitations of the real claims negotiation environment and the lack of complete information about each other, it is difficult for the agents to achieve the Pareto Optimality like many economic/game theoretical models have addressed\(^\text{14}\). Yet, there are possible solutions with a flexible range for a claims negotiation that can satisfy the negotiators' minimum requirements, and finally settle the claims. In such situations, although the solution is not an exact Pareto Optimality, agents would prefer to reach such an agreement rather than a conflict. In this study, if the MASCOT model can generate such solutions, the system is thought to be efficient.

- **Stability**: no agent should have an incentive to deviate from agreed-upon strategies. In a pure game situation, with complete information, the Nash equilibrium concept is used. If Nash equilibrium is reached, agents will have no incentive to deviate from agreed negotiation strategies. However, Nash equilibrium has two problems:

First, Nash equilibrium is based on the complete information assumption which is not true in most practical cases. One solution for this problem is the Bayesian-Nash equilibrium introduced by Harsanyi (1967, 1968). This equilibrium includes a set of beliefs (one for each agent) and a set of strategies. A strategy combination and a set of beliefs form a Bayesian-Nash equilibrium if the strategies are in Nash equilibrium.

\(^\text{14}\) For detailed discussion, see Rosenschein and Zlotkin (1994) and Harsanyi (1977).
given the set of beliefs, and the agents update their beliefs, according to Baye's rule. This study aims to reach a Bayesian-Nash equilibrium.

Secondly, the use of Nash equilibrium may not be an effective way of analysing the outcomes of the models of alternating offers (bargaining problems) (Kraus, 1995). There are two major problems. First, in some games there are no strategies that form an equilibrium (Sandholm, 1996). Secondly, some games may be in equilibrium only at the beginning of the negotiation, but may be unstable at intermediate stages. Nash equilibrium puts few restrictions on the outcome and may yield too many equilibrium points (Rubinstein, 1982), where one or more agents may prefer to diverge from their Nash equilibrium strategies. To avoid such problems, this research formalises the assumptions that are appropriate for the environments. For example, all agents suffer a loss over time, there is a finite set of agreements, and any agreement reached is better for all agents rather than opting out of the negotiations. These assumptions will push agents to an equilibrium point.

5.4.3 Roles of Players in MASCOT

There are two kinds of agents in the MASCOT model: task agents and utility agents. Task agents play the role of the original industry participants involved in construction claims negotiation. Utility agents are designed to facilitate negotiation within the MASCOT model.

5.4.3.1 Task Agents

Figure 5.1a shows a typical contractual relationship between construction project participants, whilst Figure 5.1b describes their relationships in claims management and negotiation. The relationships shown in these two figures reveal a general problem in claims management. That is, although there is no formal contractual relationship between the engineer and the contractor, it is the engineer, acting as the agent of the client, who directly supervises the contractor's work, negotiates and certifies the contractor's claims.
Many problems are created due to the conflicting authority and responsibility, because the engineer, in practice, can hardly be an impartial observer, the assumption on which the system is based.

Figure 5.2 shows the relationships between these three agents in the MASCOT model. In this model, direct communication between the client and the contractor is provided through their agents to avoid the problems caused by the engineer's unbalanced authority and responsibility. The interactions between the engineer and the client, and the interactions between the engineer and the contractor become more efficient with the support of multi-agent systems. This is one of the most important improvements for current claims negotiation. It is possible to resolve negotiation deadlocks between the contractor and the engineer due to non-agreement by the direct involvement of the client himself. Also, the problems caused by the engineer's inclination to cover personal faults are also avoided.
Besides the agents representing the major project participants, Figure 5.2 also shows the possible involvement of subcontractor and supplier agents. Although these agents are not further discussed in this study, their involvement will make the system more realistic. In practice, claims negotiation between the contractor and the engineer is often influenced by the subcontractors and suppliers because many work packages are actually done by specialist subcontractors.

The involvement of the subcontractor and supplier agents in the MASCOT model is an interesting area because the contractor agent has to negotiate with the subcontractor agents or the supplier agents before it accepts or rejects the engineer agent’s offer. Furthermore, the nature of claims and their negotiation between the contractor agent and the subcontractor agents (or the supplier agents) are different from that between the contractor agent and the engineer agent (or the client agent). In the former, claims and their negotiations are often conducted informally and effectively without a third party (like the engineer), while the latter are more formal as discussed in this study. Therefore,
different negotiation protocol and strategies would need to be developed when the subcontractor agent and suppler agent are involved in the system.

5.4.3.2 Utility Agents

There are two kinds of utility agents in the MASCOT model: the interface agents and the mediator agent\(^{15}\). The interface agents build the necessary links between task agents and their owners. The initial information and the data for expanded solution searching are input through the interface agents. The mediator agent plays two roles in the model both as a facilitator and a case-based mediator. The former role is fulfilled in the implementation as the utility agent "Facilitator" (see Chapter 6); the latter, inspired by the industrial practice, (i.e. the mediator approach in resolving construction disputes is commonly accepted and applied), will be discussed in further study. However, it is still beneficial to discuss some of the interesting points for the potential development of the mediator agent.

Like a human mediator, the proposed case-based mediator agent can provide the following services to the negotiation system:

- storing related contract documents, such as Bills of Quantities, labour rates and schedules;
- providing suggestions to negotiating agents about their possible payoffs based on the knowledge of previous claims cases;
- monitoring the negotiation process; and
- suggesting when the negotiations should break down or stall.

The involvement of the mediator agent will also change the structure of the current negotiation system, and provide a chance to improve the efficiency of the system. Sycara

\(^{15}\) The other utility agents such as Nameserver, Facilitator and Viewer will be discussed in Chapter 6.
(1987) has pointed out some of the advantages of a case-based mediator in her study, such as:

- case-based inference minimises the need for information exchange, thus minimising communication overhead;

- anticipating and avoiding problems through reasoning from past failures helps the agents minimise the exchange of proposals that will be rejected; and

- if the repair of a past failure is also remembered, computation by each agent is minimised.

During negotiation, every negotiating agent can inquire from the mediator agent about the possible outcomes of the current claims. After receiving the inquiry from negotiating agents, the mediator agent chooses an appropriate previous case from its database, identifying differences between that case and the current situation, and then using those differences to criticise and modify the previous solution to fit the current problem. Past negotiation experience from the former cases is used as a guide to present negotiations. When very similar cases are available, agents can make their initial offer or counteroffer based on the experience from these precedence cases, and a compromise may be quickly reached.

5.4.4 Negotiation Protocol and Strategies

The development of the MASCOT negotiation mechanism covers two distinct aspects: to define an acceptable protocol for agent interaction; and to formulate strategies for the agents participating in the negotiation. The protocol specifies the kinds of deals that the agents can make, as well as the sequence of offers and counter-offers that are allowed. Given the protocol, a negotiation strategy specifies precisely each agent's reaction to every possible course of events (Rosenschein and Zlotkin, 1994).
5.4.4.1 Negotiation Protocol

As shown in Chapter 3, there are two major kinds of negotiation protocols adopted in MAS negotiation systems: those that facilitate co-operative or co-operative-competitive problem solving, which are mainly developed or modified from Contract Net Protocol (CNP); and those that facilitate bargaining between different parties (Table 5.1). Monotonic Concession Protocol (MCP) belongs to the latter, and is selected as the basic protocol of the MASCOT model. This is because most of the negotiation protocols for multi-agent systems are developed for specific questions and therefore they are only suitable for these situations. On the other hand, MCP as a simple and general negotiation protocol is applicable to various situations and can therefore be modified for many specific problems.

5.4.4.1.1 Monotonic Concession Protocol

In MCP, agents start by simultaneously proposing one deal from the space of possible deals; an agreement is reached if one agent matches what the other one asked for, or exceeds what the other one asked for. The protocol continues to another round if neither agent matches or exceeds the other’s demand. An agent is not allowed to offer the other agent less than it did in the previous round. If neither agent concedes at a certain step, then the negotiation ends and the protocol specifies that a deadlock has been reached. In this protocol, agents cannot backtrack, nor can they both simultaneously stand still in the negotiation more than once.

The advantage of the MCP is that it ensures convergence, or puts a stop to the negotiation promptly when convergence is not occurring; and it is symmetrically distributed: no agent plays a special role. To be able to follow the rules in a standard MCP, each agent needs to know the other’s utility (i.e. the set of tasks and the cost function) (Rosenschein and Zlotkin, 1994). However, this complete information requirement is not true in the claims negotiation environment. To make MCP work in claims negotiation, some aspects need to be modified.
Table 5.1 Some of the important protocols

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Work domain</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract net protocol (CNP)</td>
<td>Task distribution</td>
<td>Davis and Smith, 1983</td>
</tr>
<tr>
<td>Distributed scheduling protocol (based on CNP)</td>
<td>Task distribution</td>
<td>Malone et al., 1988</td>
</tr>
<tr>
<td>Multistage negotiation protocol</td>
<td>Resource allocation</td>
<td>Conry et al., 1988</td>
</tr>
<tr>
<td>Service negotiation protocol</td>
<td>Client-server contract negotiation (task distribution)</td>
<td>Sierra et al., 1997</td>
</tr>
<tr>
<td>Voting protocol</td>
<td>Task distribution or resource allocation</td>
<td>Ephrati and Rosenschein, 1991</td>
</tr>
<tr>
<td>Time counting protocol</td>
<td>Task distribution or resource allocation</td>
<td>Kraus and Wilkenfeld, 1995</td>
</tr>
<tr>
<td>Extended CNP</td>
<td>Task distribution (Freight dispatch)</td>
<td>Sandholm, 1993</td>
</tr>
<tr>
<td>Mediator protocol</td>
<td>Conflict resolution</td>
<td>Sycara, 1987</td>
</tr>
<tr>
<td>Monotonic Concession protocol</td>
<td>Task distribution</td>
<td>Rosenschein and Zlotkin, 1994</td>
</tr>
<tr>
<td>One-step Protocol</td>
<td>Task distribution</td>
<td>Rosenschein and Zlotkin, 1994</td>
</tr>
<tr>
<td>Unified Negotiation Protocols</td>
<td>Semi-cooperation</td>
<td>Rosenschein and Zlotkin, 1994</td>
</tr>
<tr>
<td>Multi-course protocol</td>
<td>Agent learning</td>
<td>Dubios and Prade, 1994</td>
</tr>
</tbody>
</table>

5.4.4.1.2 Modified Monotonic Concession Protocol (mMCP)

Considering the important characteristics of claims negotiation discussed earlier, major modifications to MCP have been made as follows:
• Conflict deal

In the standard MCP, if neither agent concedes at the same step, then the negotiation ends with a conflict deal (Rosenschein and Zlotkin, 1994). This restriction is relaxed in the MASCOT model. Negotiations will not necessarily fall into a conflict deal even if the contractor agent and the engineer agent stand still for more than one encounter. Three factors reduce the possibility of conflict deals. They are:

a) the involvement of the client agent;
b) the expanded solution-searching; and
c) intervention by the mediator agent.

A conflict deal will be reached if there is no agreement zone existing between the contractor agent and the engineer agent. In such a situation, the involvement of the client agent provides an opportunity to avoid the conflict deal. The client agent will first check whether there is any possible deal within its negotiation zone that could be accepted by the contractor and lead to a possible agreement. If such a deal or agreement zone is found, the client agent will ask the engineer agent to expand its negotiation zone, and push the negotiation forward. If this strategy fails, expanded solution-searching strategies will be adopted, which includes searching for possible tradeoffs between negotiation items and interacting with the agents' owners to seek for constraint relaxation.

• Incomplete information

In MCP, each agent needs to know the other’s full information, especially utility functions, to monitor whether the other has offered it more (or the same) at each step of negotiation. It is mutually verifiable whether the other agent has followed the rules of the protocol. In claims negotiation, the contractor agent cannot have exact information about the engineer agent’s (and the client agent’s) final goals, utility functions or risk perceptions, and vice versa. Even the engineer agent’s utility function is not identical and fully transparent to the client agent due to their different positions, interests and attitudes.

---

16 See Section 5.4.3.2. A further study of the mediator agent will be conducted in future.
to a claim. Thus, it is impossible for an agent to know whether the other is following the rules of the MCP.

To overcome this problem, Zeuthen's strategy and a learning mechanism are adopted. By following Zeuthen's strategy, agents will make concessions to avoid the loss of conflict. In other words, agents cannot offer less or backtrack at the next encounter which the MCP addresses. Moreover, the learning mechanism allows an agent to estimate its opponent's key negotiation features and update its estimation continuously during the negotiation process. Negotiations are conducted based on estimated information rather than the complete information.

- Time penalty

The MASCOT model takes the passage of time during the negotiation process into account in facilitating convergence. The time taken by each agent during negotiation is considered as a penalty element in its utility function. Each agent has a different cost $C > 0$ for a time unit which it spends during negotiation. The value of $C$ depends on each agent’s time sensitivity (which varies with the negotiation situation). The time penalty can be a fixed amount (e.g. £5 per iteration) or a fixed rate of the agent’s utility, (e.g. a deduction of 1% of utility per iteration). For example, if an agreement is reached in time period $t$ (or $t$ iterations), the agent’s time penalty will be $Ct$. As a result, time penalty becomes a driving force for agents, especially those with a high time penalty, in conceding to reach an agreement.

- Involvement of the client agent

In this study, the client agent will get involved in a negotiation when it finds that there is no chance for the contractor agent and the engineer agent to reach an agreement. The client agent can identify this situation directly by observing both parties’ offers (e.g. the client agent may think that a party has reached its deadline if it continuously makes very small or no concessions after several steps of negotiation. However, the client agent
should also be able to distinguish the party’s ‘faked’ or ‘real’ reservation amount), or indirectly by observing the time penalties of both parties (e.g. when a party’s time penalties pass a certain threshold, it can be assumed that this party is ready to accept a conflict deal because the loss of a conflict deal for it is higher than that of the time penalty). In some other cases, the client agent may be invited to get involved in the negotiation by one of the parties because a conflict deal is very harmful to that party.

The involvement of the client agent will not change the negotiation relationship between the contractor agent and the engineer agent. That is, the client agent and the engineer agent will work together as one negotiator to negotiate with the contractor agent. The contractor agent still receives offers or counter-offers from the engineer agent (Figure 5.3). This facilitates the negotiation process in several ways:

a) The contractor agent can send its offer directly to both the client agent and the engineer agent to ensure that both of them get the same information. The client agent can make its judgement independently with the direct information. Thus, the problems caused by the engineer agent’s attempt to screen the contractor agent’s information can be avoided. This is especially important for the further development of the system, when agents may be able to include arguments with their offers.

b) In some claims, the client agent may have a different utility function to the engineer agent because of their different interests, expectations and attitudes to the claim. If the negotiation between the contractor agent and the engineer agent falls into a deadlock, and there is a possible deal within the client agent’s negotiation zone that can be accepted by the contractor agent to reach an agreement, the engineer agent (as a representative of the client agent) will be requested to adopt the client agent’s offer in the next iteration\textsuperscript{17}.

\textsuperscript{17} It is assumed that the client agent fully understands the consequences of its offer.
5.4.4.2 Negotiation Strategies

Given the above protocol, agents will adopt different negotiation strategies according to their evaluations of the possible deals in different situations. The possible negotiation strategies that agents may adopt in this circumstance include:

a) inflate the opening demands;
b) adopt an intransigent stance until the other side is ready to concede;
c) secure concessions before giving concessions;
d) make tit-for-tat counter proposals;
e) make big concessions for a quick compromise;
f) join in the negotiation (for the client agent only);
g) undertake tradeoffs between negotiating items;
h) search for new solutions outside the current negotiation items; or
i) request the owner to relax the negotiation constraints.

Considering these strategies, a) is simple and straightforward, agents may exaggerate their initial demands according to their experience; b), c), d) and e) are about issues such as when an agent should concede and how much it should concede, which are the most
important issues in claims negotiation strategies; f) is about when and how the client agent should get involved in the negotiations; g), h) and i) are about expanded solution searching. The following discussion will focus on the strategies regarding when and how an agent should concede.

An agent's decision to select a certain strategy depends on how it evaluates a possible deal and its position in that deal. In fully cooperative situations, the parties are united by the subordinate goal of achieving a globally optimal solution, which often requires sacrificing personal benefit in the interest of increased global benefit. In a co-operative-competitive negotiation, the agent will evaluate a deal from both individual and group perspectives. Decisions are made based on the principle that they maximise both personal and group utilities. In the MASCOT model, however, agents are self-interested. Each agent is concerned only about its own utility without considering that of the group or other agents. On the other hand, agents are contractually obliged by the project contract, and the inclination not to break the negotiation. Thus, agents in the MASCOT model adopt Zeuthen's negotiation strategy. A detailed discussion of the MASCOT negotiation strategies is presented as follows.

5.4.4.2.1 Utility Maximisation

In claims negotiation, an agent's utility function is mainly determined by three elements: money, time and other issues such as the possibility of further claims and long term relationships (Figure 5.4). The utility function can be expressed as:

\[ U = \sum W_i M_i + \sum W_j T_j + \sum W_k O_k \]

Where,
- \( U \) - the utility of an agent [0,1];
- \( W \) - the relative weight assigned to a certain item, \( \sum W_i + W_j + W_k = 1 \);
- \( M_i \) - the value of sub-item i in monetary terms [0,1];
- \( T_j \) - the value of sub-item j in time terms [0,1];
- \( O_k \) - the value of sub-item k in other terms [0,1].
Figure 5.4 The general utility consideration of agents

Since an agent’s utility is multi-faceted, it will be a very complex task to maximise the utility in one step whilst considering all the factors involved. Thus, this work can be separated into two steps:

- to maximise every single claim item’s utility which is mainly applicable to money and time issues;
- to maximise the agent’s overall utility through tradeoffs between different items.

Since each item may have a different weight to each agent, an agent may prefer to
increase the value of highly weighted items and give something from the other lowly weighted items to maximise their total utility. Tradeoffs are common in real claims negotiation. Both parties may benefit from the tradeoffs. This task can be achieved through the expanded solution-searching strategies.

This study essentially focuses on the first step (i.e. to maximise a single item’s utility). As stated earlier, it is assumed that the negotiation item is concerned with monetary compensation for a claim. Such an approach is also applicable to the claims for time extension.

5.4.4.2.2 Conflict Avoidance Approach

Unlike the common utility maximisation approach, this study builds a negotiation mechanism (for single claim item) based on Zeuthen’s strategies because the risk avoidance principle in Zeuthen’s model reflects project participants' risk perceptions in claims negotiation (i.e. both parties try to avoid the losses caused by conflict results). Furthermore, Harsanyi (1965, 1977) demonstrates that Zeuthen’s solution is identical to Nash’s method (1950); that is, parties settle at the point that maximises the product of the difference between what bargainers get from conflict and what each gets from the settlement point.

Zeuthen (1975) argues that each negotiator will assess the gains and losses associated with his bargaining strategies at any given point in a negotiation on an expected-utility basis. This leads to a two-stage model of the negotiation process. Firstly, the individual always compares the certain value he can obtain by accepting the opponent’s current offer with the expected value of holding out for his own most favoured outcome (that is, the expected value of settlement at his preferred outcome together with the expected value of a conflict or breakdown in the negotiations). Given specific values for the opponent’s offer, his own most favoured outcome, and the results of conflict, he then calculates the maximum probability of conflict he would be willing to accept in preference to acquiescing in the current offer of the other side.
Secondly, Zeuthen makes the explicit assumption that the next concession will always be made by the party willing to accept the smaller risk of conflict at any given moment in time. But in making a concession the negotiator need not give in entirely; he need only reduce his own demand to the point where he is willing to accept a greater risk of conflict than the opponent. Accordingly, a situation arises in which each negotiator must continue to concede until he is willing to accept a larger risk of conflict than his opponent, at which point the relationship is reversed. Under suitable conditions, this interaction process leads to a determinate solution at the point where the product of the utility of the two sides reaches its maximum value. Thus, the outcome of Zeuthen's bargaining process is similar to Nash's solution in game theory, even though the basic features of his model differ substantially from those of Nash's model.

Zeuthen (1975) concludes that "an agent makes its decision of concession based on how much it has to lose by running into conflict at that time". If the agent has already made many concessions, it will have less to lose from a conflict, and will be less willing to concede. Thus, it has a high acceptability to risk conflict. If each agent's willingness to risk conflict can be measured, the agent with less willingness to risk will make a concession. The criteria for risk evaluation can be formulated into the following equations:

\[
\text{Risk}_i = \frac{\text{utility agent 1 loses by conceding and accepting agent 2's offer}}{\text{utility agent 1 loses by not conceding and causing a conflict}}
\]

or

\[
P_{c \text{ max}} = \frac{U_{cc}^i - U_{ct}^i}{U_{cc}^i - U_{cc}(C)};
\]  
\[
P_{e \text{ max}} = \frac{U_{ec}^i - U_{ct}^i}{U_{ec}^i - U_{e}(C)}
\]

(adapted from Rosenschein and Zlotkin, 1994)

Where,

- \(P_{c \text{ max}}\) - the maximum likelihood of risk acceptable to the contractor\(^{18}\);
- \(P_{e \text{ max}}\) - the maximum likelihood of risk acceptable to the engineer;
- \(U_{cc}^i\) - the contractor agent's utility based on its offer in \(t\) iteration;

\(^{18}\) For simplicity, \(P_{\text{max}}\) is often called the maximum risk acceptability.
$U_{ce}^t$ - the contractor agent's utility based on the engineer agent's offer in $t$ iteration;

$U_{c}(C)$ - the contractor agent's utility for a conflict deal;

$U_{ee}^t$ - the engineer agent's utility based on its current offer in $t$ iteration;

$U_{ce}^t$ - the engineer agent's utility based on contractor agent's offer in $t$ iteration; and

$U_{c}(C)$ - the engineer agent's utility for a conflict deal.

At every step, each agent calculates and compares its Risk ($R_{1}$ or $P_{max}$) and that of its opponent. If Agent 1's Risk ($R_{1}$ or $P_{max}$) is higher than that of Agent 2, Agent 1 will have less to lose from a conflict, and will be less willing to concede, and risk reaching a conflict. Therefore, Agent 2 (with smaller risk acceptability) will make the next concession. The concession amount is determined by different concession mechanisms.

The application of the Zeuthen's model is limited by the perfect information assumption. To make this model work in the claims negotiation environment where both negotiation parties try to keep their private information, a learning mechanism is introduced for agents to estimate the opponent's utility function and update their beliefs during the negotiation process. The main question here pertains to how an agent uses its beliefs during the negotiation, how it updates its beliefs according to the information it gathers during the negotiation process, and how an agent influences its opponent's beliefs.

5.4.4.2.3 Bayesian Learning Approach

The importance of learning in negotiation has been recognised in the MAS research community as fundamental for understanding human behaviour as well as for developing new solution concepts. Several learning approaches in MAS have been discussed in Chapter 3. Bayesian learning approach is recognised as an extremely powerful learning method for artificial intelligence (Russell and Norvig, 1995). Based on Bayesian Inference, researchers have developed quite different learning approaches, such as: Bayesian learning (Zeng and Sycara 1998; Kellogg and Gmytrasiewicz, 1997), Bayesian network learning (Sahin, 1999) and Bayesian classifier learning (Bui et. al, 1996). This study focuses on Bayesian learning.
In this approach, once an agent receives an offer (or counteroffer) from its opponent, the agent will analyse the offer, modify its beliefs about the opponent based on the Bayesian learning approach, and adjust its negotiation strategy accordingly. These beliefs about the opponent could be: the negotiation strategies; the factual aspects of other agents (e.g. the payoff functions); the decision making process of other agents (e.g. the reservation values and the most preferred amount); the meta-level issues (e.g. the overall negotiation style and risk-taking attitude), and so on.

The following discussions focus on how the contractor agent modifies its beliefs about the engineer agent’s reservation value. A reservation value is defined as **the maximum or minimum amount which a party can offer to or accept from its opponent** (Figure 5.5). Since the negotiation process is identical for the contractor agent and the engineer agent, the method can be applied to the engineer agent as well.

<table>
<thead>
<tr>
<th>Labour, material and equipment cost</th>
<th>Loss of productivity</th>
<th>Overheads</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reservation value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimum value</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5.5a An example of the contractor’s negotiation zone**

- Contractor agent’s reservation value
- Contractor agent’s initial offer

- Agreement zone
- Engineer agent’s initial offer
- Engineer agent’s reservation value

**Figure 5.5b An example of the agreement zone between agents**
During claims negotiation, the reservation values are private information. It is impossible for an agent to know the exact value of its opponent's reservation value. However, an agent could update its beliefs about its opponent's reservation value based on its interactions with the opponent and its domain knowledge by using the Bayesian updating mechanism. Therefore, the agent could gain a more accurate expectation about its opponent's risk acceptability and choose the offer that maximises the expected payoff given the information available at this stage.

For a partition \( B_1, B_2, \ldots, B_n \) of all possible outcomes, Bayesian rule can be expressed as (Pitman, 1993):

\[
P(B_i | A) = \frac{P(B_i)P(A | B_i)}{\sum_{k=1}^{n} P(A | B_k)P(B_k)} = \frac{P(B_i)P(A | B_i)}{P(A)}
\]

Where,

- \( P(B_i | A) \) - the conditional probability that event \( B_i \) happens given the occurrence of event \( A \);
- \( P(B_i) \) - the unconditional probability that event \( B_i \) happens;
- \( P(A | B_i) \) - the conditional probability that event \( A \) happens given the occurrence of event \( B_i \);
- \( P(A) \) - the unconditional probability that event \( A \) happens.

Bayesian rule provides a means for agents to improve their beliefs about their opponent's reservation value according to the opponent's previous offers. Several elements are defined in this study as follows:

- \( R \) - the engineer agent's reservation value;
- \( e_i \) - the engineer agent's offer at encounter \( i \);
- \( R_i \) - a set of the contractor agent's partial beliefs (hypotheses) about the engineer agent's reservation value \( R \) (e.g. \( R_1 = 100; R_2 = 150 \ldots (i=1,2,\ldots n) \));
- \( P(R_i) \) - the probabilistic evaluation over the set of hypotheses \( \{R_i\} \), which are the contractor agent's prior knowledge (e.g. \( P(R_1) = 0.75, P(R_2) = 0.85 \ldots (i=1,2,\ldots n) \));
- \( \sum R_i^*P(R_i) \) - the current estimate of \( R \) can be calculated as a mean.
The Bayesian learning mechanism is applied when the contractor agent receives a new offer from the engineer agent. Based on its domain knowledge about the engineer agent, a new offer enables the contractor agent to acquire new insights about the engineer agent's reservation value in the form of posterior subjective evaluation over $R_i$. The contractor agent's domain knowledge about the engineer agent's strategy can be expressed as "usually the engineer agent will offer an amount which is 20% lower than its reservation value". Such a relationship can be represented by a set of conditional statements, for example:

$$P(e_2 | R_2) = 0.95$$, where $e_2$ represents "offer_{engineer} = 120", and "$R_2 = 150$".

Given the encoded contractor agent's domain knowledge in the form of conditional statements and the engineer agent's offer, the contractor agent can use the standard Bayesian rule to revise its beliefs about the engineer agent's reservation value $R$:

$$P(R_j | e) = \frac{P(R_j)P(e | R_j)}{\sum_{k=1}^{n} P(e | R_k)P(R_k)} = \frac{P(R_j)P(e | R_j)}{P(e)}$$

Where,

- $P(R_j | e)$ - the probability that the engineer agent's reservation value is $R_j$ under the condition that his offer is $e$;
- $P(R_j)$ - the probability that the engineer agent's reservation value is a certain $R_j$;
- $P(e | R_j)$ - the probability that the engineer agent's offer is a certain $e$ under the certain reservation value $R_j$;
- $P(e)$ - the probability that the engineer agent's offer is $e$.

The advantage of the Bayesian learning mechanism is that it provides a simple method to update an agent's beliefs about the opponent based on the opponent's offers during the negotiation process. The updated belief then becomes the agent's domain knowledge in the next updating process. An agent can finally get a relatively accurate estimate about
the opponent’s reservation value even if its initial domain knowledge is not so accurate (Figure 5.6).

For example:

a) At the beginning, the contractor agent may not have any other additional information. Its prior knowledge can be described as: $P(R_1)=0.5; P(R_2)=0.5$;

b) In addition, we suppose that the contractor agent is aware that the engineer agent will normally offer an amount which is 20% lower than his reservation value, part of which is encoded as: $P(e_1|R_1)=0.95$ and $P(e_1|R_2)=0.75$, where $e_1$ represents the engineer agent’s offer of 80, for instance.
c) Suppose the engineer agent offers 80 for the claim, the contractor agent can estimate $R$ from the following equations:

$$P(R_1 | e) = \frac{P(R_1)P(e | R_1)}{\sum_{k=1}^{2} P(e | R_k)P(R_k)} = \frac{0.5 \times 0.95}{(0.5 \times 0.95) + (0.5 \times 0.75)} = 0.56$$

$$P(R_2 | e) = \frac{P(R_2)P(e | R_2)}{\sum_{k=1}^{2} P(e | R_k)P(R_k)} = \frac{0.5 \times 0.75}{(0.5 \times 0.95) + (0.5 \times 0.75)} = 0.44$$

Prior to receiving the engineer agent's offer (80), the contractor agent would think that the engineer agent's reservation value is: $R = 0.5 \times 100 + 0.5 \times 150 = 125$. After receiving the offer, its current estimation of the engineer agent’s reservation value is: $R = 0.56 \times 100 + 0.44 \times 150 = 122$. Therefore, the contractor agent can make its counteroffer based on its new estimate of reservation value.

5.4.4.2.4 Concession Mechanism

By working out the reservation value, each agent is able to know the opponent’s utility functions, and therefore, it is possible to compare the maximum likelihood of risk ($P_{\text{max}}$) which it can stand and that of its opponent (which it estimates). The agent will make its decision on whether to concede or not accordingly. In the case that both agents think that their $P_{\text{max}}$ are higher than their opponent’s (i.e. both agents stand still), the agent with higher time penalty will concede at the next encounter or at the encounter where its $P_{\text{max}}$ is smaller than its opponent’s caused by its time penalty. This keeps Zeuthen’s strategy stable.

The above discussion addresses the question “who should make a concession at the next iteration?” However, it does not specify how much an agent should concede. Two different approaches have been studied for the concession step:
A simple approach

Rosenschein and Zlotkin (1994) suggest that the step an agent concedes should be the minimum sufficient to make its opponent's maximum risk acceptability \( P_{\text{max}} \) smaller than or equal to its own. Otherwise, the agent will offer the same deal as the previous one. By following this approach, negotiation is simple and straightforward. Agents will concede alternately until the maximum risk of conflict for both parties is zero (i.e. the settlement point). Agents will surely reach an agreement. However, this approach does not reflect the complex nature of real construction claims negotiation.

A complex approach

In practice, it is quite conceivable that an agent will have to make more than one concession before the opponent's \( P_{\text{max}} \) becomes lower than its own. Thus, Zeuthen does not suggest that Agent 1's concession will invariably be followed by Agent 2's concession, but only that this will occur if Agent 1's concession reduces Agent 2's \( P_{\text{max}} \) below that of Agent 1. Agents (or their owners) can decide a concession rate. To maximise utility, every agent will try to minimise its concession step, for example, an agent may prefer 3% (of its utility deduction) rather than 5% as its concession rate. On the other hand, the agent has to consider its utility loss caused by time penalty. The agent may prefer a 5% concession rate rather than 3% when it suffers a high time penalty.

For example, in iteration \( i \), the contractor agent concedes 3% (of its utility) which causes a loss of £6. In the mean time, the agent also suffers a time penalty of £5. However, the contractor agent's maximum risk acceptability \( P_{\text{max}} \) is still smaller than that of the engineer agent after this iteration. As a result, the contractor agent has to make another concession at the next encounter. In iteration \( i+1 \), the contractor agent will still concede 3% (which causes a loss of £6), and suffers another time penalty of £5. The total loss of the contractor agent is £22 after these two iterations (suppose that its maximum risk acceptability is higher than that of the engineer agent at the end of iteration \( i+1 \)). On the other hand, the contractor agent might prefer to concede 5% in iteration \( i \) (suppose the
contractor agent’s loss is £10) to make its maximum risk acceptability higher than the opponent’s in one encounter because the total loss of the contractor agent is only £15 in this situation. The selection of the concession rate will be made according to the agent’s time penalty. The advantage of this approach is that it is more realistic. The drawback is that it is relatively complex.

To keep the MASCOT model simple and specific, this research adopts the simple concession approach.

5.4.4.2.5 Involvement of the Client Agent

The above concession mechanism is based on the principle of conflict avoidance; thus, it is possible for agents to reach an agreement without the involvement of the client agent if there is an agreement zone between the contractor agent and engineer agent. However, there are cases where there is no agreement zone between the contractor agent and the engineer agent. An agent’s offer will be beyond its opponent’s reservation value. In these cases, if negotiations are only between the contractor agent and the engineer agent, the negotiations will fall into deadlock and have to be terminated. To avoid conflict, the client agent can get involved in the negotiations.

For example, suppose the negotiation participants’ positions are:

E1 - the engineer agent’s initial offer = 100;
E2 - the engineer agent’s reservation value = 200;
C1 - the contractor agent’s initial offer = 500;
C2 - the contractor agent’s reservation value = 300 (Figure 5.7).
Thus, there is no scope for a deal between the contractor agent and the engineer agent. In this situation, the client agent may join in the negotiation if it thinks there are some other solutions that could bring it more benefits rather than a deadlock. Unlike the peer-to-peer negotiation relationship between the contractor agent and the engineer agent, the negotiation between the client agent and the engineer agent can be understood as a client-servant negotiation in this situation. That is, if the client agent has common negotiation zones with both the engineer agent and the contractor agent (e.g. the client agent’s negotiation zone is (150, 350) in this case), the engineer agent will modify its reservation value to meet the client agent’s (i.e. 350 in this case). Consequently, the negotiation zone of the engineer agent is expanded due to the involvement of the client agent. It is possible to reach an agreement within (300, 350). In the case that both the client agent and the engineer agent do not have an agreement zone with the contractor agent, an expanded solution searching strategy will be triggered.

5.4.4.2.6 Expanded Solution-Searching

The expanded solution searching strategies are highly related to the innovation ability of agents and their owners. These strategies can resolve the conflicts which autonomous agents themselves cannot solve through the negotiation of a single claim item. Two common expanded solution searching strategies are:
• Tradeoffs between different negotiation items

Agents first check the possible payoffs and weights of all their negotiation items (of the same claim). The contractor agent may agree to reduce its reservation value for a problem negotiation item to the engineer agent's benefit in order to reach an agreement. As a tradeoff, it will ask the engineer agent to give something on other claim items to which the contractor agent may have assigned more weight. Therefore, the contractor agent's whole utility can still be maximised, and vice versa. This tradeoff strategy can be between different items of a claim (i.e. tradeoffs between time and money for a claim 1) or different items of different claims (i.e. tradeoffs between items of claim 1 and items of claim 2). In the latter case, a new solution may be found outside the current negotiating items. This strategy could also be applicable to maximise an agent's whole utility even if there is no conflict deal.

• Relaxation of negotiation constraints

This strategy requires the involvement of some or all the autonomous agents' owners. When an agent finds that it cannot move the negotiation forward by adopting the above strategies, it can either terminate the negotiation, or request its owner to input new information or to relax some negotiation constraints. If the owners could expand the negotiation sets or remove some limitations, it is possible for the negotiating agents to reach an agreement.

5.4.5 Process Model

Figure 5.8 shows the MASCOT process model based on the IDEF0 modelling method. It includes three major steps: making an initial offer, evaluating the offer and making a counteroffer, and conducting solution-searching.
5.4.5.1 Making an Initial Offer

When the contractor agent (Con-Agent) receives information from its owner and identifies that the claim is within its authority, it will calculate the actual claim cost C1 based on the input information as well as the related provisions in the contract documents (Rule 1 for the actual cost).

Based on C1, the Con-Agent decides a reservation value C2 for this claim item according to factors such as the importance, potential influence and the possibility of winning (Rule 2 for the reservation value). The Con-Agent also works out an initial offer amount C3 according to the C1 and C2, evidence, chance of success, and perceptions of the engineer agent (Eng-Agent) (Rule 3 for the initial offering amount\(^\text{19}\)). For example, if the claim is minor and clear, the contractor agent may submit an amount which is a little bit higher than C1; however, if the claim is vague and has a high potential influence on further claims, the offer amount may be much higher than C1.

Alternatively, the contractor can directly input the real cost of C1 and determine the reservation value C2 and the initial offer C3.

Finally, the Con-Agent submits the C3 to the Eng-Agent, with copies to the client agent (Cli-Agent).

5.4.5.2 Evaluating the Offer and Making a Counteroffer

- Engineer agent

After receiving the claim amount C3 from the Con-Agent, the Eng-Agent calls for all the information related to the claim (e.g. code of the claim item, nature of the claim, possible influence, and related provisions in contract documents) from its database, and inform its

\(^{19}\) The factors in rule 2 and rule 3 are determined by the attributes identified in the generic claims negotiation model.
owner if it is necessary. Like the Con-Agent, the Eng-Agent will calculate an actual cost
E\textsubscript{1} based on the above information (Rule 4), and work out its reservation value E\textsubscript{2} and
the counteroffer amount E\textsubscript{3} (Rule 5, 6).

Alternatively, the engineer can determine the E\textsubscript{1}, E\textsubscript{2} and E\textsubscript{3}.

The Eng-Agent then sends its counteroffer to the Con-Agent with copies to the Cli-Agent.

- Contractor agent

The Con-Agent will update its belief about the Eng-Agent's reservation value after
receiving the Eng-Agent's counteroffer E\textsubscript{3}, then calculate the maximum risk acceptability
of conflict for both parties, compare the result, and finally decide whether it should make
the next concession and the concession amount. The Con-Agent will make the new
counteroffer to the Eng-Agent, and inform the Cli-Agent.

- Engineer agent

The Eng-Agent will conduct the same operations as the Con-Agent once it receives the
Con-Agent's new counteroffer, and makes its own new offer with copy to the Cli-Agent if
it is its turn to make a concession.

This negotiation process will continue until an agreement or a deadlock is reached.

5.4.5.3 Conducting Expanded Solution Searching

- Involvement of the client agent

Once a deadlock is identified either by the Cli-Agent or by the other agents, it is possible
for the Cli-Agent to get involved in the negotiation. The Cli-agent may expand the Eng-
Agent's negotiation zone by searching the new possible deal which can be accepted by
the Con-Agent. If such a deal is found, a new offer based on the deal is made to the Con-Agent by the Eng-Agent. It is then possible to reach an agreement.

- **Constraints relaxation**

If the involvement of the Cli-Agent fails to resolve the deadlock, the problem will be reported to the individual agents’ owners. The owners can either relax some negotiation constraints or input some new information to their agents to facilitate negotiation into an agreement. Otherwise, a final conflict arises. This may then be resolved by arbitration or litigation.
Purpose: Design the MAS construction claims negotiation process.
Viewpoint: The MASCOT design team.

Note: 1. Agents include the contractor agent, the engineer agent and the client agent.
2. Project participants include the owners of negotiation agents.

| A-0 | Negotiate construction claims | Figure 5.8 a |
Contract & conditions of contract

Calculate cost of the claim item

Claim items

Actual cost

Assess self-situation

Contract & conditions of contract

Reservation and most preferred amount

Evaluate the opponent's situation

Contract & conditions of contract

Initial offer

Con-agent

Con-agent

Con-agent

Con-agent

Inform the Cli-agent

A1

Make initial offer

Figure 5.8 c
A2: Evaluate offer and make counteroffer
A21 | Assess the contractor agent’s offer | Figure 5.8 e
Figure 5.8f

Contract expanded solution searching

Note: The function of A34 will not be implemented in this study.
5.5 SUMMARY

Based on previous studies, this chapter presented the MASCOT model. It first addressed the nature and essential characteristics of construction claims negotiation. This was identified as contractually obliged self-interested negotiation. Furthermore, other characteristics of claims negotiation were highlighted such as: participant-dependent information, strategy-influenced process, important time impacts, and possible tradeoffs between different negotiation items. These characteristics were considered crucial for the development of the MASCOT negotiation protocol and strategies.

This chapter then discussed the negotiation model for the MASCOT model. Zeuthen's bargaining model was selected as the MASCOT negotiation model because it reflects the contractually-obliged self-interested nature of construction claims negotiation, and enables a solution which is identical to Nash's solution. Importantly, the Bayesian learning approach was integrated into the Zeuthen's model to keep the negotiation strategies stable in the claims negotiation environment.

Six major aspects of the MASCOT model were then described in detail. These included the negotiation domain; the important system assumptions and requirements; and the roles of each player, task agents and utility agents. Also described was an improved monotonic concession protocol, which ensures convergence of the negotiation, as well as the stability and effectiveness of the system. Integration of Zeuthen's negotiation strategy with the Bayesian learning approach was then described and the expanded solution-searching strategies discussed. Finally, the MASCOT process model was presented, and illustrated using the IDEF0 modelling notation.
CHAPTER 6
IMPLEMENTATION OF THE MASCOT MODEL

6.1 INTRODUCTION

Model implementation is an important stage during the MASCOT model construction process. It aims to encapsulate the developed MASCOT model in an agent application environment. The specific objectives include:

- To select a proper agent development toolkit for the implementation of the MASCOT model; and
- To develop a prototype system based on the MASCOT model and the selected toolkit;

Accordingly, the implementation process consisted of five major steps.

- First, an agent building toolkit, the ZEUS agent development toolkit, is selected. The characteristics of the agent building toolkit are examined.
- Second, the role modelling of the MASCOT model is analysed;
- Third, the application design is conducted to resolve the problems identified at the role modelling stage. A major task is to identify the input information and how to transfer this information into the form required by the MASCOT model; and
- Fourth, the model is implemented through the ZEUS toolkit.

Although the MASCOT model provides a relatively sophisticated negotiation protocol and a number of negotiation strategies, this implementation does not intend to implement all the negotiation strategies due to the time limitation. The major interest is to demonstrate the key negotiation mechanisms (i.e. the integration of Zeuthen's concession mechanism and the Bayesian learning approach, and the involvement of the client agent). Other negotiation strategies such as the 'extended solution-searching' are not implemented.
6.2 IMPLEMENTATION ENVIRONMENT

The implementation environment includes both the hardware and software systems:

6.2.1 Hardware

Considering the general applications of the developed agent system, the MASCOT model is implemented on a personal computer (PC). Its main technical features are: a 550Mhz processor, a 120M RAM, a 13G hard disk and other facilities. The final application of the developed system will be applied within a group of computers which are connected through a network.

6.2.2 Software

In this study, the ZEUS agent building toolkit is selected as the implementation toolkit because it provides general and customisable agent development methodologies and software engineering tools to facilitate large-scale collaborative agent realisation. The ZEUS toolkit allows developers to configure agents of varying functionality and behaviour; organise the agents in whatever manner using system-provided organisational relationships; and imbue each agent with system-supplied or user-defined communicative and co-ordination mechanisms. In addition, it also provides predefined information discovery agents and extensive facilities for visualising and debugging societies of ZEUS agents. Furthermore, by the time when the MASCOT system was implemented, the ZEUS toolkit was one of the very few agent development toolkits which could be operated practically. For further information about other agent development toolkits, see ECOMAS (2002).

Besides the ZEUS toolkit, Java programming language is also used to build the external classes which are integrated with the ZEUS primitive tasks to fulfil the functions of the tasks. The details of these software systems are:
- ZEUS Agent Building Toolkit (1.04) is developed by British Telecommunications plc. (BT), and is downloaded from URL http://193.113.209.147/projects/agents/zeus/;

- Visual Java (6.0) is developed by Microsoft Company. Java JDK 1.2.2 is developed by Sun Company, and is available from URL http://www.java.sun.com/j2se/.

Meanwhile, computer CLASSPATH is also set according to the requirements of the ZEUS toolkit, Java and Java JDK. After this software is installed, the new ZEUS agent cases could be created by using the ZEUS Agent Generator (java.zeus.generator.AgentGenerator).

6.3 THE ZEUS AGENT BUILDING TOOLKIT

The notion of heterogeneous autonomous agents collaborating to solve problems is a powerful metaphor for the engineering of distributed and interoperable software systems. This agent-based approach introduces a new level of abstraction - of knowledge level cooperation between autonomous systems - that enhances distributed systems, interoperability, scalability and re-configurability (BT, 1999b). However, thus far, the promise of the agent approach has been largely unrealised due to a number of factors, especially because of the inherent complexity of constructing collaborative agent systems. Toolkits are needed to support the development of multi-agent systems.

The ZEUS toolkit has been developed by BT to facilitate large-scale realisation of the collaborative agent approach to distributed software engineering, frameworks, and methodologies. It is a culmination of a synthesis of established agent technologies to provide an integrated environment for the development of MAS. The toolkit develops a MAS design approach and supports it with a visual environment for capturing developer specification of agents that are used to generate the Java source code for the agents.

---

20 The following discussions about the ZEUS toolkit are mainly abstracted from BT (1999a-g). Thus, references are often not particularly listed.
6.3.1 Issues for Building of Collaborating Agent Systems

Collaboration between agents at the knowledge level places significant demands on agents. Not least is the need for a mechanism for information discovery through which agents discover the existence, network address, capabilities and/or roles of other agents; an agent-independent inter-agent communicating language that the agents use to communicate with one another; and an ontology that defines the application domain concepts being communicated between the agents. Furthermore, for effective and coherent problem solving, agents need mechanisms for reasoning about their own and other agents' problem solving capabilities and for co-ordinating their activities. In very dynamic environments, the problems are exacerbated by the additional requirements for data-driven reactive behaviour that integrates with the goal-driven deliberative activities of the agents. Finally, in some application domains, agent systems may need to interface with legacy systems such as databases.

The main techniques proposed by ZEUS to address these issues include information discovery, communication, ontology, co-ordination and legacy software problems. Figure 6.1 is a context diagram illustrating the interplay between the various issues and their associated solutions.

![Diagram](image)

Figure 6.1 General frame of the ZEUS agent building toolkit (BT, 1999)
The Information Discovery Issue: This is typically handled using special-purpose utility agents such as nameservers and facilitators that function as society-wide white pages and yellow pages, providing a look-up service for agents' addresses and abilities respectively. Thus, agents only need to register their address with a nameserver and their abilities with a facilitator to become visible to the society.

The Ontology Issue: Agents that communicate in a common language will still be unable to understand one another if they use different vocabularies for representing shared domain concepts. Therefore, they also need to use the same ontology or vocabularies of common concepts. This can be achieved either through general-purpose ontology or by creating domain-specific ontology and using inter-ontology translators to map between them.

The Co-ordination Issue: The main co-ordinating approaches of MAS can be broadly classified as organisational structuring, contracting, multi-agent planning, and negotiation (see Chapter 3). The ZEUS toolkit provides mechanisms to facilitate co-ordination between agents. For example, an auction protocol has been built for the Contracting co-ordination mechanism.

Integration with Legacy Software: As agents are not intended as replacements for legacy software, they must be able to interact with them. Generally, there are three possible approaches: the software could be rewritten, but this is a costly approach. Alternatively a separate piece of software called a transducer could be employed to act as an interpreter between the agent communication language and the native protocol of the legacy system. Or thirdly, the wrapper technique could be used to augment the legacy program with code that enables it to communicate using the inter-agent language.

6.3.2 Essential Functions of ZEUS

To provide a relatively general and customisable, collaborative agent building environment, the ZEUS toolkit encapsulates the following major principles:

Firstly, it delineates between domain-level problem solving and agent-level
functionality. The latter covers the application-independent multi-agent issues such as communication, co-ordination, task execution monitoring, and exception handling, whilst the former covers the acquisition, representation and use of domain-specific knowledge in problem solving. With the agent-level functionality provided, the developers could concentrate on implementing the domain-specific problem solving abilities of their agents.

- Secondly, use of the toolkit is based on the 'visual programming' paradigm. Hence the toolkit would support the agent creation process by providing structured menus and tables that would enable application developers to configure the functionality and modalities required of their agents as simply as possible.

- Thirdly, the toolkit supports an open design to ensure it is easily extensible. Thus, developers are able to easily add to the library of agent level components, and configure new agents using a combination of developer-defined and system-supplied components.

Based on these design philosophies, the ZEUS toolkit allows developers to:

- configure a number of different agents of varying functionality and behaviour;
- organise the agents using system-supplied organisational relationships;
- imbue each agent with selected system-supplied and/or developer-defined communicative and co-ordination mechanisms;
- supply each agent with the appropriate application-specific problem solving code; and
- automatically generate the executables for the agents.

In addition, the ZEUS toolkit also provides predefined information discovery agents such as nameserver, facilitator, and extensive facilities for visualising and debugging societies of ZEUS agents.
6.3.3 The ZEUS Toolkit Architecture

The ZEUS toolkit consists of a set of components, written in the Java programming language, that can be categorised into three functional groups: an agent component library, an agent building tool and a suite of utility agents comprising nameserver, facilitator and visualiser agents (Figure 6.2).

![Components of the ZEUS agent building toolkit](image)

**Figure 6.2 Components of the ZEUS agent building toolkit**

### 6.3.3.1 The Agent Component Library

The Agent Component Library is a collection of classes that form the building blocks of individual agents. Together these classes implement the application-independent agent-level functionality required of collaborative agents. The contents of this library address issues such as: communication, ontology, and co-ordination. For **communication** the Component Library provides:

- a performance-based agent communication language, in this case KQML;
- an asynchronous socket-based message passing system;
- an editor for describing domain-specific ontology - the domain concepts; and
- a frame-based knowledge representation language for representing domain concepts.
Next, for reasoning and multi-agent co-ordination, the Component Library provides:

- a general purpose planning and scheduling system suitable for typical task-oriented application domains, and the co-operative problem-solving inherent to these applications, and

- a co-ordination engine that controls the social behaviour of an agent (i.e. when and how it interacts with other agents and the types of contracts it sets up with them).

The functioning of the planner and co-ordination engine are influenced by the agent’s knowledge context (i.e. its available resources and competencies, its organisational relationships with other agents and its available co-operation strategies). Thus, to support these two components, the Component Library also provides:

- a library of predefined re-usable co-ordination protocols (e.g. contract-net and various auction protocols);

- a number of predefined organisational relationships (e.g. superior, subordinate, co-worker and peer relations); and

- knowledge representation mechanisms and databases for describing and storing the resources and competencies of an agent.

6.3.3.2 The Generic ZEUS Agent

Together, the components of the Agent Component Library enable the construction of an application-independent generic ZEUS agent that can be customised for specific applications by imbuing it with problem-specific resources, competencies, information, organisational relationships and co-ordination protocols. Figure 6.3 and Table 6.1 show the architecture of the generic ZEUS agent.
Figure 6.3 Architecture of the generic ZEUS agent

Table 6.1 The components of a generic ZEUS agent and their functions

<table>
<thead>
<tr>
<th>Component</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mailbox</td>
<td>Handles communications between the agent and other agents</td>
</tr>
<tr>
<td>Message Handler</td>
<td>Processes incoming messages from the Mailbox, dispatching them to the</td>
</tr>
<tr>
<td></td>
<td>relevant components of the agent.</td>
</tr>
<tr>
<td>Co-ordination Engine</td>
<td>Makes decisions concerning the agent's goals; co-ordinates the agent's</td>
</tr>
<tr>
<td></td>
<td>interactions with other agents using the co-ordination protocols and</td>
</tr>
<tr>
<td></td>
<td>strategies.</td>
</tr>
<tr>
<td>Acquaintance Database</td>
<td>Describes the agent's relationships with other agents in the society, and</td>
</tr>
<tr>
<td></td>
<td>its beliefs about the capabilities of those agents.</td>
</tr>
<tr>
<td>Planner &amp; Scheduler</td>
<td>Plans the agent's tasks based on decisions taken by the Co-ordination</td>
</tr>
<tr>
<td></td>
<td>Engine and the resources and task specifications available to the agent.</td>
</tr>
<tr>
<td>Resource Database</td>
<td>Maintains a list of resources that are owned by and available to the</td>
</tr>
<tr>
<td></td>
<td>agent. The Resource Database also supports a direct interface to external</td>
</tr>
<tr>
<td></td>
<td>systems.</td>
</tr>
<tr>
<td>Ontology Database</td>
<td>Stores the logical definition of each fact type - its legal attributes,</td>
</tr>
<tr>
<td></td>
<td>the range of legal values for each attribute, any constraints between</td>
</tr>
<tr>
<td></td>
<td>attribute values, and any relationships between the attributes of the</td>
</tr>
<tr>
<td></td>
<td>fact and other facts.</td>
</tr>
<tr>
<td>Task/Plan Database</td>
<td>Provides logical descriptions of planning operators (or tasks) known to</td>
</tr>
<tr>
<td></td>
<td>the agent.</td>
</tr>
<tr>
<td>Execution Monitor</td>
<td>Maintains the agent's internal clock, and starts, stops and monitors tasks</td>
</tr>
<tr>
<td></td>
<td>that have been scheduled for execution or termination by the Planner/Scheduler. It also informs the Planner of successful and exceptional terminating conditions of the tasks it is monitoring.</td>
</tr>
</tbody>
</table>
6.3.3.3 The ZEUS Agent Building Software

The principle underlying the ZEUS toolkit is that application-specific agents can be constructed by configuring the generic ZEUS agent, and equipping it with the necessary application functionality. To facilitate rapid development, the ZEUS toolkit provides a high-level agent development approach that hides the complexities of the Agent Component Library from the agent developer. This approach has two key aspects: an agent creation methodology, which guides the developer through the analysis and design of the intended system, and a visual agent development environment that supports the creation methodology. The first work is fulfilled through a ZEUS Agent Generator. The second task is fulfilled through a kind of specified utility agent - Visualiser agents.

The Agent Generator is an integrated suite of editors that guide developers through the stages of comprehensive agent development methodology. During this process developers describe the agents within their application, how they interact, and the tasks they perform. Amongst the tools provided are:

- An Ontology Editor for defining the ontology items in a domain. Fact objects are defined in terms of their attributes and the valid value ranges for each attribute.
- A Fact/Variable Editor for describing specific instances of facts and variables, using the templates created using the Ontology Editor.
- An Agent Definition Editor for describing agents logically. This involves specifying each agent's tasks, its initial resources, and the dimensions of its plan diary.
- A Task Description Editor for specifying the attributes of tasks and for graphically composing summary tasks.
- An Organisation Editor for defining the organisational relationships between agents, and agents' beliefs about the abilities of other agents.
- A Co-ordination Editor for selecting the set of co-ordination protocols with which each agent will be equipped.
6.3.3.4 The ZEUS Utility Agents

The ZEUS suite of utility agents consists of a nameserver and a facilitator agent that facilitate information discovery, and a visualiser agent for visualising or debugging societies of ZEUS agents. A ZEUS agent society may contain any number of these utility agents, with at least one nameserver agent.

- **Nameserver agents** have only a Mailbox and Message Handler, the components needed for receiving and responding to agents’ requests for the addresses of other agents. In addition, nameserver agents maintain a society-wide clock; thus, on initialisation, an agent registers with a nameserver and synchronises its internal clock to that of the nameserver.

- **Facilitator agents** have a Mailbox and Message Handler for receiving and responding to queries from agents about the abilities of other agents, and an Acquaintance Database for storing the abilities of the agents. They function by periodically querying all the agents in the society about their abilities, and storing the returned information in their Acquaintance Database. Also, individual agents might advertise their abilities to facilitators. Thus, when an agent wants to find other agents that have a particular competence, they can simply send an appropriate query message to a facilitator agent.

- **Visualiser agents** can be used to view, analyse or debug societies of ZEUS agents. They function by querying other agents about their states and processes, and then collating and interpreting the replies to create an up-to-date model of the agents’ collective behaviour. This model can be viewed from different perspectives through visualisation tools supported by the visualiser agents. The current tools include society viewer, reports tool, agent viewer, control tool, and statistics tool.

6.3.3.5 Integrating ZEUS Agents with External Programs

The ZEUS toolkit also provides primary interfaces between ZEUS agents and external programs. These include the domain functions in primitive plan operator specifications,
and developer-defined Co-ordination Engine graphs whose nodes make direct calls to external programs. For routine problem solving within the declared scope of the ZEUS toolkit, it is expected that for the most part, these primary mechanisms will suffice. However, the Agent Component Library also provides a secondary, more sophisticated interface, although employing it requires significant developer programming. This is done via a ZEUS external interface class and an agent internal event model.

The ZEUS external interface class allows developers to link an external Java class to an executing ZEUS agent program. Once linked to the agent program, the external code can utilise the agent’s public methods to query or modify the agent’s internal state. Thus, for example, the resource and/or plan databases can be queried or modified.

6.4 MASCOT MODEL IMPLEMENTATION

In common with most other structured development approaches, the ZEUS approach consists of analysis, design and realisation activities, as well as runtime support facilities that enable the developer to debug and analyse their implementations. Figure 6.4 illustrates the MASCOT agent development process.

Figure 6.4 The MASCOT agent development process
The MASCOT agent development contains the following four steps:

- **Domain Analysis:** The purpose of the initial analysis stage is to model and understand the application problem (e.g. agent responsibilities and roles). An approach recommended by ZEUS to fulfil this task is Role Modelling. Role modelling is relevant to most facets of the agent development lifecycle, addressing the specification, analysis, design, implementation, and maintenance of agents (BT, 1999a). Role models are also patterns of interaction, providing a readily comprehensible means of analysing the problem in question.

- **Agent Design:** This stage involves the translation of role responsibilities into the agent-level problems they represent, and deriving appropriate solutions (i.e. tasks, rulebases and interfaces). While the analysis process involved understanding the problem requirements, the design process involves expertise, knowing when and how to reuse and adapt existing proven solutions. Also, the knowledge (ontology) modelling process is conducted. This models the declarative knowledge that will be used by the agent roles. This stage results in the concepts inherent to the application, their attributes and possible values.

- **Agent Realisation:** The objective of this process is to realise working agent implementations from the conceptual designs created during the previous stage. The agent realisation process consists of several stages, which are closely coupled to the levels of abstraction that exist within a ZEUS agent. This stage is where ZEUS begins to offer software support, providing an Agent Generator tool through which the designs can be entered, and then used to generate Java source code for the agents.

- **Runtime Support:** The ZEUS approach does not end with the creation of the agents. There is also a suite of runtime support tools that are available through the Visualiser agent. This reflects the fact that the development process is unlikely to have ended with the implementation of the agents, as they still need to be tested, debugged and optimised.

The MASCOT implementation process is described in three parts: the MASCOT role modelling, the application design and model realisation.
6.4.1 Role Modelling

Agent roles and role models provide a vocabulary for describing agent systems, with each role describing a position and a set of responsibilities within a certain context or role model. This approach encourages developers to analyse the problem in terms of the roles that need to be played, and the responsibilities associated with each role. The role models formalise the definition of an agent role so that it can be modelled, designed, and implemented in software. The work of role modelling is effectively an agent-oriented extension of role modelling, as practised in conventional object-oriented software engineering. The qualifying criteria for roles are very similar to those of objects (i.e. the role should be modular; have high cohesion; be prudent; be complete; and have low coupling) (BT, 1999a).

The role models are grouped into domains. These domains have been chosen after analysing existing agent applications and grouping together those that address similar problems or exhibit similar behaviour. The domains provide a context that enables developers to compare their planned system with existing applications. Thus each section of the subject domain possesses several role models. Role models are 'architectural patterns' that depict the high-level similarities between related systems (i.e. the problems inherent to each domain).

6.4.1.1 The Components of a Role Model

Role model notation originates from the UML (Unified Modelling Language) class diagram notation. Each role model entry begins with a description summarising its main features and general applicability. The model's constituent roles are then shown in a Role Model diagram (Figure 6.5). In role diagrams, the key concepts are roles rather than classes, (represented by rectangles), whilst containment and inheritance are depicted in the same fashion as in class diagrams, with diamond and triangle headed lines respectively. The only difference between a class diagram and role diagram is that class
diagrams describe the static relationships between classes, while role models describe the dynamic interactions between roles. Hence the UML (Unified Modelling Language) class diagram has been augmented with additional notation to depict interactions: the arrowhead line. Where an arrowhead line is shown with a filled circle, this means that more than one simultaneous interaction can occur between entities playing these roles.

Figure 6.5 The notation used within a typical role diagram

After the role diagram, the next component of each role model entry is the collaboration diagram. This abstracts away from the specialisation and containment relationships between roles and instead concentrates on how they interact, an example is shown in Figure 6.6. The main difference between collaboration and role diagrams is that only interactions are shown. Each interaction is annotated with a number, which refers to an explanation of the interaction found later in the role model. Where the collaboration diagram is made easier to understand by the inclusion of sub-roles, these are shown inside their containing role.

Figure 6.6 The notation used within a collaboration diagram

The next part of a role model entry is the role description section. The purpose of this section is to describe each role in terms of its social obligations and application-specific functionality. Hence each of the interactions shown in the collaboration diagram will
appear in the corresponding role description. As well as interactions with other roles, each entry also describes the interactions between the role and its external interfaces.

6.4.1.2 MASCOT Role Modelling

By following the role modelling procedure described above, the MASCOT role models were developed (Figure 6.7). Of the eight roles illustrated in the figure, an important role - the Case-based mediator role, will be incorporated in future study. Considering the interaction with the mediator role, an Mfacilitator role is adopted instead of the common facilitator role provided by ZEUS. Implicit in this role model is the presence of the name server role and the visualiser role. Figure 6.8 illustrates the interactions between these roles where the inquirer, registrant and mediator roles are not presented. Table 6.2 describes the collaboration relationship.

![Figure 6.7 The MASCOT role representation](image-url)
Table 6.2 Interaction summary

<table>
<thead>
<tr>
<th>Collaboration</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Registration</td>
<td>Agents register or de-register their claim items for negotiation.</td>
</tr>
<tr>
<td>2 Offer</td>
<td>Message containing agent’s offer</td>
</tr>
<tr>
<td>3 Counteroffer</td>
<td>Another agent replies to a previously submitted offer</td>
</tr>
<tr>
<td>4 Report to client</td>
<td>Report to the client on the negotiation process</td>
</tr>
<tr>
<td>5 Participant in negotiation</td>
<td>The client may get involved in the negotiation</td>
</tr>
<tr>
<td>6 Inquiry</td>
<td>Agents inquire from the Mfacilitator about any possible solution to the claim.</td>
</tr>
<tr>
<td>7 Answer</td>
<td>The Mfacilitator answers the inquiries from agents</td>
</tr>
</tbody>
</table>

Note: 1) The initial interactions between agents as they register with a Name Server are not shown.
2) The interactions 6 and 7 are currently not implemented within MASCOT (shown as dot lines in Figure 6.8).

The following tables are the role description entries (Table 6.3 - 6.7). The purpose is to describe each role in terms of its social obligations and application-specific functionality. Furthermore, each entry also describes the interactions between the role and its external interfaces. The external interfaces represent the means through which the role performs its application-specific activities, such as accessing databases, or reading information from a developer interface.
### Table 6.3 Contractor role description

<table>
<thead>
<tr>
<th>Contractor</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Role Model:</strong> MASCOT</td>
</tr>
<tr>
<td><strong>Relationships to other roles:</strong> Contained by the Negotiator role</td>
</tr>
<tr>
<td><strong>Description:</strong> This is the role played by the contractor. No assumptions are made as to the expertise of the contractor.</td>
</tr>
<tr>
<td><strong>Responsibilities:</strong></td>
</tr>
<tr>
<td>[1] To inform the Facilitator of the new claims item</td>
</tr>
<tr>
<td>[2,3] To make and respond to offers from the Engineer</td>
</tr>
<tr>
<td>[4] To inform the Client the negotiation message</td>
</tr>
<tr>
<td><strong>External Interfaces:</strong></td>
</tr>
<tr>
<td>To facilitate the entry of its owner’s pre-negotiation information</td>
</tr>
<tr>
<td>To interpret the negotiation</td>
</tr>
<tr>
<td>To facilitate further involvement of its owner</td>
</tr>
<tr>
<td><strong>Prerequisites:</strong></td>
</tr>
<tr>
<td>The specified negotiation protocol and strategies of the MASCOT model</td>
</tr>
</tbody>
</table>

### Table 6.4 Engineer role description

<table>
<thead>
<tr>
<th>Engineer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Role Model:</strong> MASCOT</td>
</tr>
<tr>
<td><strong>Relationships to other Roles:</strong> Contained by the Negotiator role</td>
</tr>
<tr>
<td><strong>Description:</strong> This is the role played by the engineer. No assumptions are made as to the expertise of the engineer.</td>
</tr>
<tr>
<td><strong>Responsibilities:</strong></td>
</tr>
<tr>
<td>[2,3] To receive and respond to offers from the Contractor</td>
</tr>
<tr>
<td>[4,5] To inform the client of the negotiation status, and receive instructions from the client</td>
</tr>
<tr>
<td><strong>External Interfaces:</strong></td>
</tr>
<tr>
<td>To facilitate the entry of its owner’s pre-negotiation information</td>
</tr>
<tr>
<td>To interpret the negotiation</td>
</tr>
<tr>
<td>To facilitate further involvement of its owner</td>
</tr>
<tr>
<td><strong>Prerequisites:</strong></td>
</tr>
<tr>
<td>The specified negotiation protocol and strategies of the MASCOT model</td>
</tr>
</tbody>
</table>
Table 6.5 Negotiator role description

<table>
<thead>
<tr>
<th>NEGOTIATOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role Model: MASCOT</td>
</tr>
<tr>
<td>Relationships to other Roles: specialist Task Agent, contains Contractor and Engineer</td>
</tr>
<tr>
<td>Implementation: Modified Task Agent</td>
</tr>
</tbody>
</table>

Table 6.6 Client role description

<table>
<thead>
<tr>
<th>CLIENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role Model: MASCOT</td>
</tr>
<tr>
<td>Relationships to other Roles: Contained by the Supervisor role</td>
</tr>
<tr>
<td>Description: This is the role played by the client. No assumptions are made as to the expertise of the client.</td>
</tr>
<tr>
<td>Responsibilities:</td>
</tr>
<tr>
<td>[4, 5] To receive information from the contractor and the engineer, and make suggestions to the engineer</td>
</tr>
<tr>
<td>External Interfaces:</td>
</tr>
<tr>
<td>To facilitate the entry of its owner’s pre-negotiation information</td>
</tr>
<tr>
<td>To interpret the negotiation</td>
</tr>
<tr>
<td>To facilitate further involvement of its owner</td>
</tr>
<tr>
<td>Prerequisites:</td>
</tr>
<tr>
<td>The specified negotiation protocol and strategies of the MASCOT model</td>
</tr>
</tbody>
</table>

Table 6.7 Mfacilitator role description

<table>
<thead>
<tr>
<th>MFACILITATOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationships to other Roles: contains Case-based reasoning mediator</td>
</tr>
<tr>
<td>Description: This is somewhat similar to the standard ZEUS Application Facilitator role (if not considering the case-based reasoning role, it can be fulfilled by a standard ZEUS Facilitator role). Currently, it mainly stores all the information about claim items and each agent’s negotiation history.</td>
</tr>
<tr>
<td>Responsibilities:</td>
</tr>
<tr>
<td>[6] To receive notifications and inquiries from participants</td>
</tr>
<tr>
<td>[7] To reply to agents’ inquiries</td>
</tr>
<tr>
<td>External Interfaces:</td>
</tr>
<tr>
<td>To store information on negotiation participants</td>
</tr>
<tr>
<td>Prerequisites: The specified negotiation protocol and strategies of the MASCOT model</td>
</tr>
</tbody>
</table>
After defining all the roles in the MASCOT model, the next task is to allocate these roles to agents. A simple way is to assign an agent to each role, but that may not be an optimum way in most cases. The ZEUS role modelling guide (BT, 1999a) provides two principles in determining how to make appropriate agents.

- The first is derived from the fact that agents should be autonomous (i.e. be responsible for the control of resources and provision of services). This area of control is known as the agent’s Sphere of Responsibility. Thus when considering what agents will exist, it is necessary to consider how the application domain will be partitioned. A simple rule is “Each sphere of responsibility should possess a single agent”. It is also worth noting another factor that may help identify candidate agents. Agents tend to be responsive (i.e. able to perceive their environment and respond accordingly to events that affect their own sphere of responsibility).

- The second is The Point of Interaction, which is related to the Sphere of Responsibility test, but extends it by considering the social dimension of agents. Agents are often distinguished from other software systems by their ability to interact intelligently and constructively with other agents and people. Hence in an agent application, resources and services may not be directly accessible, but invoked by requesting the agent responsible for their control. A Point of Interaction test, which illustrates the difference between resources and agents: agents affect resources, interactions affect agents, illustrates another rule for agent creation: “The access point for information, expertise and services is a good agent candidate”.

Thus, individual roles do not necessarily need to be played by individual agents. It is more likely that several roles will be combined and be performed by a single agent. Roles may combine in various ways: the behaviour may be just added together; some behaviour may override other behaviour; or the behaviour might be combined synergistically.

Considering these two principles (i.e. Sphere of Responsibility and The Point of Interaction), this study creates several agents to fulfil the roles identified in the role
model (Table 6.8). Each role described in the above tables is assigned to an agent. Two more utility agents provided by the ZEUS toolkit are also included.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Roles Played</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contractor Agent</td>
<td>Negotiator (offer, counteroffer, registrant, inquiry)</td>
</tr>
<tr>
<td>Engineer Agent</td>
<td>Negotiator (offer, counteroffer, registrant, inquiry)</td>
</tr>
<tr>
<td>Client Agent</td>
<td>Supervisor (supervision if necessary, registrant, inquiry)</td>
</tr>
<tr>
<td>Mfacilitator Agent</td>
<td>Mfacilitator (currently: register claim items and record negotiation history; future: case-based mediator)</td>
</tr>
<tr>
<td>Visual Agent</td>
<td>Visualiser</td>
</tr>
<tr>
<td>ANS</td>
<td>Agent name server</td>
</tr>
</tbody>
</table>

Table 6.8 Agents created in MASCOT model

Having identified what roles and agents should exist within the application, the next step is to determine how agents will realise each role. This process needs two steps: application design and model realisation.

### 6.4.2 Application Design

The application design is a process of refinement, mapping each of the responsibilities identified in the previous stage to a generalised problem, and then choosing the most appropriate solution. In many cases, the solutions can be found in the existing ZEUS toolkit, such as utility agents, and relationship, especially co-ordination protocols. However, no ZEUS co-ordination protocol and strategies can fulfil the agents' responsibilities in the MASCOT model. Thus, the focus of this application design is on:

- identifying the negotiation protocol and strategies;
- identifying the input information (interface); and
- identifying how the input information is transferred into the form which the MASCOT model requires.

Another important aspect of the application design, knowledge modelling (ontology creation), will be discussed in the next section - realisation of the model.
6.4.2.1 Negotiation Protocol and Strategies

The developed MASCOT negotiation protocol is illustrated in Figure 6.9. A detailed description of the protocol can be seen in Chapter 5.

![Flowchart of the MASCOT negotiation protocol](image)

Figure 6.9 The flowchart of the MASCOT negotiation protocol
The existing negotiation protocol and strategies provided by the ZEUS toolkit cannot fulfil the above complex tasks, and therefore will not be adopted in this prototype. In the MASCOT model, the contractor and the engineer are required to make a decision with regard to the negotiation strategies at two stages: the criteria for the acceptance of the opponent’s proposal, and the decision to determine who should concede and how much the agent should concede (Figure 6.9). This is achieved by Zeuthen’s strategy discussed in Chapter 5. The strategy for the client agent is to decide whether it should join the negotiation between the contractor agent and the engineer agent, (i.e. whether it should use its reservation value to substitute the engineer agent’s reservation value), thereby improving the chances of an agreement being reached.

6.4.2.2 Input Information

There are two essential principles underlying the decision on agents’ input information: to address all the key factors of claims negotiation defined in previous studies; and to simplify the implementation system to ensure that only the key factors are involved. Seven pieces of input information are selected for the contractor and the engineer agents. These are: ‘claims items’, ‘my reservation value’, ‘my optimum value’, ‘time allowed for negotiation’, ‘the opponent’s possible reservation value’, ‘the opponent’s negotiation habits’, and the ‘confidence in my estimate of the opponent information’. Since the client agent does not play a direct negotiator’s role, it is not necessary for it to estimate the contractor’s reservation value and negotiation habits. Thus, there are five pieces of input information involved in the client input window (Figure 6.10).

Figure 6.10 The MASCOT model input window

207
• Claims Items

When a party starts a negotiation, he first needs to identify the negotiation item (i.e. the ID of the claim item) such as labour cost claim, material cost claim or loss of productivity claim. If both parties initiate different negotiation items, the Mfacilitator agent will remind them of the difference, and arrange a negotiation for the same item. Moreover, the identification of the ID of a claim (negotiation) item is also essential for agents to conduct any of the extended solution searching strategies, where agents make trade-offs between different claim items; or ask their owners to relax some negotiation constraints, so that the opportunity to reach an agreement is increased.

• My Reservation Value and My Optimum Value

These two values represent the maximum or minimum value a party can offer to or accept from the opponent. For the contractor, reservation value is the minimum value that s/he could accept for the claim item; optimum value represents the possible maximum value that the contractor expects from the engineer. For the engineer or the client, reservation value is the maximum value that they could offer to the contractor; the optimum value represents the minimum value that they would prefer to offer for the claim item. In this study, these two pieces of information fully define a party’s utility function in negotiation. It is private information. It is assumed that every party clearly understands his/her reservation value and optimum value.

• Time Allowed for Negotiation

This value defines the deadline within which a party is willing to end a negotiation. In this study, the deadline is represented as a different time unit, from one to five. If a party wishes to finish the negotiation within a very short time period, s/he should input one (time unit). Otherwise, he may choose an appropriate time unit if he has enough time to
handle the negotiation. Based on the input time unit, the party’s utility will be reduced by a certain percentage for every iteration during the negotiation (for details, see Ren, 2001).

- **The Opponent’s Possible Reservation Value**

This value represents a party’s estimate of the opponent’s reservation value. In this study, the opponent’s reservation value is not represented by the single input reservation value. Rather, it is described by a group of reservation values with specific probability distribution determined by the estimated reservation value and the party’s confidence about the estimate. The probability distribution of the reservation values is hypothesised as the normal distribution.

The normal distribution frequently occurs in practical problems, and is most commonly used to describe the events which are possibly distributed in symmetrically in the real world. Importantly, the distribution provides an accurate approximation to a large number of other probability laws (Scheaffer and Mcclave, 1999). Its characteristics, symmetry and bell shape, reflect the distribution characteristics of one party’s estimate of the opponent’s reservation value.

The general formula for the probability density function of the normal distribution is expressed as:

$$f(x) = \frac{e^{-(x-\mu)^2/(2\sigma^2)}}{\sigma \sqrt{2\pi}}$$

Where, $\mu$ is the mean of the distribution (location parameter) which shifts the location of the normal distribution curve along the horizontal axis; and $\sigma^2$ is the variance of the distribution (scale parameter) which determines the shape of the curve. The effect of a scale parameter greater than one is to stretch the curve while the curve is compressed if a scale parameter is less than one. Figure 6.11 is the plot of normal probability density functions with different $\sigma$ values.
In this study, mean $\mu$ represents the input estimated opponent's reservation value. Standard deviation $\sigma$ is determined by the negotiating party's confidence about his estimate. It is assumed that if the party's confidence is high, $\sigma$ is one; if the confidence is low, $\sigma$ will be three (Table 6.9). The shape of the curve is very plain in this case (a straight line means that a party does not have any pre-knowledge about the opponent's possible reservation value).

**Table 6.9 Attribute assumptions of normal distribution**

<table>
<thead>
<tr>
<th>Confidence</th>
<th>High</th>
<th>Medium</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma$</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

- **The Opponent's Negotiation habits**

This value is a party's estimate of the opponent's negotiation habits. In the contractor's input window, it is expressed as "The engineer's offer is usually lower than his real value by ___ percentage". The relationship between the engineer's reservation value and his possible offers is illustrated in Figure 6.12. If the engineer has a reservation value $R_1$, he will most probably make an offer that is around $O_1$. The input negotiation habit represents the ratio $O_1 / R_1$. Such a distribution is normally asymmetric because the engineer does not have much interest in those offers which are much lower than $O_1$ since
they can seldom be accepted by the contractor, and vice versa. Such an asymmetric distribution can be figured out either through an industry survey, or by a mathematical probability distribution. This study adopts the mathematical probability approach. It is hypothesised that lognormal distribution can illustrate such an asymmetric distribution²¹.

The lognormal distribution may be defined as the distribution of a variate whose logarithm obeys the normal law of probability (Aitchison and Brown, 1963). Many applications of the lognormal distribution have been noted in nature from a variety of fields such as: small particle statistics, biology, anthropometry, astronomy, philosophy, physical and industrial processes, economics, and sociology. Two typical application cases are distribution of incomes and analysis of consumer behaviour.

The density function of lognormal distribution is described as:

\[
\frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{\ln(x) - \mu}{\sigma} \right)^2}
\]

Figure 6.12 Relationship between the engineer's reservation value and the possible offers

²¹ This hypothesis needs to be further proved theoretically and verified in various cases. However, it is beyond the scope of this study to make such a comprehensive analysis. The limited objective of this study is to discuss the lognormal distribution as a candidate for the mathematical description of the relationship between negotiation participants' reservation value and the possible offers, especially the asymmetric distribution. Although some other distributions such as the Gamma distribution and the Weibull distribution are also characterised as asymmetric distributions, the physical meaning of these distributions are not suitable for this application scenario (For details, see Aitchison and Brown, 1963).
\[ f(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2} (\log x - \mu)^2\right) \]

Where, \( \mu \) and \( \sigma \) are the mean and standard deviation of the underlying normal distribution as described in the normal distribution. The impacts of these two factors on the lognormal distribution are shown in Figure 6.13 and 6.14.

Figure 6.13 Frequency curves of the lognormal distribution for three value of \( \mu \)

Figure 6.14 Frequency curves of the lognormal distribution for three value of \( \sigma \)
By analysing the influence of different attributes on the shape of the lognormal distribution, this study makes a number of assumptions (Table 6.10). Since it is difficult to make a quantitative analysis of the assumptions, the idea is to control the shape of the curve through the attributes. For example, if a party has high confidence in his estimate, the shape of the curve will be relatively sharp; meanwhile, the relationship between mode, median and mean of the distribution is also considered.

Table 6.10 Attribute assumptions of lognormal distribution

<table>
<thead>
<tr>
<th>Confidence</th>
<th>μ</th>
<th>σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>0.4</td>
<td>0.9</td>
</tr>
<tr>
<td>Medium</td>
<td>0.7</td>
<td>0.8</td>
</tr>
<tr>
<td>Low</td>
<td>1.0</td>
<td>0.7</td>
</tr>
</tbody>
</table>

- Confidence

Confidence describes the degree of a negotiation party's understanding of the opponent's reservation value and negotiation habit. Consequently, it influences the probability distribution of a party's estimate of his opponent's reservation value and the conditional probability distribution of the opponent's possible offers given an estimated reservation value. Confidence is described by using three qualitative values: high, medium, or low. When a party has little domain knowledge about the opponent, his confidence is assumed to be low. On the contrary, if he has much negotiation experience with the opponent, it is reasonable to assume that the confidence of his estimate is high. Otherwise, it is proper to define the confidence as medium.

6.4.2.3 Determination of the Probability Distribution

After obtaining the input information, the next step is to transfer the input information into the probability distributions required for the application of the Bayesian learning approach (for details see Table 3 in Ren, 2001). The following example illustrates this process.
Suppose the contractor's input information about the engineer is:

- The engineer's possible reservation value is: 950
- The engineer's offer usually is lower than his real offer by: 0.20 (i.e. 20%)
- The confidence about the above estimate is: high

Based on this information, the distribution of the engineer's reservation value can be worked out using the normal distribution. To ensure accuracy, eleven possible reservation values are included based on the estimated reservation value. The centre of the distribution is decided by the estimated reservation value (950). The range of these data is determined by the contractor's estimate of the engineer's negotiation habit (0.2). The shape of the distribution is determined by the contractor's confidence (high) (Table 6.11).

**Table 6.11 Reservation distribution**

<table>
<thead>
<tr>
<th>Possible Reservation Value</th>
<th>760</th>
<th>798</th>
<th>836</th>
<th>874</th>
<th>912</th>
<th>950</th>
<th>988</th>
<th>1026</th>
<th>1064</th>
<th>1102</th>
<th>1140</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probability</td>
<td>0.022</td>
<td>0.044</td>
<td>0.078</td>
<td>0.116</td>
<td>0.147</td>
<td>0.160</td>
<td>0.147</td>
<td>0.116</td>
<td>0.078</td>
<td>0.044</td>
<td>0.022</td>
</tr>
</tbody>
</table>

The following work is to determine the conditional probability of the engineer's offers given the possible reservation values. This process includes three steps:

- Firstly, the probability distribution is calculated based on the lognormal distribution, which, in turn, is determined by the contractor's confidence about his estimate. To ensure a reasonable accuracy, seven data are selected to plot the probability distribution (Table 6.12).

**Table 6.12 Lognormal distribution**

<table>
<thead>
<tr>
<th>Probability Distribution</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.152</td>
<td>0.174</td>
<td>0.143</td>
<td>0.110</td>
<td>0.084</td>
<td>0.064</td>
<td>0.050</td>
</tr>
</tbody>
</table>

- Secondly, the engineer's possible offers are determined based on the range of the reservation values in Table 6.13 and the estimated engineer's negotiation habit. Since the engineer and the contractor have different tendencies in making offers, the range of their offers are quite different even if their reservation value distributions are...
similar. The first row in Table 6.13 represents the engineer's possible offers in this example. 23 possible offers are selected to describe the conditional probability.

- Thirdly, the conditional probabilities of the engineer's offers given the reservation values are determined. For each possible reservation value, there is a distribution of possible offers that is described by a lognormal distribution. For example, if the engineer's reservation value is 950 and his negotiation habit is 20% (i.e. the engineer usually makes an offer lower than the reservation value by 20%), then the engineer's most possible offer is: 950x80%=760. Thus, the maximum probability of 0.174 is assigned to an offer of 760. Consequently, the other probabilities are assigned to the other possible offers sequentially. The result is shown in the seventh row (highlighted) of Table 6.14. Similarly, the conditional probabilities for each reservation value are worked out as shown in Table 6.14.

In the above analysis, it is implicitly assumed that the engineer will always prefer to make his offer lower than the reservation value by the same percentage (e.g. 20%) during the whole negotiation process. The assumption may work well in some cases depending on the individual negotiation mechanisms. For example, Zeng and Sycara (1998) adopt a similar assumption in their project - BAZZAR. In their system, a simple concession mechanism is adopted (e.g. a buyer will always make his offer 10% lower than the seller's reservation value estimated by him) that has been proved successful. The problems, such as who should make the next concession or how much the concession should be, are not considered.

However, there is a potential problem in this assumption. In practice, the engineer may make his offer 20% lower than his reservation value at the beginning of the negotiation, and reduce it to 10% at the final stage of the negotiation after both parties have made some concessions. His offer should be close to his expected offer amount (but not necessarily the reservation value) at this stage. Since the MASCOT model aims to build a relatively sophisticated negotiation mechanism (e.g. it involves the problem such as who should make concessions and how much concession should be), it is more reasonable to consider the change of the negotiators' exaggerated amount.
Table 6.13 Conditional probability of the engineer’s offer given the reservation value

| (b)  | 494 | 532 | 570 | 608 | 646 | 684 | 722 | 760 | 798 | 836 | 874 | 912 | 950 | 988 | 1026 | 1064 | 1102 | 1140 | 1178 | 1216 | 1254 | 1292 | 1330 |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 760  |     | .152| .174| .143| .110| .084| .064| .050|
| 798  |     | .152| .174| .143| .110| .084| .064| .050|
| 836  |     | .152| .174| .143| .110| .084| .064| .050|
| 874  |     | .152| .174| .143| .110| .084| .064| .050|
| 912  |     | .152| .174| .143| .110| .084| .064| .050|
| 950  |     |     | .152| .174| .143| .110| .084| .064| .050|
| 988  |     | .152| .174| .143| .110| .084| .064| .050|
| 1026 |     |     | .152| .174| .143| .110| .084| .064| .050|
| 1064 |     |     |     | .152| .174| .143| .110| .084| .064| .050|
| 1102 |     |     |     |     | .152| .174| .143| .110| .084| .064| .050|
| 1140 |     |     |     |     |     | .152| .174| .143| .110| .084| .064| .050|

Note: (a)-Reservation Value; (b)- Possible Offer; the data in all the empty cells are zero.
This can be achieved by adopting different negotiation habits (exaggerated amounts) in different reservation values. For example, in the above example, the engineer's exaggerated amount may change from 20% to 15% when the reservation value increases from 760 to 1140. The modified conditional probability distribution is shown in Table 6.14. The implementation of the MASCOT model has shown that such an improvement is very effective. However, the estimate of the change of the exaggerated amount (e.g. from 20% to 15%) is relatively sensitive and difficult.

6.4.3 Model Realisation

This section describes aspects of the application design (i.e. the knowledge modelling) and how the developed design is realised by using the ZEUS toolkit, in the form of the Agent Generator editor. The realisation process combines the steps necessary to create a generic ZEUS agent with the steps necessary to implement the role-specific solutions identified during the previous phases. The realisation process consists of the following activities:

- ontology creation
- agent creation, for each task agent this consists of:
  a) agent definition
  b) task description
  c) agent organisation
  d) agent co-ordination
- utility agent configuration
- task agent configuration.

6.4.3.1 Ontology Creation

An ontology is declarative knowledge representing the significant concepts within the application domain. A concept is significant and must be modelled if meaningful
interactions cannot occur between agents without both parties being aware of the concept. An ontology contains (BT, 1999d):

- the key concepts within the application domain;
- the significant attributes of each concept;
- the types of the each attribute, any constraints on the attributes; and
- initial values of these attributes.

The ZEUS toolkit provides two kinds of ontology: abstract and entity. The former defines the abstract concepts which are apart from physical items, such as: time, money and name. The latter defines the physical entities, such as: beam, column and floor. In the MASCOT model, all the concepts (also termed as ‘Facts’), such as reservation value, probability and agent name, refer to abstract instances rather than physical ones, thus the ontology of the MASCOT model is based on the abstract facts. Table 6.15 describes the ontology defined in the MASCOT model to fulfil each agent’s roles (the initial value of each attribute is not listed; for details see appendix 1). Basically, the ontology can be further classified as two kinds: those concepts that describe the key facts and data in negotiation, such as claimItem, agentName and offer; and those which work as trigger facts for rulebases, such as startMakeInitialoffer, startMakeNewOffer, and startResponseToOffer. Figure 6.15 illustrates how a fact is created through the ontology editor window.
<table>
<thead>
<tr>
<th>Facts</th>
<th>Description</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>claims</td>
<td>The claim items</td>
<td>lossofproductivity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>timeextension</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lumpsumitem</td>
</tr>
<tr>
<td></td>
<td></td>
<td>labourcost</td>
</tr>
<tr>
<td></td>
<td></td>
<td>material</td>
</tr>
<tr>
<td></td>
<td></td>
<td>equipment</td>
</tr>
<tr>
<td>constraints</td>
<td>Constraints contain some of the essential</td>
<td>hasBeenSent</td>
</tr>
<tr>
<td></td>
<td>information which needs to be specified by an</td>
<td>initialOfferDone</td>
</tr>
<tr>
<td></td>
<td>agent's owner before the negotiation can proceed,</td>
<td>opReserv</td>
</tr>
<tr>
<td></td>
<td>such as agent's reservation value, optimum value</td>
<td>opDev</td>
</tr>
<tr>
<td></td>
<td>and time constraints.</td>
<td>domainknowledge</td>
</tr>
<tr>
<td></td>
<td></td>
<td>confidence</td>
</tr>
<tr>
<td></td>
<td></td>
<td>myReserv</td>
</tr>
<tr>
<td></td>
<td></td>
<td>myOptv</td>
</tr>
<tr>
<td></td>
<td></td>
<td>time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>opOptv</td>
</tr>
<tr>
<td></td>
<td></td>
<td>owner</td>
</tr>
<tr>
<td></td>
<td></td>
<td>transferConditionalProbabilityDone</td>
</tr>
<tr>
<td>agentsName</td>
<td>Each agent needs a specified name</td>
<td>name</td>
</tr>
<tr>
<td>flags</td>
<td>A set of flags which will allow certain</td>
<td>newOfferDone</td>
</tr>
<tr>
<td></td>
<td>rules to be called or inhibited according to the</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>combinations set.</td>
<td>negotiationInProgress</td>
</tr>
<tr>
<td></td>
<td></td>
<td>stop</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moveToNextExchange</td>
</tr>
<tr>
<td></td>
<td></td>
<td>registerDone</td>
</tr>
<tr>
<td>formerOfferHistory</td>
<td>A record of who accepts a particular</td>
<td>client</td>
</tr>
<tr>
<td></td>
<td>negotiation item.</td>
<td>engineer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>contractor</td>
</tr>
<tr>
<td>Facts</td>
<td>Description</td>
<td>Attributes</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>negotiationOrder</td>
<td>A feature describing negotiation sequence</td>
<td>previous, next</td>
</tr>
<tr>
<td>acceptProposal</td>
<td>A feature describing whether the opponent’s offer has been accepted</td>
<td>name, accept</td>
</tr>
<tr>
<td>initialOffer</td>
<td>An agent’s initial offer</td>
<td>initialOffer</td>
</tr>
<tr>
<td>negotiationParticipants</td>
<td>The participants of a negotiation</td>
<td>respondent, initiator</td>
</tr>
<tr>
<td>startAssignOptimumValue</td>
<td>Trigger to assign the opponent’s optimum value after receiving his initial offer.</td>
<td>start</td>
</tr>
<tr>
<td>startAssignOptimumValue1</td>
<td>Trigger to assign the opponent’s optimum value after receiving his initial offer.</td>
<td>start</td>
</tr>
<tr>
<td>startMakeInitialOffer</td>
<td>Trigger to start making initial offer</td>
<td>start</td>
</tr>
<tr>
<td>startStartNegotiation</td>
<td>Trigger to start the negotiation</td>
<td>start</td>
</tr>
<tr>
<td>startRespondToProposal</td>
<td>Trigger to start making response to the opponent’s offer/ counteroffer</td>
<td>start</td>
</tr>
<tr>
<td>startSendAcceptNotice</td>
<td>Trigger to send the accept notice</td>
<td>start</td>
</tr>
<tr>
<td>startMakeNewOffer</td>
<td>Trigger to make new offer/ counteroffer</td>
<td>start</td>
</tr>
<tr>
<td>startSendProposalBack</td>
<td>Trigger to send proposal back to the opponent</td>
<td>start</td>
</tr>
<tr>
<td>startSendProposal</td>
<td>Trigger to send offer to the opponent</td>
<td>start</td>
</tr>
<tr>
<td>startChangeNegotiationParti cipants</td>
<td>Trigger to start changing the negotiation participants (for future use)</td>
<td>respondent, initiator</td>
</tr>
<tr>
<td>startInvolvementOfClient</td>
<td>Trigger for the involvement of the client</td>
<td>start</td>
</tr>
<tr>
<td>startTransferToConditionalP robability</td>
<td>Trigger to start transferring the owner input data into conditional probability</td>
<td>start</td>
</tr>
<tr>
<td>startRegister</td>
<td>Trigger to register to the facilitator</td>
<td>start</td>
</tr>
<tr>
<td>startEvaluateProposal</td>
<td>Trigger to evaluate the opponent’s offer/counteroffer</td>
<td>start</td>
</tr>
<tr>
<td>startAssignValue</td>
<td>Trigger to assign value to an attribute to draw the negotiation process</td>
<td>start</td>
</tr>
<tr>
<td>Facts</td>
<td>Description</td>
<td>Attributes</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>domainKnowledge</td>
<td>The estimate of the opponent's domain knowledge (i.e. the opponent's negotiation habits)</td>
<td>domainknowledge</td>
</tr>
<tr>
<td>offer</td>
<td>The offer and counteroffer value.</td>
<td>status, owner, toBeEvaluatedBy, offer</td>
</tr>
<tr>
<td>reservationValue</td>
<td>The estimated opponent's reservation value in each iteration.</td>
<td>reservationValue, agName, status, owner</td>
</tr>
<tr>
<td>hypothesis</td>
<td>An array recording the opponent's possible reservation values.</td>
<td>hyp, owner</td>
</tr>
<tr>
<td>probability</td>
<td>An array recording the possibility related to the above hypotheses.</td>
<td>probability, owner</td>
</tr>
<tr>
<td>proposal</td>
<td>Values are used for agents to plot the negotiation process.</td>
<td>status, owner, proposal, toBeEvaluatedBy, oppproposal, myreserv, myoptv, opreserv, opoptv</td>
</tr>
<tr>
<td>optimumValue</td>
<td>A value recording the opponent's optimum value</td>
<td>status, owner</td>
</tr>
<tr>
<td>pevent</td>
<td>An array recording the possible events</td>
<td>owner, pevent</td>
</tr>
<tr>
<td>cprosl</td>
<td>An array recording the current offer's conditional probability</td>
<td>owner, cprosl</td>
</tr>
<tr>
<td>mem</td>
<td>A fact recording all the negotiation offers and counteroffers</td>
<td>owner, mem, mof</td>
</tr>
<tr>
<td>agreement</td>
<td>The value of final agreement</td>
<td>agreement</td>
</tr>
</tbody>
</table>
6.4.3.2 Agent Creation

During the agent creation stage, the generic ZEUS agent is configured to fulfil its application-specific responsibilities. Thus, before creating the agents, the following decisions should have already been made in the role modelling and application design stages:

- What agents exist?
- What activities will each agent perform?
- How will each agent interact with other agents? and
- What strategies should each agent adopt?

The next step is to create these agents using the ZEUS toolkit, which consists of three main activities. They are:

- Configuring planning parameters: in this study, the parameters (e.g. maximum number of simultaneous and tasks planner length) are set as default (i.e. defined by the ZEUS toolkit);
• Task identification: the new (or defined) primitive tasks and rulebases are assigned to each agent, which will perform these tasks according to the role modelling; and

• Initial agent resources allocation: the resources (defined in the ontology) that agents would possess when they are initialised are listed as the agents’ initial resources. However, it is not necessary to define any facts which will be generated by tasks during the negotiation process. In the MASCOT model, there are three kinds of value that need to be defined in the initial attributes:
  a) Name strings, such as \textit{agentsName}, are set as related agent’s name, or none if the agent does not exist at the beginning;
  b) Boolean values, such as \textit{startMakeNewOffer} or \textit{startSendAcceptNotice} are all set as false;
  c) Real values, such as \textit{domainKnowledge}, are set as zero.

Figure 6.16 illustrates how an agent is created through the agent editor window. Table 6.16 lists agents’ primitive tasks, rulebases and initial resources defined in MASCOT.

![Figure 6.16 Agent definition window in the ZEUS toolkit](image-url)
Table 6.16 Tasks, rulebases and initial resources related to agents in MASCOT

<table>
<thead>
<tr>
<th>Agent</th>
<th>Primitive Tasks</th>
<th>Rulebases</th>
<th>Initial Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contractor</td>
<td>TransferToConditionalP</td>
<td>Register</td>
<td>agentsName</td>
</tr>
<tr>
<td></td>
<td>robability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MakeInitialOffer</td>
<td>StartTransferToConditi</td>
<td></td>
<td>startStartNegotiation</td>
</tr>
<tr>
<td></td>
<td>onalProbability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MakeNewOffer</td>
<td>StartMakeInitialOffer</td>
<td></td>
<td>flags</td>
</tr>
<tr>
<td>AssignOptimumValue</td>
<td>StartMakeInitialOffer</td>
<td></td>
<td>negotiationOrder</td>
</tr>
<tr>
<td>AssignOptimumValue1</td>
<td>StartMakeInitialOffer</td>
<td></td>
<td>acceptProposal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Engineer</td>
<td>TransferToConditionalP</td>
<td>Register</td>
<td>(same as the contractor)</td>
</tr>
<tr>
<td></td>
<td>robability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MakeInitialOffer</td>
<td>StartTransferToConditi</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>onalProbability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MakeNewOffer</td>
<td>StartMakeInitialOffer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AssignOptimumValue</td>
<td>StartNegotiation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Client</td>
<td>AssignValue</td>
<td>Register</td>
<td>agentsName</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>flags</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>startInvolvementOfClient</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>acceptProposal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>startSendAcceptNotice</td>
</tr>
<tr>
<td>Mfacilitator</td>
<td>InterfaceEndRB</td>
<td>agentsName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>formerOfferHistory</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NegotiationParticipants</td>
</tr>
</tbody>
</table>
Besides the above elements, another two aspects, agent organisation and agent co-ordination, also need to be defined through the agent editor window.

### 6.4.3.3 Agent Organisation

By default, agents are ignorant of the names and abilities of their neighbours, so if an agent needs the service of another, it will need to contact a directory service. However, agents may have prior knowledge of other agents, especially if they interact with each other on a regular basis. These known agents are called 'acquaintances'. There are four different types of relationships that can exist between agents:

- **Peer**: The default relationship with no assumptions about agent interaction;
- **Superior**: The acquaintance possesses higher authority than this agent, and can issue orders that this agent must obey;
- **Subordinate**: The acquaintance has less authority than this agent, and can be issued orders that it must obey; and
- **Co-worker**: The acquaintance belongs to the same 'community' as this agent, and will be asked before peers when any resources are required.

In the MASCOT model, the relationship between the contractor agent and the engineer agent (or the client agent) is peer to peer because they have equal position in terms of the claim issues. On the other hand, the relationship between the engineer agent and the client agent can be understood as subordinate to superior. However, since this study does not adopt the ZEUS pre-defined protocol and strategies. This subordinate to superior relationship is represented in the MASCOT protocol and strategies. Peer to peer relationship is defined to all agents.

### 6.4.3.4 Agent Co-ordination

The key aspects of any agent interaction are the co-ordination protocol and the negotiation strategies. The ZEUS toolkit provides a few pre-built co-ordination protocols that implement various aspects of contract-net type conversations, and related strategies.
These protocols and strategies are not suitable for the MASCOT model. New negotiation protocol and strategies were developed at the application design stage.

Although the ZEUS toolkit also provides an approach for system developers to build their own interaction protocols and strategies in a case where the predefined protocol and strategies are not appropriate for some application scenarios, this approach is more suitable for simple cases. In fact, it would be very difficult, if not impossible, for a developer to modify ZEUS' internal key documents to build a complex application protocol and strategies. As a result, this study adopts a more practical approach to developing the negotiation protocol and strategies (i.e. to building the protocol and strategies through the integration of the ZEUS rulebases, primitive tasks, and external Java programmes). At the current stage, this approach is more suitable for building the various negotiation protocols and strategies if the ZEUS pre-defined protocol and strategies cannot be adopted for any specific application scenario.

6.4.3.5 Rulebase Definition

In the ZEUS toolkit, a rulebase is used to refer to a collection of precondition-action rules. Rules provide a means of adding reactive behaviour to agents. Unlike primitive tasks, which are invoked in order to acquire a particular fact, rules are triggered in response to the detection of particular facts or variables. A rule can have one or more conditions. Once all these conditions are satisfied, a rule's actions will occur. The ZEUS toolkit provides several rule actions, which include fact manipulation actions (assert, retract, modify, bind), activity actions (send message, execute), control actions (if, while), goal related actions (achieve, buy, sell), input and output actions, and Java runtime actions. These actions allow agents to conduct various activities required by the MASCOT model as shown in the following example.

Example:

In the rulebase: StartTransferToConditionalProbability, there are two conditions and three actions.
Conditions:

?cons <- (constraints (transferConditionalProbabilityDone no))
?sttcp <- (startTransferToConditionalProbability (start true))

Actions:

(modify ?cons (transferConditionalProbabilityDone pending))
(achieve (fact (startTransferToConditionalProbability (start false))) (end_time 4)
(confirm_time 2))
(modify ?cons (transferConditionalProbabilityDone yes))

According to this rulebase, if the String “transferConditionalProbabilityDone” in the fact constraints is “no”, and the flag “startTransferToConditionalProbability” is “true”, the agent, who owns this rulebase, will conduct three activities:

- first, modify the String “transferConditionalProbabilityDone” to “pending”;
- second, create a new goal for the agent using the supplied parameter (i.e. to achieve the Boolean value of startTransferToConditionalProbability to be “false”;
- third, modify the String “transferConditionalProbabilityDone” to “yes”.

Figure 6.17 illustrates a Rulebase editor window. Table 6.17 describes all the Rulebases in the MASCOT model.
Table 6.17 Rulebases of MASCOT model

<table>
<thead>
<tr>
<th>Rulebases</th>
<th>Function Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register</td>
<td><strong>Rule 1</strong> Register</td>
</tr>
<tr>
<td><strong>Key Trigger</strong></td>
<td>registerDone (false).</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>To register with the agent Mfacilitator once the negotiation starts.</td>
</tr>
<tr>
<td>StartTransferToConditionalProbability</td>
<td><strong>Rule 1</strong> startTransferToConditionalProbability</td>
</tr>
<tr>
<td><strong>Key Trigger</strong></td>
<td>startTransferToConditionalProbability (start true)</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>To achieve the goal startTransferToConditionalProbability (start false). To do this it calls the primitive task TransferToConditionalProbability.</td>
</tr>
<tr>
<td>StartMakeInitialOffer</td>
<td><strong>Rule 1</strong> StartMakeInitialOffer</td>
</tr>
<tr>
<td><strong>Key Trigger</strong></td>
<td>(startMakeInitialOffer (start true))</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>To achieve the goal startMakeInitialOffer (start false). To do this it calls the primitive task MakeInitialOffer.</td>
</tr>
<tr>
<td>Nprotocol</td>
<td><strong>Rule 1</strong> StartAssignValue</td>
</tr>
<tr>
<td><strong>Key Trigger</strong></td>
<td>startAssignValue (start true) which was set in the primitive task MakeNewOffer.</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>To inform the client agent the negotiation history between the engineer agent and the contractor agent.</td>
</tr>
<tr>
<td></td>
<td><strong>Rule 2</strong> InvolvementOfClient</td>
</tr>
<tr>
<td><strong>Key Trigger</strong></td>
<td>startInvolvementOfClient (start true), which was set in the primitive task MakeNewOffer.</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>To replace the engineer agent’s reservation value with the client agent’s reservation value if the latter are higher than the former.</td>
</tr>
</tbody>
</table>
Table 6.17 Rulebases of MASCOT model (cont.)

<table>
<thead>
<tr>
<th>Rulebases</th>
<th>Function Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>StartNegotiation</strong></td>
<td></td>
</tr>
<tr>
<td>Rule 1</td>
<td><strong>StartNegotiation</strong></td>
</tr>
<tr>
<td>Key Trigger</td>
<td>startStartNegotiation (start true), which was set from the primitive task MakelnitialOffer.</td>
</tr>
<tr>
<td>Goal</td>
<td>To send several negotiation information from the contractor agent to the engineer agent. In common cases, they can be set as initial agent and respondent agent.</td>
</tr>
<tr>
<td>Rule 2</td>
<td><strong>startAssignOptimumValue</strong></td>
</tr>
<tr>
<td>Key Trigger</td>
<td>startAssignOptimumValue (start true), which was sent from the initial agent (contractor agent) to the respondent agent (engineer agent) in the Rule 1.</td>
</tr>
<tr>
<td>Goal</td>
<td>To achieve the goal startAssignOptimumValue (start false). To do this it calls the primitive task AssignOptimumValue, i.e. the respondent agent (engineer) assign the initial agent (contractor) initial value as its optimum value.</td>
</tr>
<tr>
<td>Rule 3</td>
<td><strong>SendlnitialOffer</strong></td>
</tr>
<tr>
<td>Key Trigger</td>
<td>StartSendProposalBack(start true), which was set in the primitive task AssignOptimumValue.</td>
</tr>
<tr>
<td>Goal</td>
<td>To send the respondent agent (engineer)'s initial value to the initial agent (contractor).</td>
</tr>
<tr>
<td>Rule 4</td>
<td><strong>AssignOptimumValue1</strong></td>
</tr>
<tr>
<td>Key Trigger</td>
<td>startAssignOptimumValue1(start true), which was sent from the respondent agent (engineer agent) to the initial agent (contractor agent) in the Rule 3.</td>
</tr>
<tr>
<td>Goal</td>
<td>To achieve the goal startAssignOptimumValue1 (start false). To do this it calls the primitive task AssignOptimumValue1, i.e. the initial agent (contractor agent) assign the respondent agent (engineer) initial value as its optimum value.</td>
</tr>
<tr>
<td>Rule 5</td>
<td><strong>RespondTOProposal</strong></td>
</tr>
<tr>
<td>Key Trigger</td>
<td>startEvaluateProposal(start true), which was set in the primitive task AssignOptimumValue1.</td>
</tr>
<tr>
<td>Goal</td>
<td>To compare the opponent’s offer with the agent’s offer, if it satisfies the agent, then accept this offer, otherwise, start to make a new offer.</td>
</tr>
</tbody>
</table>
6.3.6 Primitive Task Definition

A primitive task is a presentation of the lowest level of agent activities. Each task can be depicted as a resource flow, where facts flow into a task, whereupon they are transformed into new facts. To define a primitive task, the following elements need to be configured:

- **Preconditions** - the resources needed for the execution of the task;
- **Effects** - the resources that will be produced upon execution of the task;
- **Cost** - an expression giving the cost of executing the task;
- **Duration** - an expression giving the time taken to execute the task;
- **Precondition Ordering** - the sequence in which preconditions should be achieved;
Within these factors, the preconditions and effects are the most essential elements, especially the setting of modifiers. The modifier in preconditions determines how the precondition resources will perform in the task. The modifiers applicable to preconditions are described in Table 6.18.

### Table 6.18 Modifiers in primitive task preconditions

<table>
<thead>
<tr>
<th>Modifier</th>
<th>Applying Situation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not</td>
<td>Task only performed if fact is not present in locally</td>
</tr>
<tr>
<td>Is Read Only</td>
<td>Fact is not consumed (i.e. it survives task execution, but during that time it is not allocated exclusively to the task)</td>
</tr>
<tr>
<td>Must be in Local Database</td>
<td>Fact must be in the agent’s own local possession, prevents agent obtaining it from another agent</td>
</tr>
<tr>
<td>Is Replaced after Use</td>
<td>Fact will be allocated to the task for the duration of its execution, but will not be consumed</td>
</tr>
</tbody>
</table>

The Modifiers field for effects has a single option: 'Is a Side-Effect Only'. If this is selected, this task will not be selected by the Planner/Scheduler seeking to achieve this effect. The difference between an effect and a side-effect is subtle. The same fact can be either defined as a side-effect or not as a side-effect depending the objective of the task itself. In this study, it is relatively easy to determine which item should be defined as a side-effect because every primitive task is initiated by a rulebase in which a trigger fact is set as a objective of the primitive task. All the effects except the trigger effect should not be set as side-effects. The following example describes this method.

**Example:**

To achieve the new goal `startTransferToConditionalProbability` (start false) defined in the rulebase, a primitive task `TransferToConditionalProbability` is created. The task preconditions include four facts:

- `StartTransferToConditionalProbability`, which is consumed in the task, thus, must be in the local database;
- **Constraints**, which is not consumed in the task, thus, is read only;
- **AgentsName**, which is not consumed in the task, thus, is read only;
- **StartMakeInitialOffer**, which is the trigger for the next rulebase, will be consumed in the task, thus, must be in the local database.

The task effects include eight facts:

- **conditionalProbability, hypothesis, probability, pevent, cprosl, and mem** are a series of arrays which are transferred from the owner’s initial data through the primitive task and the external Java program;
- **StartMakeInitialOffer** is the trigger for the next action. The attribute of this trigger is set as true in the task effects;
- **StartTransferToConditionalProbability** is the objective of this primitive task, thus, this fact is not set as side-effect. All the other facts are set as side-effects.

Figure 6.18 illustrates a task editor window. Table 6.19 describes all the primitive tasks in the MASCOT model.
### Table 6.19 Primitive tasks in MASCOT model

<table>
<thead>
<tr>
<th>Primitive Tasks</th>
<th>Task Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TransferToConditionalProbability</strong></td>
<td>To convert the interface input qualitative information into quantitative information for further calculation (see the above analysis).</td>
</tr>
</tbody>
</table>
| **MakeInitialOffer** | To make an initial offer according to the constraints.  
  - The input information includes: constraints, agentsName, and two startflags: startTransferToConditionalProbability and startMakeInitialOffer. The first two are set as READ_ONLY, while the latter two are set as LOCAL.  
  - The output information includes: offer and two startflags: startMakeInitialOffer and startTransferToConditionalProbability. The first two outputs are set as SIDE_EFFECT. |
| **MakeNewOffer** | To make a new offer according to the constraints.  
  - The input information includes: constraints, agentsName, conditionalProbability, hypothesis, optimumValue, pevet, probability, offer, reservationValue, cprosl, mem and two startflags: startSendProposalBack, startInvolvementOfClient and startMakeNewOffer. The first six ontology are set as READ_ONLY, while the others are set as LOCAL (because these elements will be modified in this task).  
  - The output information includes: offer, startMakeNewOffer, reservationValue, probability, cprosl, proposal, startInvolvementOfClient, startSendProposalBack and mem. All the elements are set as SIDE_EFFECT except startMakeNewOffer. |
| **AssignOptimumValue** | To assign the contractor agent’s initial offer as its optimum value.  
  - The input information includes: constraints, agentsName, startAssignOptimumValue, startRespondToProposal, and startAssignOptimumValue1. All the elements except agentsName are set as LOCAL.  
  - The output information includes: startAssignOptimumValue, startRespondToProposal, optimumValue, startAssignOptimumValue1, and offer. All the elements are set as SIDE_EFFECT except startAssignOptimumValue. |
Table 6.19 Primitive tasks in MASCOT model (cont.)

<table>
<thead>
<tr>
<th>Primitive Tasks</th>
<th>Task Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssignOptimumValue</td>
<td>To assign the engineer agent’s initial offer as its optimum value.</td>
</tr>
<tr>
<td></td>
<td>• The input information includes: agentsName, startRespondToProposal, startAssignOptimumValue1 and initialOffer. The two triggers are as LOCAL, while the others are set as READ_ONLY.</td>
</tr>
<tr>
<td></td>
<td>• The output information includes: startEvaluateProposal, optimumValue, and startAssignOptimumValue1. All the elements are set as SIDE_EFFECT except startAssignOptimumValue1.</td>
</tr>
<tr>
<td>AssignValue</td>
<td>To assign several essential negotiation features into an ontology proposal.</td>
</tr>
<tr>
<td></td>
<td>• The input information includes: constraints, startAssignValue, optimumValue and mem. All the elements except constraints are set as LOCAL.</td>
</tr>
<tr>
<td></td>
<td>• The output information includes: startAssignValue and proposal. The latter is set as SIDE_EFFECT.</td>
</tr>
</tbody>
</table>

Figure 6.19 illustrates how the MASCOT negotiation protocol and strategies are achieved through the integration of rulebases and primitive tasks, where one or more triggers are generated after a Rulebase is executed. For example, a trigger fact needs to be set as ‘false’. This trigger, in turn, will ignite a task. The execution of the task will turn the trigger fact into ‘true’, whilst the objective tasks are performed. A Rulebase may call several tasks or other Rulebases, and vice versa.

Figure 6.19 The co-operation between rulebases and primitive tasks
6.4.3.7 The Utility Agent Configuration

As discussed above, the ZEUS toolkit provides several utility agents to support the infrastructure, which are: name servers, facilitators, visualisers, and database proxies. The implementation needs to address the following four aspects: nameservers, facilitators, visualisers and database proxies, shown in Figure 6.20.

![Figure 6.20 The utility agents configuration panel](image)

- Configuring the Name Servers

An agent society must possess at least one Agent Name Server (ANS). The ANSs maintain a registry of known agents, enabling them to map agent identities to a logical network location. This is necessary because agents only know the names of their
acquaintances and not their locations. Name Server agents are created and configured using the Name Servers' panel. By default the Name Servers table has one entry, referring to a single ANS. However as the ANS is vital to all agent communication it is a potential bottleneck, and so it may be necessary to have multiple ANSs to support larger societies, or to provide a degree of redundancy in case one fails.

The Host field shows the I.P. address (i.e. network location) of the machine that the ANS will run on. More than one ANS can reside on a single host. If there is more than one ANS the developer needs to choose which is the root server, the only operational difference in changing this is that the root server will provide the time-grain value and be responsible for maintaining the society-wide clock. As the root ANS serves as a reference point for all other agents in the society, there must be a means of informing other agents where the root ANS is located. Hence when the root ANS starts, it will write its network location on a file called the Default Name Server (DNS) file. If agents share a network file system it is recommended that the DNS file be expressed in terms of a network pathname.

To make non-root name servers aware of the root ANS, they must be told where to find the root's DNS file. This pathname is entered into the DNS File field. If this is not a network accessible file, it will need to be copied to the local file system of the agent concerned, and hence this field will contain its local pathname and filename. In the MASCOT model, a single ANS is set up by default. It is also selected as the root ANS. The other items in the name server panel are all set as default.

- Configuring the Facilitators

Whereas every agent society must have an ANS, there is no such obligation for Facilitators. Whether or not Facilitators are included depends on the nature of the application. Most of the items in the Facilitator panel are the same as the agent name server panel. By default the Facilitators table has one entry. That suffices for small-scale applications, but as it could be a potential bottleneck it may be desirable to have multiple
Facilitators for larger applications, or where some redundancy is desired in case of failure. Here, all the items in the Facilitator panel are set as default because an Mfacilitator is created.

- Configuring the Visualisers

Like Facilitator, there is no requirement for an application to contain a Visualiser. Whether one is included or not depends on whether the application is to be debugged, monitored or analysed. Given that the Visualiser offers some very useful functionality for free, it will usually be included in the list of agents to be created. By default the Visualisers table has one entry, whether or not more than one is necessary will probably depend on the number of locations where developers will want to visualise some aspect of the society. Another influencing factor is that Visualisers are not essential to the operation of an agent society, and so that the implications of failure are less serious there is less need for redundancy. In the MASCOT model, all the items in the Visualiser panel are left as default.

6.4.3.8 The Task Agent Configuration Stage

This stage takes the same process as the previous stage for the application’s utility agents. All task agents are configured through the 'Task Agents' pane of the Code Generator window, which includes nine factors, such as: Status (modified or saved), Host and DNS file, and Icon (Figure 6.21). Three factors need to be emphasised:
• **External Database**: A task agent can obtain the resources it needs to perform its tasks from one of three sources: its resource database, other agents, or an external data source. The latter can be connected to the task agent through a Java class that implements the `zeus.actors.ExternalDb` interface. This is achieved through the Database External in the Task Agent panel.

• **External Program**: Different from the utility agents, a task agent can be linked to external programs that enable it to send or receive information from the outside world. In the MASCOT model, the external programs are agents’ interfaces (`ConstructorUI`, `EngineerUI` and `ClientUI`) through which instructions are received and to which agents send results. Figure 6.22 illustrates the external programs and their related external classes.

• **Agent Viewer GUI**: The Agent Viewer GUI is a specialised tool that displays detailed information on the internal components of the agent. This is independent of any other application front-end and provides the best means for understanding how
the agent works, as well as being an excellent debugging aid. In the MASCOT model, most of the key information of the Contractor agent, the Engineer agent and the Client agent are sent the agent viewers to observe their activities during negotiations.

<table>
<thead>
<tr>
<th>External Program</th>
<th>External Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>• ContractorUI</td>
<td>OwnerFrontEnd: This class defines the input window.</td>
</tr>
<tr>
<td>• EngineerUI</td>
<td>call NegotiationPanel: Once the interface window has had constraints entered and 'start' pressed then this rule fires to send the constraints to the agent. It also sends the trigger startTransferToConditionalProbability (start true) to the contractor agent and the engineer agent so that their status can become active.</td>
</tr>
<tr>
<td>• ClientUI</td>
<td>call</td>
</tr>
</tbody>
</table>

Figure 6.22 External programs and their related external classes

6.4.3.9 Primitive Task Implementation and External Classes

There are two files for each primitive task. The first is a machine-generated 'stub' file to deal with the input and output information with a single call to another external Java class containing the actual task. The second file is the external Java class. These external classes, in the MASCOT model, are differentiated by the prefix 'My' in front of the name. This two-file structure enables new domain-specific functionality to be integrated with the automatically created agent-specific code without needing to modify the latter.

Figure 6.23 illustrates the relationship between agents, tasks, and external classes. In the course of interacting with other agents, an agent may need to perform a task, whereupon information is passed from agent to task. Once in the task it can be used for whatever purpose it is required, before the information is returned to the agent (modified or not). Meanwhile, external Java classes, written by the developer, may conduct various complex calculations and numerical value assignments by using the data from the tasks.
Each task consists of three sections: an initial section where the information is read in from the agent, a mid-section where it is used and a final section when it is returned to the agent. As agents are unaware of what happens inside tasks, when one is performed the agent will pass information into it regardless of whether it will be used or not. Furthermore the agent will expect to receive information in return. The interface between agent and task, i.e. the facts sent and expected back is what is specified when the developer describes the preconditions and effects of a task.

Consequently, of the three sections in a task the mid-section is optional and the first and last sections are generated by default in the task stub file. This means that if the task does not need to use the information passed into it, the developer need not alter the task stub file. Thus, this activity is only relevant if the developer wishes to alter how the information received is used, and its scope is limited to the shaded area depicted in Figure 6.23.

Typical actions that are encoded into task bodies display information for the developer’s benefit, and passing information to an external class for processing. The external classes
are different from the external programs. The external classes are linked to the tasks, which are invoked when a particular activity is performed and terminated when the activity is completed, and launched when tasks are executed. An external class may also call some other external classes. By contrast, the external programs are linked to agents through an interface. They are launched when the agent starts, and may persist for as long as the agent does. Table 6.19 shows the external programs, primitive task codes generated by the ZEUS toolkit, and the external classes.

Table 6.20 Primitive tasks and external classes

<table>
<thead>
<tr>
<th>Primitive Task</th>
<th>External Class</th>
<th>Function Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TransferToConditionalProbability</td>
<td>MyTransferToConditionalProbability</td>
<td>To transfer the input information into four calculation attribute data, calls another class, PriorDistribution, with these data, then visualises the output of PriorDistribution, i.e. hypothesised probability and the conditional probability.</td>
</tr>
<tr>
<td>AssignOptimumValue</td>
<td>MyAssignOptimumValue</td>
<td>To assign the contractor agent's initial offer as its optimum value.</td>
</tr>
<tr>
<td>AssignOptimumValue1</td>
<td>MyAssignOptimumValue1</td>
<td>To assign the engineer agent's initial offer as its optimum value.</td>
</tr>
<tr>
<td>MakeInitialOffer</td>
<td>MyMakeInitialOffer</td>
<td>To create initial offer for the contractor agent and the engineer agent.</td>
</tr>
<tr>
<td>MakeNewOffer</td>
<td>MyMakeNewOffer</td>
<td>To make new offers for each agent. This class achieves the Zeuthen's concession strategy in negotiation.</td>
</tr>
<tr>
<td>AssignValue</td>
<td>MyAssignValue</td>
<td>To assign several essential negotiation features into an ontology proposal. This is mainly used for drawing the negotiation process.</td>
</tr>
<tr>
<td></td>
<td>PriorDistribution</td>
<td>This is an external class called by an external class MyTransferToConditionalProbability. It accepts attribute data from this class and calculates the probability of hypothesised reservation values and the probabilities.</td>
</tr>
</tbody>
</table>

The link between primitive tasks and external classes is achieved by adding codes in the stub file that call an external class. The stub files are machine generated. They are over-
written whenever the primitive task editor is modified, and the code is regenerated. Thus, developers have to modify the stub file every time after it is regenerated. The following examples (Listing 6.1) show how a stub file calls an external Java file. The first section of Java code is the stub file generated by the ZEUS Agent Generator, the bold lines are developed by developers. The second section is the external Java code developed by developers, which fulfils the function of real task.

**Primitive Task Code: MakeInitialOffer**

```java
import java.util.*;
import zeus.util.*;
import zeus.concepts.*;
import zeus.actors.TaskContext;
import zeus.actors.ZEUSTask;

public class MakeInitialOffer extends ZEUSTask {
    protected void exec() {

        // The Input Facts:
        Fact[] _smioin = inputArgs[0]; // startMakeInitialOffer
        Fact[] _cons = inputArgs[1]; // constraints
        Fact[] _ssnin = inputArgs[2]; // startStartNegotiation
        Fact[] _an = inputArgs[3]; // agentsName

        // The Output Facts:
        Fact[] _inofout; // offer
        Fact[] _smioout; // startMakeInitialOffer
        Fact[] _ssnout; // startStartNegotiation

        /* DEVELOPER CODE STARTS */

        System.out.println("-Expected Input-");
        for(int i = 0; i < expInputArgs.length; i++)
            System.out.println(expInputArgs[i].pprint());
    }
}
```

243
System.out.println("-Input-");
for(int j = 0; j < inputArgs.length; j++) {
    System.out.println("Input Fact\["+j+\"]\n");
    for(int i = 0; i < inputArgs[j].length; i++)
        System.out.println(inputArgs[j][i].pprint());
}
System.out.println("-Expected Output-");
for(int i = 0; i < expOutputArgs.length; i++)
    System.out.println(expOutputArgs[i].pprint());

System.out.println("-Output-");
_inofout = new Fact[1];
_inofout[0] = new Fact(Fact.FACT, expOutputArgs[0]);

MyMakeInitialOffer id = new MyMakeInitialOffer();
id.exec(_cons, _an, _inofout);

System.out.println(_inofout[0].pprint());
_smioout = new Fact[1];
_smioout[0] = new Fact(Fact.FACT, expOutputArgs[1]);
System.out.println(_smioout[0].pprint());
_ssnout = new Fact[1];
_ssnout[0] = new Fact(Fact.FACT, expOutputArgs[2]);
System.out.println(_ssnout[0].pprint());

/* DEVELOPER CODE ENDS */
outputArgs = new Fact[3][];
outputArgs[0] = _inofout;
outputArgs[1] = _smioout;
outputArgs[2] = _ssnout;
}

External Class: MyMakeInitialOffer

import java.util.*;
import zeus.util.*;
import zeus.concepts.*;
import zeus.concepts.fn.*;
import zeus.actors.ZEUSTask;

public class MyMakeInitialOffer extends MakeInitialOffer
{
    private final String pn="MyMakeInitialOffer";
    protected void exec(Fact[] _cons, Fact[] _an, Fact[] _inofout)
    {
        System.out.println(pn+" agent "+_an[0].getValue("name");

        //set up proposal facts
        Fact of= new Fact(Fact.FACT,_inofout[0]);

        //declare variables and get values for constraints
        double myOptv=Double.valueOf(_cons[0].getValue("myOptv")).doubleValue();

        //the initial offer is the optimum value
        of.setValue("offer", myOptv);

        // put the figure into offer
        _inofout[0].setValue("offer", new FactFn(of));
        _inofout[0].setValue("offer", myOptv);
        _inofout[0].setValue("toBeEvaluatedBy", _an[0].getValue("name");
        _inofout[0].setValue("status", "mine");
        _inofout[0].setValue("owner", _an[0].getValue("name");
    }
}

Listing 6.1 The integration of primitive tasks and external Java codes

6.4.3.10 Display Windows

Display windows help users to understand more clearly what is going on within the system through the provided visual information. The function of the display windows is similar to the Visualisar agent; however, in this case, it is more specific to the negotiation
features. The MASCOT implementation provides for three display windows programmed using Java language. The data is transferred from ZEUS external programmes, primitive tasks and classes.

- The first display window is generated after a negotiator inputs all the necessary information in the input window (Figure 6.24). It provides the negotiator with two kinds of information: the probability distribution curve of the opponent’s possible reservation value; and the conditional probability distribution curve of the opponent’s real offer given a hypothesised reservation value. Such information can help the users to check whether the input information is within a reasonable range.

![Figure 6.24. Display window of a party’s estimate of the opponent’s information](image)

- The second display window is generated during the negotiation process. This illustrates all the key information in which a negotiating agent (the contractor or engineer agent) and its owner are interested (Figure 6.25). This information includes the agent’s reservation value, optimum value, current offer, its opponent’s current offer, optimum value and the agent’s estimation of opponent’s reservation value at the current stage. This display window helps the agent’s owner to monitor the negotiation process, and understand how the negotiation is conducted between agents.
The third display window is also generated during the negotiation process. It shows the client agent and its owner the negotiation history between the contractor agent and the engineer agent; and in which situations, the client agent may join the negotiation between the negotiating agents (Figure 6.26). This display window also reminds the client of his reservation value and optimum value.
6.4.4 Running the Program

To run the program, there are three batch files to run in this order:

- Run1: sets up the name server;
- Run2: starts the application agents; interface agent, contractor agent, engineer agent and the client agent; and
- Run3: starts the visualiser.

6.5 PROBLEMS AND SOLUTIONS

The developed prototype was tested in various situations, and one problem was found. That is, a negotiation party may make an unreasonably large concession in the first iteration, or does not make any concession in the first few iterations. This problem could happen in two extreme cases:

- Case one: a party does not have adequate information about the negotiation item, or lacks enough domain knowledge about his opponent, but he still believes that his estimate of the opponent's key negotiation features is reliable. In this case, this party's input of the opponent's reservation value or negotiation habit could be very far from the opponent's real value, whilst he still thinks that his confidence about the estimate is high (or at least, medium). Consequently, his updated reservation value in the first iteration will be quite different from the opponent's real value. Thus, he may concede too much (e.g. he gives up all his bargaining amount) in the first iteration or may not concede at all. Consequently, the negotiation will converge with an unreasonably high rate or never converge.

- Case two: the opponent may adopt a misleading strategy (i.e. he purposely makes an extremely high or low initial offer). In this case, if the negotiator holds his current
position (or makes very little concession as a gesture) for several iterations, it is expected that the opponent will come back to the normal negotiation offer.

The first case is harmful for the negotiator because he may give too much to the opponent than he would otherwise get, or a party may lose the opportunity to reach a possible agreement. The second case is the negotiator's right response. A distinction between these two cases is that in the first case, the opponent will not make a big concession even if the negotiator does not concede for several iterations; whilst in the second case, the opponent may concede very quickly after the first few iterations.

To overcome this problem, two extra rules were applied in the external class 'myMakeNewOffer':

- first, each party cannot make a concession that is more than a threshold (e.g. one quarter of his bargaining amount) in the first (or first three) iteration(s); and
- second, each party should make a minimum concession, (e.g. one tenth of his bargaining amount) in the first (three) iteration(s).

By adopting these two simple rules, the problems caused by the wrong input information can be avoided.

Besides the problem mentioned above, there is a trend where both parties prefer to make high concessions in the first few iterations, whilst making low concessions at the late stages of the negotiation. This is caused by the MASCOT negotiation mechanism in which both parties try to reduce the risk of conflict.

6.6 SUMMARY

This chapter has described the implementation of a MASCOT model prototype. It first explored the major characteristics of the ZEUS toolkit, where the issues for building the DAI application tool, design philosophy, functions, and architecture of the toolkit were
addressed. It particularly explored the system component library, building software, structure of a generic agent, utility agents, task agents, and external programs. This chapter then presented the three major steps of the system implementation process; which included: role modelling, application design and model realisation:

- Firstly, the concept, component, process, and applications of role modelling were introduced. The development of the MASCOT role model was further discussed;

- Secondly, the application design was performed, where the MASCOT negotiation protocol and strategies were further developed; the input information for the prototype was determined; and the probability distributions for agents’ input information were selected;

- Thirdly, the realisation process was discussed in detail, where the ontology, agent creation process, agent organisation, agent co-ordination were first addressed. Then, the creation and functions of Rulebases and Primitive Tasks were discussed. Particular attention was paid to how the Rulebases and Primitive Tasks were integrated to achieve the system negotiation protocol and strategies. Furthermore, the configuration of the utility agents and task agents was discussed. Then, it emphasised the difference between the external programme and external class, and how they worked with the ZEUS primitive task. In the last stage, the display windows of the prototype were introduced.

- Finally, the major problem of the initial prototype (i.e. the negotiation agent could make unreasonable concession in the initial iteration(s)) was addressed. The solutions adopted were also discussed.

The next chapter describes the operation of the MASCOT conceptual model using a practical problem. All the major aspects of the model are presented.
CHAPTER 7

SYSTEM OPERATION

7.1 INTRODUCTION

This chapter presents the operation of the MASCOT model in a real application scenario involving a claim for loss of productivity. A number of the key points of system operation are described including: the utility functions, the negotiation preparation, the negotiation process, and the comparisons between the MASCOT negotiation mechanism with other negotiation mechanisms.

The aim of system operation is to demonstrate the working of the MASCOT conceptual model. The key negotiation strategies of the system are illustrated using a practical problem. Although the system, as currently implemented, is far from being able to handle very complex practical problems, it is important to ensure that the system works properly in a variety of situations, and that the outcomes generated are reasonable. The advantages of the negotiation mechanism are shown by comparing the solution provided with a simple gradient descent approach for the same problem.

The focus of this operation process is on the contractor agent's offering mechanism which includes how the contractor agent updates its beliefs and uses its beliefs to concede. Since the simple concession approach is adopted (see Section 5.4.4.2), negotiation strategies such as the involvement of the client agent and expanded solution searching are not demonstrated. The first example is presented in full, while only the assumptions and outcomes of the other examples (which are variants of the first example) are presented. This problem has also been run using the developed MASCOT prototype.
7.2 BACKGROUND TO THE EXAMPLE

The project presented below is based on a real water supply project. The author, working as the project manager, was involved in the whole claims negotiation process.

The work of the project mainly comprised the construction of intakes, pipelines, and treatment plants in seven towns. The contract amount was about £10 million. The estimated construction duration was 26 months. The contract followed the FIDIC 4th edition. The project was designed by the engineer three years before the tender without a detailed geological investigation. According to the design, the intakes of A, G and P towns adopted the same structural style (i.e. the walls of intake chamber were based on the driven concrete piles). The length of the piles was 8m with 7.5 m driven into the riverbed. The remaining 0.5m would be used to connect with the wall of intake. The number of piles was 225. The contractor's quotation for piles was £50/m. According to the contractor's schedule, G town's intake would be started first. Once the piling work in G town was completed, the piling team would move to P and then A town.

After two days' piling work, the contractor found that piles could only be driven 3.5m into riverbed. The contractor informed the engineer immediately. The engineer replied that it was the contractor's responsibility to drive the piles to the designed level. The failure of piling work was caused by the contractor's old piling machine. Any delay would be the responsibility of the contractor. As a result, the contractor continued the work with all the possible methods. However, only three piles were driven 4.5m into the riverbed by the time all the piling work was completed. The work took 80 days while it was planned to be completed within 20 days.

Following a request by the contractor, a borehole was drilled by the National Geology Laboratory (NGL). The test showed that a 3m dense gravel-sand layer lay underneath the riverbed at about 3-4m while the original drawing showed this as soft clay. By showing the evidence and informing the client, the engineer finally agreed that the problem was primarily created by the faulty design. Consequently, a claim was submitted by the
contractor which included time extension, new pile rate, cost of pile cutting and removing, overheads and loss of productivity. All these claims were supported with a detailed cost breakdown. Since town A and P’s intake might encounter the same problem, both the contractor and the engineer were very cautious about the claim. The real case took more than four months to be settled. The most difficult negotiation item was loss of productivity, and more than 10 negotiation meetings were held at different levels. In this study, the MASCOT model will be adopted to resolve the claim for loss of productivity.

7.3 UTILITY FUNCTION

In claims negotiation, neither the contractor agent nor the engineer agent can have complete information about the opponent. Uncertainty about the opponent’s utility function is a critical feature of construction claims negotiation. In the MASCOT prototype, it is assumed that both the contractor agent and the engineer agent’s utility functions are linear, which can be determined by two points: the optimum (amount) point and the reservation (amount) point. Here, the utility is assumed as 1 at the optimum point, and 0.6 at the reservation point (Figure 7.1a, b). Each agent can estimate the opponent’s utility function based on these two critical points. Meanwhile, it is reasonable to assume that each agent’s initial offer is its optimum amount (i.e. each agent starts the negotiation by offering the opponent the deal that is best for itself). Thus, an agent can know the opponent’s utility function if it can estimate the opponent’s reservation amount through the learning mechanism. The relationship between the utility functions of the contractor agent and the engineer agent are shown in Figure 7.1c.

7.4 NEGOTIATION PREPARATION

The reasons for the contractor’s claim for the loss of productivity were:

- many different methods such as increased piling times and increased weight of hammer, have been tried to drive the piles to the designed level, therefore, the efficiency of piling work was seriously affected;
Figure 7.1a The Contractor agent’s utility function

Figure 7.1b The Engineer agent’s utility function

Figure 7.1c The correlation between the two agents’ utility functions
the efficiency of piling was adversely affected as the piling team was mobilised and
demobilised several times;
the extra length of piles above riverbed influenced site transportation;
the delay of piling work influenced the construction of intake well which was just
beside the chamber;
the precast piles occupied the site store during the delay, therefore, all the other
construction materials have to be stored a mile away from the site; and
the schedules of P and A town were influenced.
Based on these factors, the contractor estimated his real loss of productivity as £9,000.
Considering his current situation and the importance of the claim, he prepared his critical
negotiation figures as follows:

Table 7.1 The contractor's major negotiation figures

<table>
<thead>
<tr>
<th>Reservation value</th>
<th>Optimum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>£9,000</td>
<td>£11,000</td>
</tr>
</tbody>
</table>

Meanwhile, the contractor also tried to estimate the engineer's critical negotiation figures
based on his domain knowledge. In this case, the contractor was interested in the
engineer's reservation value. Table 7.2 shows the contractor's estimate about the possible
distribution of the engineer's reservation value. Table 7.3 shows the contractor's estimate
about the conditional probabilities of the engineer's offer given his hypotheses of the
engineer's reservation value. The data in Table 7.3 is encoded from the contractor's
perception about the engineer negotiation strategy; for example, 'the engineer will
normally make his offer 10% lower than what he really wants'.

Table 7.2 The contractor's prior knowledge about the probability distribution of the
engineer's reservation value

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>R₁</th>
<th>R₂</th>
<th>R₃</th>
<th>R₄</th>
<th>R₅</th>
<th>R₆</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(Rᵢ)</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>R₁</th>
<th>R₂</th>
<th>R₃</th>
<th>R₄</th>
<th>R₅</th>
<th>R₆</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probability P(Rᵢ)</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
</tr>
</tbody>
</table>

£7,000 £8,000 £8,500 £9,000 £10,000 11,000
Table 7.3 The contractor's prior knowledge about the conditional probabilities of the engineer's offer

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>£6,000</th>
<th>£7,000</th>
<th>£8,000</th>
<th>£8,500</th>
<th>£9,000</th>
<th>£9,500</th>
<th>£10,000</th>
<th>£11,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,000</td>
<td>0.35</td>
<td>0.35</td>
<td>0.25</td>
<td>0.04</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,000</td>
<td>0.1</td>
<td>0.4</td>
<td>0.35</td>
<td>0.12</td>
<td>0.03</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,500</td>
<td>0</td>
<td>0.14</td>
<td>0.5</td>
<td>0.30</td>
<td>0.05</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£9,000</td>
<td>0</td>
<td>0.1</td>
<td>0.15</td>
<td>0.45</td>
<td>0.25</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£10,000</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.20</td>
<td>0.45</td>
<td>0.25</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td>£11,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.1</td>
<td>0.25</td>
<td>0.40</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Similarly, the engineer also calculated and determined his key negotiation figures (Table 7.4), and estimated the contractor's key information: the possible distribution of reservation value (Table 7.5) and the conditional probabilities of the contractor's offers given the hypothesised reservation value (Table 7.6).

Table 7.4 The engineer's major negotiation figures

<table>
<thead>
<tr>
<th>Reservation value</th>
<th>Optimum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>£9,800</td>
<td>£7,000</td>
</tr>
</tbody>
</table>

Table 7.5 The engineer's prior knowledge about the probability distribution of the contractor's reservation value

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,500</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td>0.2</td>
<td>0</td>
</tr>
<tr>
<td>Probability P(R)</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.2</td>
<td>0</td>
</tr>
</tbody>
</table>
Table 7.6 The Engineer’s prior knowledge about the conditional probabilities of the contractor’s offer

<table>
<thead>
<tr>
<th>Possible event</th>
<th>Hypothesis</th>
<th>£7,500</th>
<th>£8,000</th>
<th>£8,500</th>
<th>£9,000</th>
<th>£10,000</th>
<th>£11,000</th>
<th>£12,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,500</td>
<td>0.35</td>
<td>0.5</td>
<td>0.1</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>£8,000</td>
<td>0.15</td>
<td>0.35</td>
<td>0.4</td>
<td>0.1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>£9,000</td>
<td>0</td>
<td>0.01</td>
<td>0.11</td>
<td>0.3</td>
<td>0.4</td>
<td>0.18</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>£10,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.15</td>
<td>0.4</td>
<td>0.4</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>£11,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.25</td>
<td></td>
</tr>
</tbody>
</table>

7.5 NEGOTIATION PROCESS (Example 1)

1) The contractor agent’s initial offer

In this research, the contractor agent makes its initial offer of £11,000 that is assumed as its optimum claim amount.

2) The engineer agent’s initial counter-offer

After receiving the contractor agent’s initial offer, the engineer agent makes a counteroffer of £7,000 to the contractor agent, which is also assumed as the engineer agent’s optimum value.

3) The contractor agent’s offer in the 2\textsuperscript{nd} iteration

- **Updating the probability of the engineer agent’s reservation value**

Based on the engineer agent’s counteroffer and the contractor’s prior knowledge about the engineer shown in Table 7.2 and 7.3, the contractor agent updates its belief about the probability of the engineer agent’s reservation value \( R \) according to the Bayesian rule:
$P(R_i | e) = \frac{P(R_i)P(e | R_i)}{\sum_{k=1}^{n} P(e | R_k)P(R_k)} = \frac{P(R_i)P(e | R_i)}{P(e)}$

Where,

- $P(R_i | e)$ - the probability that the engineer agent's reservation value is $R_i$ given the condition that its offer is $e$;
- $P(R_i)$ - the probability that the engineer agent's reservation value is a certain $R_i$;
- $P(e | R_i)$ - the probability that the engineer agent's offer is a certain $e$ given the certain reservation value $R_i$;
- $P(e)$ - the probability of the engineer agent's counteroffer is $e$.

In this case, the engineer's offer is £7000 ($e_i$=7000), thus,

$P(R_3 | e_1) = \frac{P(R_3)P(e_1 | R_3)}{\sum_{k=1}^{6} P(e_1 | R_k)P(R_k)} = \frac{0.25*0.14}{(0.25*0.14) + (0.25*0.1)} = 0.5833$

$P(R_4 | e_1) = \frac{P(R_4)P(e_1 | R_4)}{\sum_{k=1}^{6} P(e_1 | R_k)P(R_k)} = \frac{0.25*0.1}{(0.25*0.14) + (0.25*0.1)} = 0.4166$

Where,

- $P(R_1 | e_1) = 0$, $P(R_2 | e_1) = 0$ since $P(R_1) = P(R_2) = 0$;

- $P(R_3 | e_1) = 0$, $P(R_6 | e_1) = 0$ since $P(e_1 | R_3) = P(e_1 | R_6) = 0$ (see Table 7.3)

**Estimating the engineer agent's reservation value**

Prior to receiving the engineer agent's offer (£7,000), the contractor agent would think that the engineer agent's reservation value is:

$R = \sum P(R_i) * R_i = 0.25*8500+0.25*9000+0.25*10000+0.25*11000 = 9625.$

After receiving the counteroffer, the contractor agent's current estimation of the engineer agent's reservation value is:
R = ΣP(R_i) R_i = 0.5833*8500+0.4166*9000 = 8707.

- Utility functions

a) **The contractor agent's utility function:** Since the agents' utility functions are linear, they can be expressed as \( u_c = kx + b \). Also, the two key points in the utility functions are known, i.e. optimum point: (11000, 1) and reservation point: (9000, 0.6). Thus, the contractor agent's utility function can be worked out as: \( u_c = 2 \times 10^{-4}x - 1.2 \).

b) **The contractor agent's estimate of the engineer agent's utility function:** The engineer agent's utility function is \( u_e = kx + b \), where the contractor agent knows two points along this line based on its updated beliefs: optimum point: (7000, 1); reservation point: (8707, 0.6); Thus, the contractor agent estimates the engineer agent's utility function as: \( u_e = -2.3 \times 10^{-4}x + 2.64 \).

b) **The combined utility function:** Since the contractor agent's and the engineer agent's utility functions are: \( u_c = 2 \times 10^{-4}x - 1.2 \) and \( u_e = -2.3 \times 10^{-4}x + 2.64 \), the correlation between the contractor agent and the engineer agent's utility functions can be calculated as: \( u_e = -1.17u_c + 1.234 \).

- Risk evaluation

According to Zeuthen's model, the maximum likelihood of risk acceptable to the contractor agent (\( P_{c_{\text{max}}} \)) and the engineer agent (\( P_{e_{\text{max}}} \)) can be calculated as:

\[
P_{c_{\text{max}}} = \frac{U_{c_{\text{cc}}}^t - U_{c_{\text{ee}}}^t}{U_{c_{\text{cc}}}^t - U_{c_{\text{e}}}^t (C)}; \quad P_{e_{\text{max}}} = \frac{U_{e_{\text{cc}}}^t - U_{e_{\text{ee}}}^t}{U_{e_{\text{cc}}}^t - U_{e_{\text{e}}}^t (C)}
\]

Where,

\( U_{cc}^t \) - the contractor agent's utility generated by its offer in \( t \) iteration;
U^{t}_{cc} - the contractor agent's utility which offered by the engineer agent's offer in t iteration;

U_c(c) - the contractor agent's utility of a conflict deal, it is assumed as 0 in this case;

U^{t}_{ee} - the engineer agent's utility bring by its current offer in t iteration;

U^{t}_{cc} - the engineer agent's utility which offered by the contractor agent's offer in t iteration; and

U_e(c) - the engineer agent's utility for a conflict deal, it is assumed as 0 in this case (Figure 7.1c).

In this iteration, the offer of the contractor agent and the engineer agent are (11000, 7000). Thus, the maximum likelihood of risk acceptability to the contractor and the engineer are:

\[ P_{c_{\text{max}}} = \frac{U^{t}_{cc} - U^{t}_{ee}}{U^{t}_{cc} - U_e(c)} = \frac{1 - 0.2}{1} = 0.8 \]

\[ P_{e_{\text{max}}} = \frac{U^{t}_{ee} - U^{t}_{cc}}{U^{t}_{ee} - U_e(c)} = \frac{1 - 0.11}{1} = 0.89 \]

- Concession

Since the \( P_{c_{\text{max}}} < P_{e_{\text{max}}} \) (i.e. the contractor's maximum risk acceptability is less than that of the engineer agent), the contractor agent knows that it should make a concession in the next iteration. In this example, a simple concession approach is adopted to calculate the concession rate (i.e. the contractor agent will make the minimum concession sufficient to make the engineer agent's maximum acceptable risk smaller than its own in the next iteration). The concession step can be calculated as:

\[ P_{e_{\text{max}}} = \frac{U_e(w_e) - U_e(D_e)}{U_e(w_e) - U_e(e)} = \left| \frac{1 - u_{ee}}{1 - 0} \right| = 0.8 \]

\[ u_{ee} = 0.2, \Rightarrow u_e = 0.8837 \Rightarrow x = 10418 \]

Thus, the contractor agent's new offer will be equal to, or lower than £10418.
4) The engineer agent's counter-offer in the 2nd iteration

Based on the contractor agent's new offer of £10418, the engineer agent updates its belief about the probability of the contractor's reservation value.

\[
P(R_1 | e_1) = \frac{P(R_1) P(e_1 | R_1)}{\sum_{k=1}^{6} P(e_1 | R_k) P(R_k)} = \frac{0.3 \times 0.308}{(0.3 \times 0.308) + (0.2 \times 0.4)} = 0.536
\]

\[
P(R_2 | e_1) = \frac{P(R_2) P(e_1 | R_2)}{\sum_{k=1}^{6} P(e_1 | R_k) P(R_k)} = \frac{0.2 \times 0.4}{(0.3 \times 0.308) + (0.2 \times 0.4)} = 0.464
\]

Where, \(P(e_1 | R_1)\) is obtained through the method of linear interpolation from data in Table 7.6. Meanwhile, \(P(R_3 | e_1) = P(R_4 | e_1) = P(R_5 | e_1) = 0\) because \(P(e_1 | R_3) = P(e_1 | R_4) = P(R_5) = 0\).

- Estimating the contractor agent's reservation value

Prior to receiving the contractor agent's offer (£10418), the engineer agent would think that the contractor agent's reservation value is:

\[
R = \sum P(R_i) * R_i = 0.2 \times 7500 + 0.3 \times 8000 + 0.3 \times 9000 + 0.2 \times 10000 = 8600.
\]

After receiving the new offer of 10418, the engineer agent's current estimation of the contractor agent’s reservation value is:

\[
R = \sum P(R_i) * R_i = 0.536 \times 9000 + 0.464 \times 10000 = 9464.
\]

- Utility functions

a) The engineer agent's utility function: In this case, the data of the two points, which determine the engineer agent's utility functions, are: optimum value point: (7000, 1); and reservation value point: (9800, 0.6). Thus, the engineer agent's utility function is:

\[
u_e = -1.43 \times 10^{-4} x + 2.
\]
b) The engineer agent's estimate of the contractor agent utility function: At this stage, the engineer agent's information about the contractor agent's utility function is: optimum point: (11000, 1); reservation point: (9464, 0.6). Thus, it calculates the contractor agent's utility function as $u_c = 2.6 \times 10^{-4} x - 1.87$.

c) The combined utility function: Based on the identified two utility functions:

$$u_e = -1.43 \times 10^{-4} x + 2;$$
$$u_c = 2.6 \times 10^{-4} x - 1.87;$$

The function between the contractor agent's utility and the engineer agent's utility can be worked out as: $u_e = -0.55u_c + 0.972$.

- Risk evaluation

In this iteration, the contractor agent and the engineer agent's offers are (10418, 7000), thus, the maximum likelihood of risk acceptability of the engineer agent and contractor agent are:

$$P_{e_{\text{max}}} = \frac{U_{te}^{t} - U_{ee}^{t}}{U_{ee}^{t} - U_{e}(C)} = \frac{1 - 0.51}{1} = 0.49;$$

$$P_{c_{\text{max}}} = \frac{U_{ce}^{t} - U_{ce}^{t}}{U_{ce}^{t} - U_{e}(C)} = \frac{0.84 - 0}{0.84} = 1;$$

- Concession

Since the $P_{e_{\text{max}}} < P_{c_{\text{max}}}$, the engineer agent knows it should make a concession in the next iteration. The concession step can be calculated as:

$$P_{c_{\text{max}}} = \frac{U_{ce}^{t} - U_{ce}^{t}}{U_{ce}^{t} - U_{e}(C)} = \frac{0.84 - U_{ee}^{t}}{0.84} = 0.49,$$

$u_{ce} = 0.428, \Rightarrow u_{c} = 0.736 \Rightarrow x = 8838$

Thus, the engineer agent's new counteroffer will be equal to, or higher than £8838.
5) The contractor agent’s offer in the 3rd iteration

After receiving the engineer agent’s new counteroffer of £8838, the contractor agent repeats the evaluation process as it did in 3) to decide what it should offer for the next iteration.

- **Updating the probability of the engineer agent’s reservation value**

After the last iteration, the contractor agent has already updated its belief about the probability of the engineer agent’s reservation value as:

Table 7.7 The contractor agent’s estimate about the probability of the engineer agent’s reservation value at the beginning of 3rd iteration (Iversen, 1984)

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,000</td>
<td>0</td>
<td>0</td>
<td>0.5833</td>
<td>0.4166</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Probability $P(R/e_1)$</td>
<td>0</td>
<td>0</td>
<td>0.5833</td>
<td>0.4166</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Based on this prior knowledge (at £7000) and the engineer agent’s new counteroffer £8838, the contractor agent updates its belief as follows:

$$P(R | e_1, e_2) = \frac{P(R | e_1)P(e_1, e_2 | R)}{\sum_{k=1}^{6} P(e_1, e_2 | R_k)P(R_k)}$$

Where,

$$P(e_1, e_2, ..., e_n | R_i) = P(e_1 | R_i) * P(e_2 | R_i) * ....... * P(e_n | R_i)$$

$$P(e_1, e_2 | R_i) = P(e_1 | R_i) * P(e_2 | R_i) = (0.14, 0.1) * (0.12, 0.306) = (0.0168, 0.0306)$$

The above equations are based on the assumption that event e1 and e2 are independent (Pearl, 1988). In this iteration, the new prior conditional probabilities are:
\[
P(R_1 | e_1, e_2) = \frac{P(R_1 | e_1)P(e_1, e_2 | R_1)}{\sum_{i=1}^{k} P(e_1, e_2 | R_i)P(R_i)} = \frac{0.5833 \times 0.0168}{(0.5833 \times 0.0168) + (0.4166 \times 0.0306)} = 0.435
\]
\[
P(R_2 | e_1, e_2) = \frac{P(R_2 | e_1)P(e_1, e_2 | R_2)}{\sum_{i=1}^{k} P(e_1, e_2 | R_i)P(R_i)} = \frac{0.4166 \times 0.035}{(0.5833 \times 0.0245) + (0.4166 \times 0.035)} = 0.565
\]

(i.e. after receiving the engineer agent two offers, the contractor agent believes that there are 43.5 % chance that the engineer agent’s reservation value is £8500 and 56.5% chance that the amount is £9000).

- **Estimating the engineer agent’s reservation value**

After receiving the new counteroffer, the contractor agent’s current estimation of the engineer agent’s reservation value is: \(R = 0.435 \times 8500 + 0.565 \times 9000 = 8782\).

- **Utility functions**

  a) **The contractor agent’s utility function**\(^{22}\): \(u_c = 2 \times 10^{-4} x - 1.2\)

  b) **The contractor agent’s estimate of the engineer agent**: Based on its updated beliefs about the engineer agent’s reservation value, the contractor agent identifies the two points in the engineer agent’s utility function: optimum point: \((7000, 1)\); and reservation point: \((8782, 0.6)\). Thus, the contractor agent estimates the engineer agent’s utility function as: \(u_e = -2.245 \times 10^{-4} x + 2.571\).

c) **The combined utility function**

Since
\[
u_c = 2 \times 10^{-4} x - 1.2; \quad u_e = -2.245 \times 10^{-4} x + 2.571;
\]

\(^{22}\) The contractor agent has worked out its utility function at 2nd iteration, so does the engineer agent.
Thus, the function between the contractor agent's and the engineer agent's utility function is: \( u_e = -1.1225u_c + 1.224 \).

- **Risk evaluation**

In this iteration, the offer of the contractor agent and the engineer agent are (10418, 8838). Thus, the maximum likelihood of risk acceptability to the contractor agent and the engineer agent are:

\[
P_{c_{\text{max}}} = \frac{U'_{ce} - U_{ce}}{U'_{ce} - U_{ce}(C)} = \frac{0.8836 - 0.572}{0.8836} = 0.353
\]

\[
P_{e_{\text{max}}} = \frac{U'_{ee} - U_{ee}}{U'_{ee} - U_{ee}(C)} = \frac{0.582 - 0.1015}{0.582} = 0.826
\]

- **Concession**

Since the \( P_{c_{\text{max}}} < P_{e_{\text{max}}} \), the contractor agent knows it should make a concession in the next iteration. The concession step can be calculated as:

\[
P_{e_{\text{max}}} = \frac{U(w_e) - U(D_e)}{U(w_e) - U(e)} = \frac{0.582 - u_{ce}}{0.582 - 0} = 0.353
\]

\( u_{ce} = 0.377 \), \( u_c = 0.755 \) \( \Rightarrow x = 9775 \)

Thus, the contractor agent's new offer will be equal to, or lower than £9775.

6) **The engineer agent's counter-offer in the 3\textsuperscript{rd} iteration**

Based on the last two encounters, the engineer agent's prior probability about the contractor agent's reservation value is shown in Table 7.8.
Table 7.8 The engineer agent's estimate about the probability of the contractor agent's reservation value at the beginning of 3rd iteration

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>R₁</th>
<th>R₂</th>
<th>R₃</th>
<th>R₄</th>
<th>R₅</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probability P(R₁</td>
<td>e₁)</td>
<td>0</td>
<td>0</td>
<td>0.536</td>
<td>0.464</td>
</tr>
<tr>
<td>Probability P(R₂</td>
<td>e₁)</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probability P(R₃</td>
<td>e₁)</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probability P(R₄</td>
<td>e₁)</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probability P(R₅</td>
<td>e₁)</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

After receiving the new offer of 9775, the engineer updates its condition probability as:

\[
P(e_1, e_2 \mid R_1) = P(e_1 \mid R_1) \cdot P(e_2 \mid R_1) = (0.3775, 0.343) \cdot (0.291, 0.4) = (0.1099, 0.1375)
\]

\[
P(R_2 \mid e_1, e_2) = \frac{P(R_2 \mid e_1)P(e_1, e_2 \mid R_2)}{\sum_{i=1}^5 P(e_1, e_2 \mid R_i)P(R_i)} = \frac{0.464 \cdot 0.1375}{(0.536 \cdot 0.1099) + (0.464 \cdot 0.1375)} = 0.52
\]

\[
P(R_1 \mid e_1, e_2) = \frac{P(R_1 \mid e_1)P(e_1, e_2 \mid R_1)}{\sum_{i=1}^5 P(e_1, e_2 \mid R_i)P(R_i)} = \frac{0.536 \cdot 0.1099}{(0.536 \cdot 0.1099) + (0.464 \cdot 0.1375)} = 0.48
\]

The engineer agent's current estimation of the contractor agent's reservation value is:

\[
R = 0.48 \cdot 9000 + 0.52 \cdot 10000 = 9520.
\]

- **Utility functions**

  a) **The engineer agent's utility function**: The engineer agent's utility function is:

  \[
u_e = -1.43 \cdot 10^{-4} x + 2,
\]

  which is not changed since the engineer agent knows the exact information about its utility function line, (i.e. optimum value point: (7000, 1); and reservation value point: (9800, 0.6)).

  b) **The engineer agent's estimate of the contractor agent utility function**: At this stage, the engineer agent's information about the contractor agent's utility function is: optimum point: (11000, 1); reservation point: (9520, 0.6). Thus, the engineer agent's estimate about the contractor agent's utility function is:

  \[
  u_c = 2.703 \cdot 10^{-4} x - 1.973
\]

  c) **The combined utility function**: 

266
Since:

\[ u_e = -1.43 \times 10^{-4} x + 2 \]
\[ u_c = 2.703 \times 10^{-4} x - 1.973 \]

Thus, the function between the utility functions of the contractor agent and the engineer agent can be calculated as: \( u_e = -0.529 u_c + 0.956 \).

- **Risk evaluation**

In this iteration, the current offers of the contractor agent and the engineer agent are (9975, 8838). Thus, the maximum likelihood of risk acceptability to the engineer agent and the contractor agent are:

\[
P_{e_{\text{max}}} = \frac{U'_{ee} - U'_{ce}}{U'_{ee} - U'_{ce}(C)} = \frac{0.733 - 0.639}{0.733} = 0.129
\]

\[
P_{c_{\text{max}}} = \frac{U'_{ee} - U'_{ce}}{U'_{ee} - U'_{ce}(C)} = \frac{0.6 - 0.422}{0.6} = 0.297
\]

- **Concession**

Since the \( P_{c_{\text{max}}} < P_{e_{\text{max}}} \), the engineer agent knows it should make a concession in the next iteration. The concession step can be calculated as:

\[
P_{c_{\text{max}}} = \frac{U'_{ee} - U'_{ce}}{U'_{ee} - U'_{ce}(C)} = \frac{0.6 - U_{ce}}{0.6} = 0.129
\]

\[ u_{ce} = 0.5226, \Rightarrow u_c = 0.6795 \Rightarrow x = 9232 \]

Thus, the engineer agent's new offer will be equal to, or higher than £9232.
7) The contractor agent’s offer in the 4th iteration

- **Updating the probability of the engineer agent’s reservation value**

After the previous iterations, the contractor agent’s new belief about the engineer agent’s reservation value can be calculated as:

Table 7.9 The contractor agent’s estimate about the probability of the engineer agent’s reservation value at the beginning of 4th iteration

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,000</td>
<td>0</td>
<td>0</td>
<td>0.4949</td>
<td>0.5050</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,000</td>
<td>0</td>
<td>0</td>
<td>0.5050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,500</td>
<td>0</td>
<td>0</td>
<td>0.5050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£9,000</td>
<td>0</td>
<td>0</td>
<td>0.5050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£10,000</td>
<td>0</td>
<td>0</td>
<td>0.5050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£11,000</td>
<td>0</td>
<td>0</td>
<td>0.5050</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Based on this prior knowledge and the engineer’s new counteroffer of £9,232, the contractor agent updates its belief as follows:

\[
P(R_1 | e_1, e_2, e_3) = \frac{P(R_1 | e_1, e_2)P(e_1, e_2, e_3 | R_1)}{\sum_{k=1}^{6} P(e_1, e_2, e_3 | R_k)P(R_k)}
\]

Where,

\[
P(e_1, e_2, e_3 | R_i) = P(e_1 | R_i) * P(e_2 | R_i) * \ldots * P(e_n | R_i)
\]

\[
P(e_1, e_2, e_3 | R_i) = P(e_1 | R_i) * P(e_2 | R_i) * P(e_3 | R_i) = (0.0168 * 0.0306) * (0.03, 0.15) = (0.0005, 0.00459)
\]

Here, it is supposed that the conditional probabilities are independent with each other (Pearl, 1988). In this iteration, the new prior conditional probabilities are:

\[
P(R_1 | e_1, e_2, e_3) = \frac{P(R_1 | e_1, e_2)P(e_1, e_2, e_3 | R_1)}{\sum_{k=1}^{6} P(e_1, e_2, e_3 | R_k)P(R_k)} = \frac{0.4949 * 0.0005}{(0.4949 * 0.0005) + (0.505 * 0.00459)} = 0.096
\]

\[
P(R_2 | e_1, e_2, e_3) = \frac{P(R_2 | e_1, e_2)P(e_1, e_2, e_3 | R_2)}{\sum_{k=1}^{6} P(e_1, e_2, e_3 | R_k)P(R_k)} = \frac{0.505 * 0.00459}{(0.4949 * 0.0005) + (0.505 * 0.00459)} = 0.094
\]
• Estimating the engineer agent’s reservation value

After receiving the new counteroffer, the contractor agent’s current estimation of the engineer agent’s reservation value is: \( R = 0.096 \times 8500 + 0.904 \times 9000 = 8952 \)

• Utility functions

a) The contractor agent’s utility function: \( u_c = 2 \times 10^{-4} \times x - 1.2 \)

b) The contractor agent’s estimate of the engineer agent’s utility function: Based on its updated beliefs about the engineer agent’s reservation value, the contractor agent identifies the two points along the engineer agent’s utility function: optimum point: (7000, 1) and reservation point: (8952, 0.6). Thus, the contractor agent estimates the engineer agent’s utility function is: \( u_e = -2.05 \times 10^{-4} \times x + 2.434 \).

c) The combined utility function

Since:

\[ u_c = 2 \times 10^{-4} \times x - 1.2 \]
\[ u_e = -2.05 \times 10^{-4} \times x + 2.434 \]

Thus, the relationship between the utility functions of the contractor agent and the engineer agent can be expressed as: \( u_e = -1.025u_c + 1.204 \).

• Risk evaluation

In this iteration, the current offers of the contractor agent and the engineer agent are (9775, 9232). Thus, the maximum likelihood of risk acceptable to the contractor is:
\[ P_{\text{c max}} = \frac{U_\text{cc} - U_\text{ce}}{U_\text{cc} - U_\text{c}(C)} = \frac{0.755 - 0.6464}{0.755} = 0.143 \]
\[ P_{\text{e max}} = \frac{U_\text{ee} - U_\text{ec}}{U_\text{ee} - U_\text{e}(C)} = \frac{0.541 - 0.43}{0.541} = 0.205 \]

- **Concession**

Since the \( P_{\text{c max}} < P_{\text{e max}} \), the contractor agent knows it should make a concession in the next iteration. The concession step can be calculated as:

\[ P_{\text{e max}} = \frac{U(w_e) - U(D_e)}{U(w_e) - U(e)} = \frac{|0.541 - u_{\text{ec}}|}{0.541 - 0} = 0.143 \]

\( u_{\text{ec}} = 0.4636 \Rightarrow x = 9612 \)

Thus, the contractor agent's new offer will be equal to, or lower than \£9612.

8) The engineer agent's counter-offer in the 4th iteration

After the last three encounters, the engineer agent's prior probability about the contractor agent's reservation value is shown in Table 7.10.

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypothesis</td>
<td>£7,500</td>
<td>£8,000</td>
<td>£9,000</td>
<td>£10,000</td>
<td>£11,000</td>
</tr>
<tr>
<td>Probability</td>
<td>0</td>
<td>0</td>
<td>0.48</td>
<td>0.52</td>
<td>0</td>
</tr>
</tbody>
</table>

The condition probability is:

\[ P(e_1, e_2, e_3 | R_i) = P(e_1 | R_i) * P(e_2 | R_i) * P(e_3 | R_i) = (0.1099, 0.1375) * (0.3612, 0.303) = (0.0397, 0.0417) \]
After receiving the counteroffer (£9612), the engineer agent’s current estimation of the contractor agent’s reservation value is: \( R = 0.468 \times 9000 + 0.532 \times 10000 = 9532 \).

- **Utility functions**

  a) *The engineer agent’s utility function*: \( u_e = -1.43 \times 10^{-4} x + 2 \)

  b) *The engineer agent’s estimate of the contractor agent utility function*: At this stage, the engineer agent’s information about the contractor agent’s utility function is: optimum point: (11000, 1) and reservation point: (9532, 0.6). Thus, the contractor agent estimates the engineer agent’s utility function is: \( u_c = 2.725 \times 10^{-4} x - 1.9973 \).

  c) *The combined utility function*

Since:

\[
\begin{align*}
    u_e &= -1.43 \times 10^{-4} x + 2 \\
    u_c &= 2.725 \times 10^{-4} x - 1.9973
\end{align*}
\]

Thus, the relationship between the utility functions of the contractor agent and the engineer agent can be expressed as: \( u_e = -0.525u_c + 0.952 \).
• Risk evaluation

In this iteration, the current offers of the contractor agent and the engineer agent are \((9612, 9232)\). Thus, the maximum likelihood of risk acceptable to the contractor is:

\[
P_{e_{\text{max}}} = \frac{U_{i_{ee}} - U_{i_{ce}}}{U_{i_{ee}} - U_{e}(C)} = \frac{0.68 - 0.625}{0.68} = 0.081
\]

\[
P_{c_{\text{max}}} = \frac{U_{i_{cc}} - U_{i_{ce}}}{U_{i_{cc}} - U_{c}(C)} = \frac{0.622 - 0.518}{0.622} = 0.167
\]

• Concession

Since the \(P_{e_{\text{max}}} < P_{c_{\text{max}}}\), the engineer agent knows it should make a concession in the next iteration. The concession step can be calculated as:

\[
P_{c_{\text{max}}} = \frac{U_{i_{cc}} - U_{i_{ce}}}{U_{i_{cc}} - U_{c}(C)} = \frac{0.622 - 0.518}{0.622} = 0.167
\]

\[u_{ce} = 0.572, \Rightarrow u_{c} = 0.6517 \Rightarrow x = 9429\]

Thus, the engineer agent's new offer will be equal to, or higher than £9429.

9) The contractor agent's offer in the 5th iteration

After receiving the engineer agent's new counteroffer, the contractor agent starts its new evaluation process to decide what it should offer for the next iteration.

• Updating the probability of the engineer agent's reservation value

After the above iterations, the contractor agent's new belief about the engineer agent's reservation value are shown in Table 7.11.
Table 7.11 The contractor agent’s estimate about the probability of the engineer agent’s reservation value at the beginning of 5th iteration

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,000</td>
<td>£8,000</td>
<td>£8,500</td>
<td>£9,000</td>
<td>£10,000</td>
<td>£11,000</td>
<td></td>
</tr>
</tbody>
</table>

| Probability P(R_i | e_1, e_2, e_3) | 0     | 0     | 0.096 | 0.904 | 0     | 0     |

Based on this prior knowledge and the engineer’s new counteroffer £9429, the contractor updates his belief as follows:

\[
P(R_1 | e_1, e_2, e_3, e_4) = \frac{P(R_1 | e_1, e_2, e_3)}{\sum_{k=1}^{6} P(e_1, e_2, e_3, e_4 | R_k)P(R_k)}
\]

Where,

\[
P(e_1, e_2, ..., e_n | R_i) = P(e_1 | R_i) * P(e_2 | R_i) * ... * P(e_n | R_i)
\]

\[
P(e_1, e_2, e_3, e_4 | R_i) = P(e_1 | R_i) * P(e_2 | R_i) * P(e_3 | R_i) * P(e_4 | R_i) = (0.0005, 0.00459)(0.01568, 0.0784) = (7.84 * 10^{-6}, 3.6 * 10^{-4})
\]

It is supposed that the conditional probabilities are independent with each other (Pearl, 1988). In this iteration, the new prior conditional probabilities are:

\[
P(R_1 | e_1, e_2, e_3, e_4) = \frac{P(R_1 | e_1, e_2, e_3, e_4)}{\sum_{k=1}^{6} P(e_1, e_2, e_3, e_4 | R_k)P(R_k)} = 0.0023
\]

\[
P(R_2 | e_1, e_2, e_3, e_4) = \frac{P(R_2 | e_1, e_2, e_3, e_4)}{\sum_{k=1}^{6} P(e_1, e_2, e_3, e_4 | R_k)P(R_k)} = 0.9977
\]

- **Estimating the engineer agent’s reservation value**

Therefore, after receiving the new counteroffer, the contractor agent’s current estimation of the engineer agent’s reservation value is: \( R = 0.0023*8500 + 0.9977*9000 = 8999 \).
• **Utility function calculation**

a) *The contractor agent's utility function:* \( u_c = 2 \times 10^{-4} x - 1.2 \)

b) *The contractor agent's estimate of the engineer agent's utility function:* At this stage, the two points which the contractor agent has are: optimum point: \((7000, 1)\) and reservation point: \((8999, 0.6)\). Thus, the contractor agent estimates the engineer agent's utility function is: \( u_e = -2 \times 10^{-4} x + 2.4 \).

c) *The combined utility function*

Since,
\[
u_c = 2 \times 10^{-4} x - 1.2;
\]
\[
u_e = -2 \times 10^{-4} x + 2.4;
\]

Thus, the relationship between the utility functions of the contractor agent and the engineer agent can be expressed as: \( u_e = -u_c + 1.2 \).

• **Risk evaluation**

In this iteration, the current offers of the contractor agent and the engineer agent are \((9612, 9429)\). Thus, the maximum likelihood of risk acceptable to the contractor is:

\[
P_{\max} = \frac{U_{\text{cc}}^i - U_{\text{ce}}^i}{U_{\text{cc}}^i - U_{\text{ee}}^i (C)} = \frac{0.7224 - 0.6858}{0.7224} = 0.0366
\]

\[
P_{e \max} = \frac{U_{\text{ce}}^i - U_{\text{ee}}^i}{U_{\text{ce}}^i - U_{\text{ee}}^i (C)} = \frac{0.514 - 0.478}{0.514} = 0.07
\]
• Concession

Since the $P_{\text{cmax}} < P_{\text{e}}$, the contractor agent knows it should make a concession in the next iteration. The concession step can be calculated as:

$$P_{e_{\text{max}}} = \frac{U(w_e) - U(D_c)}{U(w_e) - U(e)} = \left| \frac{0.514 - u_{ec}}{0.514 - 0} \right| = 0.0366 ,$$

$$u_{ec} = 0.495, \Rightarrow u_c = 0.705 \Rightarrow x = 9524$$

Thus, the contractor agent’s new offer will be equal to or lower than £9524.

10) The engineer agent’s counter-offer in the 5th iteration

Based on the last four encounters, the engineer agent’s prior probability about the contractor agent’s reservation value is shown in Table 7.12.

Table 7.12 The engineer agent’s estimate about the probability of the contractor agent’s reservation value at the beginning of 5th iteration

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>$R_1$</th>
<th>$R_2$</th>
<th>$R_3$</th>
<th>$R_4$</th>
<th>$R_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,500</td>
<td>£8,000</td>
<td>£9,000</td>
<td>£10,000</td>
<td>£11,000</td>
<td></td>
</tr>
<tr>
<td>Probability</td>
<td>$P(R_1</td>
<td>e_1, e_2, e_3)$</td>
<td>0</td>
<td>0</td>
<td>0.468</td>
</tr>
</tbody>
</table>

Thus, the condition probability is:

$$P(e_1, e_2, e_3, e_4 | R_1) = P(e_1 | R_1) \cdot P(e_2 | R_1) \cdot P(e_3 | R_1) \cdot P(e_4 | R_1) = (0.0397, 0.0417) \cdot (0.358, 0.295) = (0.0142, 0.0162)$$

$$P(R_1 | e_1, e_2, e_3, e_4) = \frac{P(R_1 | e_1, e_2, e_3) P(e_1, e_2, e_3, e_4 | R_1)}{\sum_{i=1}^{5} P(e_1, e_2, e_3, e_4 | R_1) P(R_1)} = \frac{0.468 \cdot 0.0142}{(0.468 \cdot 0.0142) + (0.532 \cdot 0.0123)} = 0.504$$

$$P(R_2 | e_1, e_2, e_3, e_4) = \frac{P(R_2 | e_1, e_2, e_3) P(e_1, e_2, e_3, e_4 | R_2)}{\sum_{i=1}^{5} P(e_1, e_2, e_3, e_4 | R_2) P(R_2)} = \frac{0.532 \cdot 0.0123}{(0.468 \cdot 0.0142) + (0.532 \cdot 0.0123)} = 0.496$$

275
After receiving the counteroffer (£9524), the engineer agent’s current estimation of the contractor agent’s reservation value is: \[ R = 0.504 \times 9000 + 0.496 \times 10000 = 9496. \]

- **Utility functions**

  a) *the engineer agent’s utility function:* \[ u_e = -1.43 \times 10^{-4} x + 2. \]

  b) *The engineer agent’s estimate of the contractor agent utility function:* At this stage, the engineer agent’s information about the contractor agent’s utility function is: optimum point: (11000, 1) and reservation point: (9496, 0.6). Thus, the contractor agent estimates the engineer agent’s utility function is: \[ u_c = 2.66 \times 10^{-4} x - 1.926. \]

  c) *The combined utility function*

Since,
\[ u_e = -1.43 \times 10^{-4} x + 2 \]
\[ u_c = 2.66 \times 10^{-4} x - 1.926 \]

Thus, the relationship between the utility functions of the contractor agent and the engineer agent can be expressed as: \[ u_e = -0.525u_c + 0.952. \]

- **Risk evaluation**

In this iteration, the current offers of the contractor agent and the engineer agent are (9524, 9429). Thus, the maximum likelihood of risk acceptable to the contractor is:

\[
P_{e \text{ max}} = \frac{U_C - U_C}{U_C - U_C(C)} = \frac{0.652 - 0.63}{0.652} = 0.0337
\]
\[
P_{e \text{ max}} = \frac{U_C - U_C}{U_C - U_C(C)} = \frac{0.622 - 0.582}{0.622} = 0.064
\]
Concession

Since the $P_{e_{\text{max}}} < P_{c_{\text{max}}}$, the engineer agent knows it should make a concession in the next iteration. The concession step can be calculated as:

$$P_{e_{\text{max}}} = \frac{U_{c}^{t_{e}} - U_{c}^{t_{e}}} {U_{c}^{t_{e}} - U_{c}^{t_{e}}(C)} = \frac{0.622 - U_{c}^{t_{e}}} {0.622} = 0.0337$$

$u_{ce} = 0.601$, $\Rightarrow u_{c} = 0.642 \Rightarrow x = 9500$

Thus, the engineer agent's new offer will be equal to, or higher than £9500.

Table 7.13 and Figure 7.2 shows the above negotiation process. It can be seen that the negotiation will converge at the next few encounters. However, the speed of convergence slows down when the negotiation nears an agreement. Practically, it may be assumed that an agreement is reached when the difference between both parties' offers is small enough (e.g. £50 or within 1% of the offer amount).

Table 7.13 The negotiation process of example 1

<table>
<thead>
<tr>
<th></th>
<th>Initial offer</th>
<th>2\textsuperscript{nd} iteration</th>
<th>3\textsuperscript{rd} iteration</th>
<th>4\textsuperscript{th} iteration</th>
<th>5\textsuperscript{th} iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>The contractor agent</td>
<td>11,000</td>
<td>10,418</td>
<td>9,975</td>
<td>9,612</td>
<td>9,524</td>
</tr>
<tr>
<td>The engineer agent</td>
<td>7,000</td>
<td>8,838</td>
<td>9,232</td>
<td>9,429</td>
<td>9,500</td>
</tr>
</tbody>
</table>

\textsuperscript{23} The unit of vertical axis is 100.
7.6 OTHER CASES AND DISCUSSION

The above example shows the negotiation process where a simple case that the MASCOT model can address. To enable better understanding of the MASCOT prototype, few further examples are presented below, as variations of Example 1.

7.6.1 Negotiation with Time Penalty (Example 2)

In this case, time penalties are incorporated. It is assumed that the time penalty for the contractor agent is 3% utility reduction per iteration and 5% for the engineer agent. By following the same procedure as case 1, the negotiation result is shown in the following
An agreement is reached after the contractor makes its offer in the third iteration under the pressure of time penalty (Table 7.14 and Figure 7.3).

**Table 7.14 The negotiation process with time penalty (Example 2)**

<table>
<thead>
<tr>
<th>Time penalty</th>
<th>Initial offer</th>
<th>1st iteration</th>
<th>2nd iteration</th>
<th>3rd iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The contractor agent</strong></td>
<td>3% deduction of current utility</td>
<td>11,000</td>
<td>10,285</td>
<td>9,730</td>
</tr>
<tr>
<td><strong>The engineer agent</strong></td>
<td>5% deduction of current utility</td>
<td>7,000</td>
<td>9,143</td>
<td>9,674</td>
</tr>
</tbody>
</table>

**Figure 7.3 The negotiation with time penalty (Example 2)**
7.6.2 Negotiation with a Conflict Outcome (Example 3)

In this case, it is assumed that the reservation value of the engineer agent is reduced from £9800 to £8800. Obviously, there is no agreement zone between the contractor agent and the engineer agent since their reservation values are £8800 and £9000. Table 7.15 shows such a negotiation process using the MASCOT model. A conflict deal is reached in the third iteration (Figure 7.4).

Table 7.15 The negotiation process with a conflict outcome (Example 3)

<table>
<thead>
<tr>
<th></th>
<th>Initial offer</th>
<th>1st iteration</th>
<th>2nd iteration</th>
<th>3rd iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>The contractor agent</td>
<td>11,000</td>
<td>1,0418</td>
<td>9,596</td>
<td>9,210</td>
</tr>
<tr>
<td>The engineer agent</td>
<td>7,000</td>
<td>8,008</td>
<td>8,515</td>
<td><strong>8978 &gt; 8800</strong></td>
</tr>
</tbody>
</table>

Figure 7.4 The negotiation reaches a conflict deal (Example 3)
7.6.3 Negotiation with Random Conditional Probabilities (Example 4 and 5)

In practice, a negotiation party may not have more knowledge about his opponent. Therefore, the probability distribution and conditional probability encoded from his prior knowledge may be unreasonable. In this case, we assume two cases with extreme conditional probability (Example 4 and 5). The data are shown in the following tables.

Table 7.16 The contractor's prior knowledge about the conditional probabilities of the engineer's offer (Example 4)

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>£6,000</th>
<th>£7,000</th>
<th>£8,000</th>
<th>£8,500</th>
<th>£9,000</th>
<th>£9,500</th>
<th>£10,000</th>
<th>£11,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,000</td>
<td>0.35</td>
<td>0.4</td>
<td>0.2</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,000</td>
<td>0.1</td>
<td>0.4</td>
<td>0.35</td>
<td>0.12</td>
<td>0.03</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,500</td>
<td>0</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td>0.05</td>
<td>0.15</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£9,000</td>
<td>0</td>
<td>0.3</td>
<td>0.25</td>
<td>0.25</td>
<td>0.2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£10,000</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.20</td>
<td>0.45</td>
<td>0.25</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td>£11,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.1</td>
<td>0.25</td>
<td>0.40</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Note: only rows £8,500 and £9,000 are modified since the data in other rows do not work in this case. The distribution of £8,500 is abnormal, while the distribution of £9,000 is shifted forward.

Table 7.17 The contractor's prior knowledge about the conditional probabilities of the engineer's offer (Example 5)

<table>
<thead>
<tr>
<th>Hypothesis</th>
<th>£6,000</th>
<th>£7,000</th>
<th>£8,000</th>
<th>£8,500</th>
<th>£9,000</th>
<th>£9,500</th>
<th>£10,000</th>
<th>£11,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>£7,000</td>
<td>0.35</td>
<td>0.4</td>
<td>0.2</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,000</td>
<td>0.1</td>
<td>0.4</td>
<td>0.35</td>
<td>0.12</td>
<td>0.03</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,500</td>
<td>0</td>
<td>0.2</td>
<td>0.3</td>
<td>0.3</td>
<td>0.05</td>
<td>0.15</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£9,000</td>
<td>0</td>
<td>0.3</td>
<td>0.15</td>
<td>0.15</td>
<td>0.1</td>
<td>0.3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£10,000</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.20</td>
<td>0.45</td>
<td>0.25</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td>£11,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.1</td>
<td>0.25</td>
<td>0.40</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Note: in this case, the distribution of £9,000 is high at two side and low at middle.
Table 7.18 The engineer's prior knowledge about the conditional probabilities of the contractor's offer (Example 4, 5)

<table>
<thead>
<tr>
<th>Possible event</th>
<th>e₁</th>
<th>e₂</th>
<th>e₃</th>
<th>e₄</th>
<th>e₅</th>
<th>e₆</th>
<th>e₇</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypothesis</td>
<td>£7,500</td>
<td>£8,000</td>
<td>£8,500</td>
<td>£9,000</td>
<td>£10,000</td>
<td>£11,000</td>
<td>£12,000</td>
</tr>
<tr>
<td>£7,500</td>
<td>0.35</td>
<td>0.5</td>
<td>0.1</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£8,000</td>
<td>0.15</td>
<td>0.35</td>
<td>0.4</td>
<td>0.1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£9,000</td>
<td>0</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>£10,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.333</td>
<td>0.333</td>
<td>0.334</td>
<td>0</td>
</tr>
<tr>
<td>£11,000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Note: this is pure even distribution

The results of Example 4 and 5 are shown in Figure 7.5 and Figure 7.6.

In Example 4, the likelihood distribution in Table 7.16 shows that the contractor agent clearly understand that the engineer agent's offer will not be higher than £9,000, when the engineer agent's real reservation value is £9,000. By chance, this assumed distribution is more accurate than the original case. Thus, it converges much faster than the original case (Table 7.19 and Figure 7.5).

Table 7.19 The negotiation process with random conditional probability (Example 4)

<table>
<thead>
<tr>
<th>Initial offer</th>
<th>1st iteration</th>
<th>2nd iteration</th>
<th>3rd iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>The contractor agent</td>
<td>11,000</td>
<td>1,0418</td>
<td>9,910</td>
</tr>
<tr>
<td>The engineer agent</td>
<td>7,000</td>
<td>8,926</td>
<td>9,517 (&gt; the contractor agent's reservation value)</td>
</tr>
</tbody>
</table>

Note: although the contractor agent's calculated offer 8926 is smaller than the reservation value 9000 in 3rd iteration, this is different from the case conflict deal. In conflict deal, party A's offer is smaller than its reservation value while party B does not offer any amount which can satisfy party A's reservation value.
In Example 5, the unreasonable high value of conditional probability $P(9500|9000)$ (Table 7.17) causes the contractor agent reduce its concession speed at this point. However, the negotiation is still converging, and finally will reach an agreement. The negotiation process is shown in Table 7.20 and Figure 7.6.

Figure 7.5 The negotiation with random conditional probability (Example 4)
### Table 7.20 The negotiation process with random conditional probability (Example 5)

<table>
<thead>
<tr>
<th></th>
<th>Initial offer</th>
<th>1\textsuperscript{st} iteration</th>
<th>2\textsuperscript{nd} iteration</th>
<th>3\textsuperscript{rd} iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The contractor agent</strong></td>
<td>11,000</td>
<td>1,0418</td>
<td>9,910</td>
<td>9,770</td>
</tr>
<tr>
<td><strong>The engineer agent</strong></td>
<td>7,000</td>
<td>8,926</td>
<td>9,517</td>
<td>9,662</td>
</tr>
</tbody>
</table>

![Diagram](image.png)

**Figure 7.6 The negotiation with random conditional probability (Example 5)**

#### 7.6.4 Negotiation through Simple Gradient Descent With Time Penalty (Example 6)

To make a comparison with the MASCOT model, two examples adopting simple gradient descent mechanism with consideration of time penalty are discussed. The negotiation mechanism in this case is simple and straightforward. However, the
concession approach is far from real claims negotiation. The negotiation simply depends on the selection of concession rate or time penalty. The calculation is shown as follows:

- Critical negotiation features

**Table 7.21 The contractor agent’s major negotiation figures in Example 6**

<table>
<thead>
<tr>
<th>Reservation value</th>
<th>Optimum value</th>
<th>Time penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>£9,000</td>
<td>£11,000</td>
<td>£50/per iteration</td>
</tr>
</tbody>
</table>

**Table 7.22 The engineer agent’s major negotiation figures in Example 6**

<table>
<thead>
<tr>
<th>Reservation value</th>
<th>Optimum value</th>
<th>Time Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>£9,800</td>
<td>£7,000</td>
<td>£80/per iteration</td>
</tr>
</tbody>
</table>

- Negotiation Mechanism

Monotonic concession protocol is adopted. Meanwhile, both parties will adopt a simple negotiation strategy: both parties have fixed concession rates 1.5%, and make concessions alternatively until an agreement is reached.

- Negotiation process

a) **The 1st iteration:** Contractor agent’s initial offer is of £11,000. The engineer agent’s initial counteroffer is of £7,000.

b) **The 2nd iteration:** The contractor agent checks the engineer agent’s offer. Since the offer 7000 is lower than its reservation value, the contractor agent makes its counteroffer of 11000*(1-1.5%)=10835. Considering the time penalty, the real value that the contractor agent can get is 10835-50= 10785 at this stage. Similarly, since the contractor agent’s offer 10835 is larger than the engineer agent’s reservation value 9800, the engineer agent will make a concession 7000*1.5%=105 in its new offer. Therefore, its new offer is 7105. The real value that the engineer agent can get is 7105+80=7185 at this stage.
The whole negotiation process is shown in Table 7.23. From the table, it can be seen that the contractor agent's offer can satisfy the engineer agent's reservation value in the 7th iteration. It means the engineer agent might accept the contractor agent's offer and reach an agreement at £9757. Otherwise, the negotiation will go on, and an agreement will be reached in 11th iteration at a price of £8989 or £8980.

**Table 7.23 The negotiation process with same concession rate (Example 6)**

<table>
<thead>
<tr>
<th>Iteration</th>
<th>The contractor agent</th>
<th>The engineer agent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Concession amount</td>
<td>New offer</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>11000</td>
</tr>
<tr>
<td>2</td>
<td>165</td>
<td>10835</td>
</tr>
<tr>
<td>3</td>
<td>162</td>
<td>10623</td>
</tr>
<tr>
<td>4</td>
<td>159</td>
<td>10415</td>
</tr>
<tr>
<td>5</td>
<td>155</td>
<td>10209</td>
</tr>
<tr>
<td>6</td>
<td>152</td>
<td>10007</td>
</tr>
<tr>
<td>7</td>
<td>149</td>
<td>9807</td>
</tr>
<tr>
<td>8</td>
<td>146</td>
<td>9611</td>
</tr>
<tr>
<td>9</td>
<td>143</td>
<td>9418</td>
</tr>
<tr>
<td>10</td>
<td>141</td>
<td>9227</td>
</tr>
<tr>
<td>11</td>
<td>138</td>
<td>9039</td>
</tr>
<tr>
<td>12</td>
<td>135</td>
<td>8855</td>
</tr>
</tbody>
</table>

The drawback of the above negotiation process is the agent who has a higher initial offer will always make more concession since both parties' concession rates are same. To overcome this problem, we assume that the concession amount of agents is same (with different concession rates). Suppose, the concession step is £50/per iteration, and the time penalty for the contractor agent is £50 and £80 for the engineer agent per iteration. The negotiation result is shown in Table 7.24.
In this case, the contractor agent's offer satisfies the engineer agent's reservation value in the 13th iteration. An agreement is possible to be reached if the engineer agent is willing to accept such an amount. Otherwise, an agreement will be reached in the 18th iteration at an amount around £9250.
7.7 DISCUSSION

The above demonstration of the system operation shows that the MASCOT negotiation mechanism works properly in different situations (Examples 1-5). Also, the results achieved in these situations were reasonable, which is an essential requirement for all negotiation systems. However, it is not easy to achieve. For example, the simple gradient descent approach, adopted by many negotiation systems, can seldom reflect real cases. Negotiation may be very inefficient if an agent proposes a high initial offer (Example 6). In other cases, an agent may benefit from an unreasonably high initial offer if a binary divisive approach is adopted. Since the MASCOT negotiation mechanism is based on a thorough analysis of negotiation theories and the characteristics of claims negotiation, the learning approach and the risk avoidance model ensure that negotiation can be conducted reasonably. The problem in binary divisive mechanism will be reduced because an agent may have prior knowledge about the opponent's approach to negotiation. Thus, no matter how high the opponent offers, the agent will make a reasonable counter-offer based on its domain knowledge about the opponent. The large number of claims in a construction project makes it possible for a party to obtain enough domain knowledge. On the other hand, if an agent cannot get enough prior knowledge about its opponent, the negotiation can still converge and reach a reasonable result (shown in Examples 4 and 5) since the agent can make its decision according to the opponent's offers during negotiation.

With consideration of its particular advantages, the various aspects of the MASCOT negotiation mechanism have great potential to be further applied in other domains. For example:

- Modified Monotonic Concession Protocol (mMCP): MCP is a general negotiation protocol which could be used in many different areas, however, its assumptions such as complete information and conflict deal limits its application (see Chapter 5). mMCP relaxes some of these assumptions. For example, it changes the situations for conflict deal; it also takes consideration of the time issue, and the involvement of the client during negotiation. These improvements allow mMCP to be applied more
effectively in multi-agent negotiation systems where time, conflict situations, and the involvement of new agents are generally important.

- Zeuthen’s negotiation strategy, Bayesian learning approach, and the integration of the two: These major components of the MASCOT negotiation strategy could be effective negotiation strategies when they are applied to environments, where risk acceptability is crucial for negotiators to make their decisions; or when it is essential for a negotiator to learn about the others’ approaches to negotiation. These situations are particularly meaningful for the application of multi-agent systems in construction because the construction industry is characterised as risky, complex and dynamic. In application environments such as: collaborative design, supply chain management, plan and schedule monitoring, and material management, an agent’s learning ability is particularly important for it to learn from the other specialist agents through interactions. Furthermore, the overall MASCOT negotiation mechanism could be adopted in facilitating or mediating the negotiations between project participants in large-scale civil engineering projects, where Pena-Mora and Wang (1998) have developed a simple MAS negotiation mechanism based on game theory. The adoption of the MASCOT negotiation mechanism will greatly enhance the capability of their negotiation system.

- Expanded-solution searching strategy: the key aspects of this strategy such as the involvement of a third agent (e.g. the client agent or a mediator agent), the trade-offs between different negotiation items, and the involvement of the agents’ owners are generally useful for the development of multi-agent systems. The adoption of these strategies is not limited to agent negotiation systems, but could be adopted or further developed in various agent collaboration systems.
7.8 SUMMARY

This chapter has presented the system operation process. The key negotiation strategies of the MASCOT conceptual model were demonstrated using a practical problem. The utility function was first defined, and negotiation preparation process was described. Then, the key stages of negotiation processes were presented using several examples. The rational of results were discussed. Also, the advantages of the negotiation mechanism were shown by comparing the solution provided with that based on a simple gradient descent approach. The examples serve to aid understanding of the operation of the system and illustration of the key features of the system. The next chapter will discuss the evaluation of the system in terms of its negotiation mechanism and prototype software.
CHAPTER 8
SYSTEM EVALUATION

8.1 INTRODUCTION

System evaluation is an integral part of the model development. This chapter describes the system evaluation process which includes two major steps: theoretical evaluation and prototype assessment. It first introduces the objectives of system evaluation, and explains the methodology adopted for the evaluation. A theoretical evaluation of the MASCOT negotiation mechanism is then carried out. Section 8.5 presents the prototype evaluation process including the selection of the evaluation group, design of the questionnaire, and analysis of the responses. All the major aspects of the system are discussed in this section. These include the suitability of the system, its industry background, the features of the system, and perspectives for the application of MAS in other construction fields. The results of the evaluation and the evaluation method are also discussed.

8.2 EVALUATION OBJECTIVES

The aim of system evaluation is to determine how well the developed MASCOT system facilitates construction claims negotiation. The particular objectives are:

- To assess the quality of the MASCOT negotiation mechanism. Like many other MAS research projects (Kraus, 1993, 1996, 1997; Rosenschein and Zlotkin, 1994), it is essential to analyse the quality of the negotiation mechanism of a developed system in terms of a number of key evaluation criteria, which include:
  a) Suitability;
  b) Simplicity;
  c) Distribution;
d) Efficiency;
e) Stability; and
f) Symmetry.

- **To assess the performance of the MASCOT prototype.** Unlike the system evaluations performed in most of the current MAS research projects where a theoretical analysis of negotiation mechanism is deemed adequate, an industry-oriented prototype assessment is necessary because this project focuses more on the development and application of the system to real industry problems. The prototype evaluation was designed to assess the efficiency and effectiveness of the MASCOT prototype in supporting claims negotiation. A number of questions to be addressed include:

  a) Is the analysis of the industry background sound for the MASCOT system bases?
  b) Do the key features of the system reflect the major characteristics of construction claims negotiation?
  c) Is the system coming up with reasonable answers?
  d) What are the major problems of the system?
  e) How could the system be improved?
  f) What are the major limitations and difficulties for the application of MAS in the industry?

8.3 METHODOLOGY

To achieve the system evaluation objectives described above, different approaches have been taken. The details are discussed as follows:

8.3.1 Theoretical Evaluation

To achieve the first objective, a theoretical approach was adopted to analyse the quality of the MASCOT negotiation algorithm. Since the MASCOT negotiation mechanism was developed based on both the economic and behaviour theories, some of the key issues
required by these theories must be satisfied. Kraus (1993, 1996, 1997), and Rosenschein and Zlotkin (1994) have defined a number of important evaluation criteria, such as efficiency, stability and suitability, by which the quality of the negotiation mechanism of a MAS could be assessed. A successful MAS negotiation system should satisfy the key criteria. Such a theoretical analysis is commonly used in the MAS community for assessing the developed negotiation mechanism.

8.3.2 Prototype Evaluation

To achieve the second objective, a prototype evaluation approach was adopted. In this approach, the major characteristics of MAS, the MASCOT development process, and the MASCOT prototype were demonstrated to an evaluation group, and the evaluators' feedback was obtained on the appropriateness of the application of MAS to construction claims negotiation would be. A few major issues related to this are discussed below:

8.3.2.1 Evaluation Techniques

Different approaches have been adopted to evaluate different systems (or prototypes). Miles et al (2000) describe two evaluation methods for knowledge-based systems, which could be extended for general evaluations. One of them is to provide industry users with a working version of the system under evaluation and, leave them to make use of it over a prolonged time period (e.g. a number of weeks). This would give project personnel an opportunity to get used to its functionality and form an opinion on whether the stated benefits are actually achieved. A diary is used to provide a record of usage by the evaluators, and includes information on any difficulties that occurred and any features that are felt to be lacking. This is certainly not an easy task as there are many difficulties involved with carrying out a trial in a working environment. The difficulties are magnified when the trial system must be integrated with existing systems. Another approach is where a relatively large number of evaluators are available for a short period of time. An evaluation session is held in a single location with all the evaluators participating simultaneously. The session consists of a hands-on usage portion, where the
evaluators are guided through a usage scenario with the use of appropriate notes. This is followed by the distribution and completion of a questionnaire by each evaluator.

Of these two evaluation methods, the second approach (i.e. the prototype evaluation approach) was chosen. A demonstration was considered adequate to provide the evaluators with the necessary exposure to the main concepts. The chosen method means that the evaluation could be carried out with several people at one time - more people than could have been included had individual evaluation sessions been carried out. More importantly, since the study on MAS, especially for its application in construction, is quite new, the most important thing at this stage is to gain an understanding of whether the MAS approach is appropriate and applicable to the industry. There would be little to gain at the current stage from having the industry users try out the system because an immature prototype can hardly provide industry users with satisfactory solutions to real problems and fully present the advantages of the techniques. In fact, much work needs to be done to build a more realistic and robust negotiation system which could be ready for the industrial development.

The drawback of the demonstration and prototype approach is that its effectiveness is sometimes limited due to the short evaluation session with a pre-determined set of data. Also, it is very difficult to assess the problems generated beyond a normal working environment. The effectiveness of the system in meeting these goals would need to be addressed in later evaluation processes.

8.3.2.2 Choice of Evaluators

A sample group of nine people from both the industry and academia were involved in the prototype evaluation. These evaluators had various backgrounds in the industry and experience of construction claims (i.e. the participants represented all the claims parties, such as: client, contractor, engineer or architecture; and various positions in their organisation, such as: project engineer, project manager, quantity surveyor, or IT developer). Importantly, they also had some knowledge of IT and its applications in the
industry. Although in no way statistically significant, it was expected that their opinions would be indicative of how industry practitioners would view the system. As Eason (1988) states that “for a scientific study it should be necessary to obtain numbers for the sample to be representative of all level and types of users. An evaluation for practical rather than scientific purpose may not need to be rigorous about numbers but it should include the full range of users.”

8.3.2.3 Evaluation Questionnaire

The evaluation questionnaire is an important part of the prototype evaluation. It consisted of five major parts that roughly correspond to various aspects of the evaluation objectives, such as the system environment, feature, functions, and further improvement (see Appendix A). It included questions about:

- the background of the respondents in terms of the number of years of practical experience in the industry, the role(s) they have held, and the area(s) of experience;
- the essential characteristics and features of construction claims negotiation on which the MASCOT system is based;
- the adoption of multi-agent systems to support claims negotiation;
- the various aspects of the MASCOT model; and
- the MASCOT model in general, and possible improvements.

These questions were designed to allow respondents to express their ideas in both quantitative and qualitative manners. Quantitative questions consisted of a number of statements to which the respondents could express their level of agreement or disagreement by circling a number on a five-point scale. The five-point scale was chosen as it was deemed to provide a sufficient range of responses without being overly complex. Respondents were also allowed to make specific comments via open questions.

---

24 A pilot prototype demonstration was conducted with a small sample group of industry participants who had very little knowledge of the applications of IT in the industry. During the evaluation session, considerable effort had to be made to introduce the background knowledge to the participants for them to further evaluate the system. Therefore, it was considered more effective to involve the evaluators who had knowledge of the industry practice and the application of IT in the industry.
Thus, the evaluators' responses were not limited by the format of questions proposed by the system developer.

This mixed approach is endorsed by Eason (1988) who states that "structured questions have the virus of easy analysis and direct comparability. Their weakness is that they pre-defined the answers it is possible to give and may not therefore permit the user to report the most important issues. We have always found it useful to use a structured approach to reveal issues and, once an issue is located, to use an unstructured method to explore the nature of the issues".

8.3.2.4 Discussions

Besides the questionnaire, a short discussion session was also carried out after the evaluation to explain some respondents' questions, and gain more evaluators' feedback on the prototype. This fitted in well with the demonstration approach since they are both appropriate for use with a group of people. Although fuller responses may have been gained from individual interviews or smaller discussion groups, these would have been more time-consuming. For this reason, the method chosen was deemed to be adequate.

8.4 EVALUATION OF THE MASCOT NEGOTIATION MECHANISM

8.4.1 Criteria for Effective Assessment System

Different negotiation systems will bring different qualities to the overall system. There are several well-accepted criteria in evaluating negotiation quality (Kraus, 1995, 1996; Rosenschein and Zlotkin, 1994); these include:

- **Suitability**: the system should best reflect the characteristics of the negotiation scenarios, and could generate similar or reasonable negotiation outcomes as human negotiations (to a certain degree).

- **Simplicity**: the negotiation process should be short and consume only a reasonable amount of communication and computation resources. The simpler the system the
easier it is to make it efficient if effort is not being wasted on complicated communication. It also makes the process more transparent.

- **Distribution:** the decision making process should be distributed. There should be no central unit or agent that manages the problem. The level of distribution of agents, not only physically but also their authority is a key issue. There are issues such as security and simplicity that may impose a central arbiter. However, the greatest performance benefits will come from a fully distributed system.

- **Efficiency:** this essentially means the agents as a group arrive at the most optimal solution. This has been defined in many ways. One might be when the sum of the agents' benefits is maximised. More commonly accepted is the status of Pareto Optimality, where no agent could derive more from a different agreement without some other agent deriving less.

- **Stability:** there should be no incentive for agents to deviate from agreed negotiation strategies. In a pure game situation, with complete information, the Nash equilibrium concept is used. A pair of strategies \((\sigma, \tau)\) is in the Nash Equilibrium if, given \(\tau\), no strategy of Agent 1 results in an outcome that Agent 1 prefers to the outcome generated by \((\sigma, \tau)\), and similarly for Agent 2, given \(\sigma\) (Nash, 1950). If Nash equilibrium is reached, agents will have no incentive to deviate from agreed negotiation strategies.

- **Symmetry:** the negotiation process should not treat agents differently. Individual agents may need and wish to discriminate but the interaction process itself should not be unfairly weighted against certain agents.

These attributes need not be universally accepted. There will sometimes be trade-offs between one attribute and another. Ultimately, there are some criteria that determine the acceptability of one interaction protocol over another. In most cases, distribution, stability and efficiency are regarded as the key points which a system should satisfy, whilst others, like symmetry, may not be reached if the real application situation requires one of the agents to play a more important role than the others.
8.4.2 Analysis

Of these attributes, the distribution characteristic is the basis of the MASCOT model. Also, the symmetry characteristic of the MASCOT model has been addressed in the system's assumptions. This section will focus on the three major attributes: efficiency, stability and simplicity. The suitability of the system would be evaluated through the prototype evaluation.

- Efficiency

Ideally, agents should not squander resources when they come to an agreement. There should not be wasted utility. For example, it makes sense for the agreements to satisfy the requirement of Pareto-optimality, where no agent could derive more from a different agreement without some other agents deriving less. Given the full information assumption and the Monotonic Concession Protocol, Zeuthen's strategy is efficient. Agents using the Zeuthen strategy will not run into conflict because at least one of them will always make a concession. Since agreement is guaranteed, and offers are restricted to the negotiation set, the result will be efficient (Rosenschein and Zlotkin, 1994). Harsanyi (1977) have confirmed that if both agents use Zeuthen's strategy, they will agree a deal that maximises the product of their utilities (i.e. Pareto-optimal).

In MASCOT, the application of Zeuthen's strategy is based on the agents' beliefs, and the final settlement point is Bayesian-Nash equilibrium. Therefore, the agreement may not be Pareto-optimal. However, the result is sufficiently efficient (i.e. the agreement is Pareto-optimal if the agents' beliefs are true - this is reasonably true at the final negotiation stage in the MASCOT model).

- Stability

No agent should have an incentive to deviate from the agreed-upon strategies. This is known as the notion of strategies in equilibrium. The strategy that agents adopt can be proposed as part of the interaction environment design. Once these strategies have been
proposed, the individual agent designers (e.g. different industry parties) should not have an incentive to go back and build their agents with different, manipulative, strategies.

Three levels of equilibrium are commonly used in economic (game) theory, which are Nash equilibrium, perfect equilibrium and dominant equilibrium (Binmore, 1990; Rasmusen, 1994). Each level of equilibrium is stronger than the preceding one. As defined above, two strategies $S, T$ are in Nash equilibrium if, assuming that one agent is using $S$, the other agent cannot do better using some strategy other than $T$, and vice versa. Perfect equilibrium means that when the game has multiple steps, and one player is using $S$, there exists no state in the game where the other player can do better by not sticking to the strategy $T$. There do exist situations where strategies might be in Nash equilibrium, but not in perfect equilibrium. In that case, although strategy $T$ was best at the start of the game, as the game unfolds it would be better to diverge from $T$. Dominant strategy equilibrium means that no matter what strategy your opponent chooses, you cannot do better than to play strategy $T$; strategies $S$ and $T$ are in dominant strategy equilibrium when $S$ is the dominant strategy for one player and $T$ is the dominant strategy for the other. However, the above equilibria are based on the complete information assumption, which is not true in most practical cases. To overcome this problem, Bayesian-Nash equilibrium is introduced by Harsanyi (1965, 1977). This equilibrium includes a set of beliefs (one for each agent) and a set of strategies. A strategy combination and a set of beliefs form a Bayesian-Nash equilibrium if the strategies are in Nash equilibrium given the set of beliefs, and the agents update their beliefs, according to Baye’s rule.

The use of Nash equilibrium may not be an effective way of analysing the outcomes of the models of alternating offers (Kraus, 1995). Some games may not have any equilibrium strategy (Sandholm, 1996), while some others may be in equilibrium only at the beginning of the negotiation, but may be unstable at intermediate stages. Rosenschein and Zlotkin (1994) have proved that Zeuthen’s strategy is not stable in the case when two agents find themselves one step away from concluding a negotiation, and have equivalent risk. One agent, knowing that the opponent is using Zeuthen’s strategy, could diverge.
from Zeuthen's strategy by not making a concession, and benefit from this divergence. To avoid such problems, this study formalises the assumptions that are appropriate for the claims negotiation environment. For example, all agents suffer a loss over time, there is a finite set of agreements, and any agreement reached is better for all agents than opting out of the negotiations. When an agent tries to diverge from Zeuthen's strategy, the time penalty will push the agent with high time penalty to concede at the next encounter. Therefore, the negotiation will finally converge at an equilibrium point (i.e. Bayesian-Nash equilibrium).

- Simplicity

By adopting the MASCOT negotiation protocol and strategies, agents need to compute the entire negotiation set in order to carry out their strategy. This may require a great deal of information about the encounter. The learning process also increases the calculation expense. However, such calculation is performed locally by each agent, thus, it is less costly compared with communication. Communication is typically expensive in terms of time and resources and can become a bottleneck of the negotiation process. The learning mechanism allows an agent to get more realistic beliefs about its opponent by updating its beliefs during negotiation, thus, reducing the frequency of communication.

8.5 EVALUATION OF THE MASCOT PROTOTYPE

8.5.1 Evaluation Process

Following the theoretical evaluation of the MASCOT negotiation mechanism, a demonstration assessment of the MASCOT prototype was carried out in line with the objectives outlined in section 8.2. This was carried out by presenting the prototype to a number of evaluators, and obtaining their feedback via a questionnaire and discussions. All those who were invited to participate had practical experience in the industry as well as experience in the use of information technology. In all, the opinions of nine people were obtained.
In order to ensure that the respondents fully understood the concepts involved, they were first shown a presentation. It first introduced the essential characteristics and features of MAS with focus on the negotiation algorithms between agents; then discussed the theoretical foundations of the MASCOT model; finally, it explained the development process of the model, especially the nature and key characteristics of construction claims negotiation. The presentation also addressed the broad objectives of the application of MAS in the industry. The slides of this presentation are shown in Appendix B.

The demonstration of the MASCOT prototype immediately followed the presentation. The demonstration illustrated a typical negotiation process between the contractor agent and the engineer agent regarding the claim for loss of productivity. Three cases were illustrated (i.e. an agreement was reached by the contractor agent and the engineer agent; an agreement was reached by the involvement of the client agent; and no agreement was reached). The participants were invited to interject at any point during the demonstration to ask questions of clarification.

Following the demonstration, the questionnaire was distributed. Key questions were further explained. Discussions were held later regarding the details of the model and the further development of MAS in the industry.

8.5.2 Analysis of Results

Using the feedback from the questionnaire, an analysis on various aspects of the MASCOT model was conducted:

8.5.2.1 Evaluators' Background

From the responses to the five questions regarding the respondents' industry backgrounds, it can be seen that respondents held a wide variety of roles in the industry, as shown in Table 7.25. Some participants often had experience representing more than one background.
Table 8.1 The Evaluators' Background

<table>
<thead>
<tr>
<th>Contractor</th>
<th>Engineer</th>
<th>Architect</th>
<th>Client</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

Their working experience in the industry ranged from 1 to 37 years with an average of 10 years. 90% of the respondents had more than three years working experience. The descriptions of positions held in their organisations produced a wide variety of answers including quantity surveyor, project manager, civil engineer, research engineer, and project engineer (representing the client). 78% were familiar with construction claims and claims negotiation while 50% had experience of applying IT in construction management.

Based on the above information about the respondents' level of experience, roles played, positions held, knowledge of claims negotiation, and IT applications in the industry, it was considered that the evaluation group, sufficiently qualified to provide a fair assessment of the system.

8.5.2.2 Responses to Questions

For each section of the questionnaire, the results of the quantitative questions are shown in the followings tables, where a response of 1 represents 'strongly disagree' or 'poor', whilst 5 represents 'strongly agree' or 'excellent'. The results are also discussed based on the mean ranking of each question. This was followed by a selection of the most pertinent of the responses to the qualitative question. The achievements against objectives were assessed at each stage.

1) Responses to Construction Claims Negotiation

This section contained the questions about essential features of construction claims negotiation on which the MASCOT model was developed and could be further developed (Table 8.2).
### Table 8.2 Responses to construction claims negotiation

<table>
<thead>
<tr>
<th>Statements</th>
<th>Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Inefficiency is one of the major problems in current construction claims negotiation.</td>
<td>2 5 2 4.00</td>
</tr>
<tr>
<td>1.2 A negotiator can have a certain perception about the opponent's fraud offer (exaggerated amount) after s/he negotiates with the opponent for some time in a project.</td>
<td>5 2 4.29</td>
</tr>
<tr>
<td>1.3 Negotiation tactics play an important role in deciding the outcome of the claim negotiations</td>
<td>2 6 1 3.89</td>
</tr>
<tr>
<td>1.4 The involvement of the client in the claims negotiation is helpful for the negotiation</td>
<td>2 4 3 4.11</td>
</tr>
<tr>
<td>1.5 It is possible to adopt a computer-aided approach to improve the efficiency and resolve some problems of construction claims negotiation</td>
<td>1 2 4 2 3.78</td>
</tr>
<tr>
<td>1.6 Two factors influence a negotiators' attitude in claims negotiation:</td>
<td>3</td>
</tr>
<tr>
<td>a) Try to maximise his/her own benefits;</td>
<td>2</td>
</tr>
<tr>
<td>b) Try to avoid breaking the negotiation;</td>
<td>2</td>
</tr>
<tr>
<td>Within these two factors,</td>
<td></td>
</tr>
<tr>
<td>• a) is more important than b)</td>
<td>3</td>
</tr>
<tr>
<td>• b) is more important than a)</td>
<td>2</td>
</tr>
<tr>
<td>• a) is as important as b)</td>
<td>2</td>
</tr>
<tr>
<td>• It depends on the claim cases</td>
<td>2</td>
</tr>
</tbody>
</table>
1.7 Two factors may influence a negotiator's decision on the concession amount:
   a) The gap between his/her current offer and his/her reservation value;
   b) His/her estimate of the opponent's reservation value and the opponent's current offer;

Within these two factors:

- a) is more important than b) 4
- b) is more important than a) 3
- a) is as important as b) 0
- It depends on the claim case 2

The respondents gave a relatively high ranking to the first four statements. Also, the range of responses was fairly low for these statements. This confirmed some important features of the MASCOT system, particularly the rationale for the development of the system, input information, and negotiation strategy. Furthermore, a few points need to be discussed:

- Firstly, the responses to statement 1.5 received the lowest mark (3.78) in this section. The respondents' attitude to the adoption of computer supported negotiation was relatively neutral rather than positive. As a key pre-condition of the system, the responses were further analysed. The five respondents who claimed to have experience of applying IT in construction gave a high rank to the statement (two strongly agreed; while three ranked 'agreed'). On the other hand, those who had little knowledge of IT gave it a relatively low rank (one ‘agreed’; two were neutral; and one ‘disagreed’). This reflects the respondents' general confidence in the application of IT in the industry.

- Secondly, the almost evenly distributed rank to the two motivations of claims negotiation (i.e. to maximise each party's own benefits, or to avoid breaking the negotiation) in question 1.6 (Q1.6) indicates that the respondents agreed that both aspects were essential for construction claims negotiation. In other words, it
confirmed the contract-obliged self-interested reasoning model of the MASCOT system.

- Thirdly, the responses to Q1.7 indicated that both claims negotiators’ own negotiation features and their estimation of their opponent’s negotiation features were important for decision-making during negotiation. This suggested an interesting point: that the dual responsive model – (a behaviour theory-based negotiation model) would be appropriate for the further development of the MASCOT system. This will be discussed further in Chapter 9.

The results of this section suggested that the respondents agreed in general that the essential features of construction claims negotiation on which the MASCOT model based were defined appropriately in this research project.

2) Responses to Multi-agent Systems

Compared with the responses to Q1.5, this result was much more positive (Table 8.3). The respondents had relative high confidence about the solutions provided by multi-agent systems in construction claims negotiation. This supported the major hypothesis of this research project.

Table 8.3 Responses to multi-agent systems

<table>
<thead>
<tr>
<th>Statement</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Mean Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Compared with other information technologies, such as expert systems and decision making systems, multi-agent systems are more suitable to solve the problems caused by the fragmentation of the construction industry</td>
<td></td>
<td>2</td>
<td>5</td>
<td>2</td>
<td></td>
<td>4.00</td>
</tr>
</tbody>
</table>

3) Responses to the MASCOT Model

Various aspects of the MASCOT mode were evaluated in this section (Table 8.4).
Table 8.4 Responses to the MASCOT model

<table>
<thead>
<tr>
<th>Statement</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Mean Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 How well does the MASCOT model ensure that all the essential perspectives of construction claims negotiation are represented?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>4.11</td>
</tr>
<tr>
<td>3.2 There are seven pieces of input information in the MASCOT model:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the negotiated claim item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the negotiator’s reservation value for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the negotiator’s optimum value for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the negotiation deadline for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the opponent’s reservation value for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the opponent’s negotiation habit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- the confidence about this input information</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>To what extent does this input information represent the information needed for real negotiations?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>4.00</td>
</tr>
<tr>
<td>3.3 How effective is the modified Monotonic concession protocol in the MASCOT model?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.89</td>
</tr>
<tr>
<td>3.4 How effective are the listed MASCOT negotiation strategies?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a) the adoption of the maximum risk acceptability as the criterion for concession</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>4.22</td>
</tr>
<tr>
<td>b) the learning ability of agents in the claims negotiation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td>4.38</td>
</tr>
<tr>
<td>c) the time penalty in the claims negotiation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>3.78</td>
</tr>
<tr>
<td>d) the involvement of the client agent to facilitate claims negotiation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>4.00</td>
</tr>
<tr>
<td>e) the extended solution-searching to facilitate claims negotiation</td>
<td></td>
<td>7</td>
<td>2</td>
<td>4.22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5 To what extent does the MASCOT system improve the efficiency of claims negotiation?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6</td>
<td>4.33</td>
</tr>
<tr>
<td>3.6 To what extent does the MASCOT system improve the effectiveness of claims negotiation?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>3.67</td>
</tr>
</tbody>
</table>
The respondents gave relatively high rankings to Q3.1, Q3.2, Q3.3, Q3.4a), b), d), e) and Q3.5. The results suggested that the respondents felt in general that the principal features of system were affective. In other words, they agreed that:

- the MASCOT system represented all the essential characteristics of construction claims negotiation;
- the input information of the MASCOT prototype represented the information needed for real negotiations;
- the modified monotonic concession protocol was relatively effective for the MASCOT model;
- the adoption of the maximum risk acceptability as the criterion for concession was appropriate;
- the learning ability was important for agents in claims negotiation;
- the involvement of the client agent could facilitate claims negotiation;
- the extended solution-searching strategy was appropriate for the MASCOT model; and
- the MASCOT system could improve the efficiency of claims negotiation.

However, the respondents gave relative low scores to two questions regarding agents' time penalty during negotiation process (Q3.4c) and the improvement of claims negotiation effectiveness through the MASCOT system (Q3.6). These two questions were also discussed after completing the questionnaire.

A major question that respondents asked regarding the Q3.4c was "since computer negotiation could be very efficient, the time penalty would have little function in practice". In this aspect, the respondents failed to recognise that an agent could just hold its position for many iterations if the agent was not given a time penalty for such time-wasting strategy. In fact, the adoption of time penalty as a functional attribute has been discussed by many game theorists, economists and MAS researchers. In the MASCOT model, an important role played by the time penalty was to keep Zeuthen's strategy stable.
A more important discussion was about Q3.6. Although IT is commonly recognised as a powerful tool in improving system efficiency, it could also improve system effectiveness provided that the IT system is properly designed and adopted in an appropriate circumstance. An example is the application of the word processor and spreadsheet that not only improves the speed of writing and calculation but also reduces mistakes during these processes. Regarding the MASCOT system, a typical case that improves the effectiveness of claims negotiation is when the system reduces the problems caused by the engineer's conflicting roles in claims management. Nevertheless, the evaluators felt that the MASCOT system does not necessarily improve the overall effectiveness of construction claims negotiation.

Besides ranking the statements, two closely related questions were asked in Q3.2 regarding the input information. These responses covered a wide area which should or could be considered at the next stage, such as:

- Please specify the problems if you are not satisfied with the input information:
  
  a) how elegant the guessing skills involved are;
  
  b) how to reflect the opponent’s negotiation tactics; and
  
  c) how to reflect the history of claims better.

- Which kind of information needs to be added or deleted?
  
  a) the honesty of the opponent;
  
  b) the correlation between past history and current claims; and
  
  c) the possibility for the users to modify the utility functions.

Most of the comments were related to human or other complex issues of construction claims negotiation (This point will be further addressed in the next section). For example, a major concern was how to encode human negotiators' domain knowledge into agents' domain knowledge. More industry surveys are needed to address such issues.
4) Responses to General Questions

This Section contained some of general questions regarding the usage and further development of the MASCOT model, and the potential application of MAS in other construction areas (Table 8.5, 8.6, 8.7).

Table 8.5 Responses to general questions (1)

<table>
<thead>
<tr>
<th>Statement</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Mean Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 This study has clearly analysed the characteristics of construction claims negotiation, and identified the problems properly.</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td></td>
<td></td>
<td>4.22</td>
</tr>
<tr>
<td>4.2 How useful is the MASCOT system to different participants?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Contractor</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td>4.33</td>
</tr>
<tr>
<td>• Engineer</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td>4.11</td>
</tr>
<tr>
<td>• Client</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td></td>
<td></td>
<td>4.22</td>
</tr>
<tr>
<td>• Others: sub-contractors, suppliers</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>3.56</td>
</tr>
<tr>
<td>4.3 How easy is the MASCOT prototype to use?</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td></td>
<td></td>
<td>3.78</td>
</tr>
<tr>
<td>4.4 How useful is the MASCOT system to the overall claims management or project management?</td>
<td>5</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>4.44</td>
</tr>
<tr>
<td>4.5 What is your overall rating of the system?</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td>4.33</td>
</tr>
</tbody>
</table>

The questions in this section were in two parts. Q4.1 to Q4.5 summarise the respondents' overall evaluations of the MASCOT system. The first statement received a high response of 4.22 indicating that the MASCOT model was built based on a clear analysis of claims negotiation characteristics and problems. Furthermore, the respondents confirmed that the system would benefit all the major claims negotiation participants by ranking them almost equally high. However, a low mark (3.56) was given to the subcontractors and suppliers since the demonstration did not include these project participants. In fact, further explanations needed to be made to the evaluators.
Importantly, the respondents gave very high scores to the overall system (Q4.5), and its usability in claims management and project management (Q4.4). This means that the essential goal of the system has been achieved. However, the respondents gave a lower rank (3.78) to the question ‘how easy is the prototype to use?’ (Q4.3). The main reason addressed by the respondents was due to the terms used to describe the input information and the figure used to check the input information. During the discussion, it was also recognised that there should be a balance between the complexity of the input information and the power of the system (i.e. the comments for Q3.2 and Q4.3).

The second part (Q4.6-Q4.8) is concerned with improvements to the MASCOT system. The following questions were answered:

- Q4.6 What is the risk of application of the MASCOT system?

The responses given to this question were:

  a) The reliability of the system (including both software and hardware), such as bugs or virus;
  
  b) The honesty and reliability of the other parties. A party could modify his agent and benefit from it;
  
  c) People may depend too much on the system; and
  
  d) No one would be responsible for mistakes if any problem happened. Problems would be attributed to machine failure as seen in the case where machines are used to take actions on a human’s behalf.

The first two responses revealed the security problems for the application of the system. The problems of network security issues related to agents could be created by external agents (e.g. viruses, pilfering databases, or program modifications by the opponent) or by internal agents (e.g. unauthorised data modification, or failure of actions). All these problems could lead to serious results. Claims negotiation involves many sensitive issues, particularly because it aims to reach a deal for a claim. Therefore, industry users may be
reluctant to deploy the system without adequate guaranteed security. The last two responses were concerned with how the MASCOT system could be applied appropriately in the industry, which is highly related to the human factors. This will be further discussed later.

- Q4.7 Suppose you are using the MASCOT system, to what range will you allow your agent to determine the final negotiation amount?

Two kinds of responses were given:

a) The first was directly related to the maximum value of agent negotiation. The respondents' answers ranged from £100-500, less than £1000, less than £2000, to maximum claim amount given that the accuracy was within 5-10%.

b) The second kind of responses suggested that agents should submit the agreement reached by them to their owners as a suggestion value, or as a possible negotiation starting point. Any agreement should be confirmed by their owners.

There are potential legal implications in abdicating responsibility for critical decisions to agents that have independent execution autonomy, especially when this happens in claims negotiation. Thus, agents' authorities become very important and more sensitive in the MASCOT system. The respondents addressed this question from two different perspectives. The first comment suggested the maximum amount authorised by agent. Although the value varied widely, this approach was also suggested by the other industry partners of this project. The second comment suggested that all the deals should be accepted by the owners in order to achieve the twin objectives of giving negotiators ultimate control in the final decisions, whilst facilitating the negotiation process. In practice, this issue would be determined by the complexity of claims, significance of the system, and users' confidence in the system. All these aspects are closely related. An agreement should be reached regarding agents' authorities before the system is adopted in any project.
Q4.8 What might discourage people from using the system?

The comments made by the respondents included:

a) new technologies, thus, difficult to accept;
b) possibly difficult to use for the people who lack IT knowledge;
c) loss of benefits from human interaction during negotiation;
d) project participants' ignorance of the system;
e) user interruptions to the system;
f) lack of transparency; and
g) lack of trust of machines in settling financial matters.

There was a wide spectrum of response to this question with the respondents giving a number of issues and potential barriers that might discourage people from using the system. By far, the most popular opinion was the lack of exposure to IT that currently exists in the industry. This problem is more serious for the adoption of MAS than other IT tools because MAS is based on AI. The failure of AI to deliver on some of its previous promises of the 1980s has contributed to users' reluctance in accepting any new AI technology. This is principally due to the overselling of AI systems (Knowledge based and expert systems), with regard to their capabilities to mimic human intelligence in solving complex industrial problems. This experience at industry level has resulted in most AI research projects being confined to academic research labs. Furthermore, development of the MASCOT system should take this on board and ensure that the specific concerns and needs of industry are addressed.

Another major barrier to the adoption of the system is the lack of trust of machines in settling financial matters. In many projects, claims management is thought too complex and important to be handled by computers alone. The reliability and benefits of the system to a project must be clear and demonstrated to encourage its use. Closely related to this problem is another common theme (i.e. the cultural issues such as the feeling of
loss of control). Enough flexibility for the user would need to be built in to allow them to remain in effective control.

Further evaluation work in the form of a trial on an actual construction project may be the best way to determine both the problems involved and the potential benefits to both the users and the project in general. This is considered as future work.

Table 8.6 Responses to general questions (2)

<table>
<thead>
<tr>
<th>4.9 In what ways can the MASCOT system be further improved?</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Identify more clearly the characteristics of construction claims negotiation, such as:</td>
</tr>
<tr>
<td>• Build a more sophisticated negotiation mechanism by considering more human aspects and claim elements</td>
</tr>
<tr>
<td>• Others, please specify:</td>
</tr>
</tbody>
</table>

Within the three potential areas, most respondents suggested improving the system through building a more sophisticated negotiation mechanism by considering more human elements and claim factors. A common comment regarded the involvement of qualitative information in claims negotiation. Others were concerned the factors such as: honesty of negotiators, negotiation tactics, and the information transfer from other project management systems. This is consistent with the comments in Q3.2. Although it would be desirable to involve more human and claims features in the system, such work should be sound and systematic. This will be the major concern of future work (see Chapter 9).

One respondent chose the option that the system could be further improved by focusing on the clear analysis of the characteristics of construction claims negotiation. An example given was to add the overall claims level as an attribute to the system. Also, two respondents suggested that the system could be improved by working on other aspects, such as: the better integration of human negotiators and agents, adopting more
sophisticated implementation systems, and improving the security and safety of the system.

Table 8.7 Responses to general questions (3)

<table>
<thead>
<tr>
<th>Q</th>
<th>Question</th>
<th>1</th>
<th>6</th>
<th>2</th>
<th>4.11</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.10</td>
<td>Through this study, do you think multi-agent systems have a high potential of resolving distributed construction problems?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.11</td>
<td>In which areas can the Multi-agent systems be best used in the construction industry?</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

|   | a) engineering design         | 2 | 5 | 2 | 4.00 |
|   | b) concurrent engineering     | 5 | 3 | 1 | 3.56 |
|   | c) material management        | 2 | 4 | 3 | 4.11 |
|   | d) scheduling and control     | 3 | 4 | 1 | 3.75 |
|   | e) supply chain management    | 3 | 3 |   | 3.50 |
|   | f) others, such as:           |   |   |   |      |

Q4.10 received a relatively high mark (4.11) indicating that the respondents have recognised the potential benefits brought by the application of MAS in the industry. The responses to the five potential areas listed in Q4.1 further confirmed the respondents' positive attitude towards this point. Within these areas, the possible application of MAS in facilitating project material management has received the highest mark. In rank order, the others were engineering design, scheduling and control, concurrent engineering and supply chain management. The reasons for such ranking were twofold: first, it was thought that material management was less related to complex human factors, thus, more suitable for the application of MAS. Second, limited by time, some background knowledge was not fully introduced to the respondents during the evaluation process. The lowest and the second lowest ranking received by supply chain management and concurrent engineering were probably due to the respondents' limited background knowledge because some researchers (e.g. Swaminathan et al., 1998; Chen et al. 1999; Chedmail, 2001) have applied MAS in supply chain and concurrent engineering environments, and proved that MAS is an effective tool. Besides the listed areas, the
respondents also suggested a few other industry processes which could be improved using MAS, such as: general project management, and co-ordination between main contractor and sub-contractors.

8.5.3 Discussion

The following sections distil the results of the evaluation results and further discuss the effectiveness of the evaluation methods.

8.5.3.1 Results

Firstly, the theoretical analysis has proved that the MASCOT negotiation mechanism satisfies all the key evaluation criteria (i.e. the MASCOT system is reasonably simple, stable and efficient).

Secondly, the responses to the MASCOT system, and the applicability of multi-agent systems in construction provide a positive assessment for each of the objectives laid out in Section 7.2. As a whole, the potential benefits of the prototype based upon the evaluation can be distilled as:

- The MASCOT model clearly addresses the major characteristics and problems of construction claims negotiation;
- The major features of the model such as the system architecture and negotiation mechanism match the requirements of construction claims negotiation. The system possesses some particular advantages which can overcome problems which are not easily addressed by other approaches;
- The methods used to develop the system are both appropriate and suitably well implemented;
- MAS has great potential in facilitating construction engineering and management activities.
Despite this positive feedback obtained from the evaluators, there are some points that need to be further explained:

- Firstly, the lack of IT knowledge, the low usage of IT tools and the fear of using IT tools to handle financial issues in the construction industry are potential barriers for the adoption of the MASCOT system. Moreover, the difficulty in quantifying the immediate return, especially the economic gains, on the system investment may make it more difficult to be accepted.

- Secondly, although the MASCOT system can facilitate claims negotiation, the system could operate more effectively by adding further features. The maximum benefit can be gained through the integration of the system with other legacy systems, or more importantly, by considering more practical human factors and qualitative claims negotiation elements.

- Thirdly, unlike other industries, the construction industry is less willing to adopt advanced technologies due to the specific characteristics of the industry. Moreover, claims management and claims negotiation are one of the most complex and difficult areas in construction management. Although this research project adopted MAS in this area, it would be much easier to adopt MAS to resolve other engineering or management problems, such as material management, engineering design or concurrent engineering.

- Finally, it has been pointed out that the system alone will not necessarily improve the overall construction claims management. It needs to be an integral part of a claims management strategy.

8.5.3.2 Effectiveness of the Evaluation Approach

The reasoning behind the selection of the demonstration and questionnaire evaluation method has been discussed in Section 8.3. This is manifested by the evaluation process and the results achieved. The chosen methodology helped test all aspects of the system as required in the evaluation objectives.
The evaluation process has resulted in a body of quantitative and qualitative results. The quantitative results are in general positive and give a good indication that the MASCOT prototype is considered to be appropriate for construction claims negotiation. The qualitative questions promoted the respondents to highlight areas that could be improved and to suggest what those improvements might be. This naturally leads to a more negative set of responses. In sections where lower marks were given for the quantitative questions there were fuller responses to the qualitative questions. Moreover, the respondents were encouraged to raise their queries during the evaluation process. Most of the queries related to those questions that received low marks in the questionnaire. This resulted in a good set of potential issues and concerns being given by the respondents. In addition, a wide variety of suggestions for improvements were also provided.

The assessment process has been successful in that it has provided an indication of how appropriate the prototype is as well as important issues that need to be addressed to improve the system. However, it should be recognised that there are limitations in the results gained through the demonstration assessment. A major drawback was that the respondents' information about the system heavily depended on the presentation and the demonstration of the prototype. Due to time constraints, some aspects of the system could not be fully introduced to the respondents. As a result, those points received relatively low marks.

8.6 SUMMARY

This chapter has presented a two-step system evaluation process. Firstly, a theoretical evaluation was conducted to analyse the quality of the MASCOT negotiation mechanism. It was proved that the MASCOT negotiation mechanism satisfied all the key criteria of the MAS negotiation mechanism defined by previous MAS researchers (i.e. the system is simple, stable and efficient). Secondly, a prototype evaluation session was carried out to assess the MASCOT prototype and perspectives for the applications of MAS in the industry. The results were highly positive and indicate that the objectives of the MASCOT system have been achieved. Moreover, the majority of respondents thought
that MAS would be a powerful technique in solving problems in the industry. The respondents also pointed out some important areas that were essential for the further development and applications of the MASCOT prototype in the construction industry. The next chapter will summarise the work conducted in this research, draw conclusions and make recommendations for further development.
CHAPTER 9
CONCLUSIONS AND FURTHER DEVELOPMENTS

9.1 INTRODUCTION

This chapter concludes this research project, which explored the application of MAS to construction claims negotiation. It first summarises the model development process, the implementation, and the evaluation of the resulting prototype. It then concludes that MAS could be an effective and powerful tool in supporting construction engineering and management activities and resolving the problems associated with these activities. It ends by making recommendations for future work.

9.2 SUMMARY

The rationale of this research is that the particular characteristics of MAS are ideally suited to distributed collaboration. The applications of MAS in other industries, such as military, telecom and manufacture also suggested that MAS could be used to tackle the fragmentation problem that the construction industry faces. The aim of this research was to explore the applicability of MAS to facilitating the construction claims negotiation process, and resolving the major problems associated with the negotiation process. To achieve this aim, a number of research objectives were identified with respect to construction claims negotiation, MAS, negotiation theory, model development and implementation. Various activities were undertaken to achieve these objectives, which included: extensive literature reviews, interviews, group discussions, participation in conferences, prototyping, and system evaluations.
• Construction Claims Negotiation

Initially, an extensive review provided general background on construction claims management. It shows that, although many efforts have been made at both the pre-construction and construction phases, the number and amount of construction claims and disputes are still escalating. The reasons are complex and multi-dimensional. The inefficient and ineffective claims negotiation is one of the major contributors.

By investigating industry practice, exploring the negotiation environment and process, building and analysing the generic claims negotiation model, the major problems of claims negotiation were identified during the negotiation preparation and negotiation processes. These problems include distributed project teams (either geographical or organisational), inefficient communication, poor documentation, negative attitude to claims, lack of negotiation knowledge and expertise, lack of effective supporting tools, the conflicts between the engineer's authority and responsibility, and various problems inherent in negotiations as conflict resolution methods.

Although various techniques have been adopted to improve the efficiency and effectiveness of negotiations, such as decision-making systems and communication techniques, their effects are limited because they can only support a single party's decision-making or only facilitate communications between negotiation parties. A newly developed computer technology - MAS has great potential to overcome the above problems.

• Multi-Agent Systems

This leads to the second research objective, which was to investigate the nature, principles and major characterises of MAS. Particular attention was paid to the development of negotiation mechanisms between agents (i.e. negotiation protocol and strategies), and various agent learning approaches. The major features of agents, such as autonomy, social ability, reactivity and pro-activeness make them ideal tools for resolving collaborative problems. As a flexible and powerful tool, MAS could be
designed to work in different environments to achieve their particular goals by adopting different co-operation mechanisms. Agents could be group-benefit oriented or self-interested; the latter was the focus of this research project.

The key points are that system developers should, based on the real application environment, develop appropriate system architectures and, more importantly, negotiation protocol and strategies. As a result, various negotiation mechanisms have been developed, such as contract-based negotiation, plan-based negotiation, market-based negotiation, game theory-based negotiation, AI-based negotiation, psychology-based negotiation and other approaches. These approaches have been proved to be successful in resolving specific problems. Also, a number of learning approaches have been developed to allowing agents to obtain more knowledge about their working environment or key features of their partners during negotiations, such as inductive learning, analytic learning, genetic algorithms, and connectionist learning methods.

However, since MAS technique is quite new, many important issues need to be further addressed in developing and applying MAS, such as system development methodology, agent communication languages, resource allocations, interaction mechanisms, agents' reasoning model, and implementation tools. Many MAS were developed in an ad hoc manner. There is not a generic approach to developing MAS, particularly agent negotiation algorithms.

- Negotiation Theories

Consequently, the third research objective was to explore negotiation theories in order to identify the theoretical foundation for the MASCOT negotiation mechanism. Three approaches based on negotiation theories were discussed; these included game theory, economic theory and behaviour theory. The first two approaches are closely related. Both focus on providing a device to clarify and isolate key factors in negotiation, to address the obstacles that stand in the way of solutions, and explore what special cases of the problem can be treated by what methods with this facility. For example, they both share
the same important assumptions, and adopt the concept of utility functions. The
difference between these approaches are that game theory focuses on anticipating the
negotiation results based on the rational assumptions, whilst the economic approach
focuses on analysing the negotiation process. Game theory approach represents a static
approach to negotiation analysis, whilst economic theory approach represents a dynamic
approach. Some important economic models provide a sound theoretical foundation for
the further development of MAS negotiation mechanism such as: Zeuthen’s model, Cross
model, and Osborne and Rubinstein model; they address the time issues and/or learning
issue during the negotiation.

Unlike these two approaches, behaviour theoretical models focus on the human aspects of
negotiation. They attempt to analyse the negotiation processes where negotiators
influence each other’s expectations, perceptions, assessments, and decisions during the
search for an outcome, thereby affecting the outcome. Much attention is given to the
nature of changing expectations and negotiator’s tactics, and to the significance of
uncertainties of information, perception, and evaluation - all matters that tend to be
ignored by the game theory approach and the economic theory approach. All this
involves a closer approximation to the real world. The psychological model, learning
model, dual-response model and joint decision-making model are all important behaviour
theory models focusing on different aspects of human behaviour in negotiations. These
aspects are discussed by construction claims negotiation researchers. Thus, it is essential
to consider these human issues in the MAS negotiation mechanism.

- Development of MASCOT Conceptual Model

The fourth research objective was to develop a MAS model for construction claims
negotiation (MASCOT) based on the studies of construction claims negotiation, MAS,
and negotiation theories. This process included five steps:

a) Firstly, the essential characteristics of construction claims negotiation were further
analysed in terms of the application of MAS, where the contractually-obliged self-
interested negotiation was identified as the key nature of construction claims
negotiation, and the reasoning model of the MASCOT system. Furthermore, some other characteristics of claims negotiation were also identified such as participant-dependent information, strategy-influenced process, time consumed for negotiation, the involvement of the client, and tradeoffs between different claim items. These factors play important roles in the development of the MASCOT system.

b) Secondly, the negotiation theory model — (Zeuthen’s model) was selected as the theoretical model for the MASCOT system because it reflected the contractually obliged self-interested nature of the claims negotiation. Importantly, the result reached by this model also maximises each party’s utility which the Nash solution (1950) addresses. Furthermore, a learning approach - the Bayesian learning approach was also adopted to keep Zeuthen’s model stable in an incomplete information environment.

c) Thirdly, the MASCOT negotiation mechanism was developed. This included the development of the modified Monotonic Concession Protocol (mMCP) and various negotiation strategies which reflected the major characteristics of claims negotiation. An essential negotiation strategy was the integration of Zeuthen’s strategy and the Bayesian learning approach. Also, other strategies, such as the involvement of the client agent and the expanded solution-searching strategy, were developed.

d) Fourthly, the MASCOT processing model was developed and illustrated using the IDEF0 diagram, where a whole claims negotiation process was described.

• Model Implementation

The fifth research objective concerned the implementation of the MASCOT model. This included two major parts. Firstly, the implement environment - the ZEUS agent building toolkit - was selected and studied. The ZEUS toolkit architecture was particularly analysed; this included the agent component library, the generic agent, the agent building software, the utility agents, and the integrating agents with external programs. Secondly,
a full description of the MASCOT model implementation process was given. It included a three-step process.

Initially, MASCOT role modelling was conducted where the roles and domains of the claims negotiation were identified. Following this process, the implementation design was undertaken where agents were created; ontology was defined; negotiation protocol and strategies were further clarified to suit the ZEUS environment; more importantly, the input information was determined, and the probability distributions representing the input information were selected. Finally, the model was implemented to address issues such as: negotiation protocol and strategies were addressed through the integration of the Rulebases, Primitive tasks and external Java programme. Furthermore, the utility agents and task agents were set up in the system. The display windows were also presented.

- System Operation

The sixth objective was to execute the developed MASCOT system. A practical example was created to examine and validate the model. All the major features of the model were presented during the operation process; which include: utility function definition, negotiation preparation, and detailed negotiation process. Results of the system execution were discussed. Comparisons between the MASCOT negotiation mechanism and other negotiation mechanisms such as simple gradient descent strategy were also made. The example was also run using the developed prototype.

- Evaluation

The final research objective was to evaluate the developed system using appropriate techniques. It included two steps: a theoretical assessment of the MASCOT negotiation mechanism, and a prototype evaluation by industry practitioners and academic researchers. A detailed discussion about the selection of the evaluation technique, the selection of the evaluators, the design of the questionnaire, the feedback mechanism, and
The analysis of the feedback result were described in Chapter Seven. The results of the evaluation in a nutshell showed that:

a) The MASCOT negotiation mechanism satisfied all the major evaluation criteria such as stability, efficiency and distribution. It meant that the design of the MASCOT negotiation mechanism was successful.

b) The respondents gave a positive response to the system, its further development, and most importantly, to the future application of MAS in the industry.

The evaluation confirmed that the MASCOT system could be an effective tool for facilitating construction claims negotiation. Furthermore, MAS, with their great potential for supporting distributed collaboration, could be an effective tool in supporting construction engineering and management activities, such as collaborative design, concurrent engineering, and supply chain management.

9.3 CONCLUSIONS

This thesis has demonstrated the applicability and role of a MAS in construction claims negotiation, and developed a robust negotiation model for agents. Three main conclusions can be drawn from this research:

- The MASCOT prototype system presented in this thesis represents an innovative and effective approach to supporting construction claims negotiation

Based on a thorough analysis of industry practice, framework, essential characteristics of claims negotiation, multi-agent systems, and negotiation theories, the MASCOT architecture was built; the negotiation mechanism was developed. As shown by the evaluation, the system not only facilitates the claims negotiation by breaking the time and location barriers between the distributed project participants, but also resolves some particular problems associated with construction claims negotiation (e.g. the unbalanced responsibility and role for the engineer during the claims management) by involving the
intelligent agents. Hence, the MASCOT system has great potential to improve both the effectiveness and efficiency of construction claims negotiation. A number of particular conclusions can be drawn:

a) The adoption of Zeuthen's negotiation model is proven as appropriately representing the contractually obliged self-interested nature of construction claims negotiation. It clearly addresses claims negotiation participants' attempts to maximise their benefits and avoid the risk of conflict result.

b) The learning mechanism incorporated in the MASCOT prototype greatly enhances the negotiation efficiency. Agents' inference abilities allow them to estimate the opponent's key negotiation features (e.g. reservation value) based on their domain knowledge and the opponent's offers during negotiation - an important ability of human negotiators. With the power of learning, negotiations converge quickly and positively.

c) The MASCOT prototype development process, particularly the method adopted for the development of the MASCOT negotiation mechanism (e.g. the selection of the economic theory model and the behaviour theory model) provides a general approach to building agent negotiation mechanisms in other application environments.

d) The ZEUS agent building toolkit, integrated with Java external program, provides an effective tool to develop multi-agent system prototypes. As shown in this research, the co-operation between ZEUS primitive tasks, rulebases and external Java programs is able to fulfil various complex tasks. Also, the ZEUS development methodology such as role modelling, implementation design (e.g. defining system ontology, agent organisations, and negotiation protocol and strategies), and system realisation (e.g. defining agents, tasks and utility agents) provides system developers a systematic approach to build a multi-agent system prototype.
• MAS offers great potential for the improvement of construction claims management

Besides the work on contract, risk management and procurement systems, much effort have been made to improve the efficiency and effectiveness of construction claims management using information technologies such as: Adams (1988), Diekmann and Gjertsen (1992) and Keane (1994). Information technology provides two crucial advantages: the power to communicate over great distances, and the analytic power to help users engage in private and structured analysis (Shell, 1995). However, conventional information technologies such as expert systems and the Internet can either only facilitate communications between negotiation participants or only support decision-making for an individual party. Therefore, their functions in facilitating claims management are limited.

The power of MAS lies in the fact that it possesses both the analytic power and communication ability, and goes far beyond that. Each individual agent's capabilities of autonomy, responsive, learning, and more importantly the ability to work as a virtual society provide great potential and flexibility for facilitating group work, especially distributed groups. The successful application of MAS in facilitating claims negotiation shows that MAS has a great potential to tackle the more complex issues in the whole claims management process. For example, a sophisticated MAS could be extended from the MASCOT prototype, in which agents with the knowledge of contract documents, project schedules and progresses, site reports, principles of compensation calculation, and negotiation mechanism can trace the reasons of a claim, document it, and settle it by collaborating with other agents. Such a system not only increase the efficiency of overall claims management by reducing the complex paper work, and time-consuming meetings and arguments, but also improve the effectiveness of the claims management by avoiding the errors such as late submission of claims notification, misrepresent claims information and so on.
• MAS have great potential for solving the fragmentation problems of the construction industry

Besides facilitating construction claims negotiation and management, multi-agent systems are also a tool with great potential for enhancing many construction engineering and management activities. MAS provides a very useful framework within which social aspects of intelligent behaviour can be modelled, analysed, and evaluated under a wide variety of domains, behaviour and knowledge assumptions. Therefore, they are particularly powerful in solving (Chapter 3):

a) problems requiring the interconnecting and inter-operation of multiple, autonomous, "self-interested" existing legacy systems (e.g. expert systems or decision-support systems);

b) problems whose solutions draw from distributed autonomous experts. Agents in a multi-agent system are often designed based on functionality: such systems have a modular design and hence can be easier to develop and maintain;

c) problems that are inherently distributed in nature (e.g. co-ordination of self-interested agents, etc.)

d) problems whose solutions require the collation and fusion of information, knowledge or data from distributed, autonomous and selfish information sources.

Most of the above problems are common in the construction industry. A key benefit for the application of MAS in the construction industry is that MAS provides a decentralised approach to model construction engineering and management activities between distributed project participants. Although the distributed problem of the industry has been addressed for a long time, no tool, thus far, can resolve it effectively. The incorporation of MAS into the industry represents a novel approach to resolving the fragmentation problem. Some other key benefits include effective decomposition of large-scale problems; improved collaborative and concurrent working; and easier and cheaper access to specialist information. The use of MAS is expected to result in increased
competitiveness of the industry as the decentralisation of complex, large-scale problems and the collaborative input to their resolution, will lead to better quality, more economic, safer and more optimal designs.

9.4 RECOMMENDATIONS FOR FURTHER DEVELOPMENT

Since the studies on construction claims negotiation and MAS are relatively new, many problems need to be further addressed. This research project has revealed a number of such areas, which mainly include: the further development of the MASCOT model, the development of MAS for construction claims management, and the applications of MAS in the construction industry.

9.4.1 Further Development of the MASCOT Prototype

The use of MAS to perform negotiations instead of human beings, especially in the complex claims negotiation, is indeed very creative and challenging. Many aspects of the MASCOT prototype need to be further explored. For example:

- The encoding of claim participants' domain knowledge (Knowledge representation): in the MASCOT model, a party's domain knowledge about its opponent was encoded in terms of the opponent's "negotiation habits" (e.g. the engineer usually has a tendency to make his offers 10% lower than his reservation value). Although this assumption has been discussed by other researchers, and proved to be suitable for certain negotiation groups, it is far from real claims negotiation situation. Therefore, effort needs to be made on how to model a negotiator's domain knowledge, and transfer the knowledge into his agent's domain knowledge.

- Further development of the negotiation mechanism: besides the possible involvement of a mediator agent in the system, the improvement on some other areas
is also necessary to make the negotiation mechanism more robust and realistic. For example,
a) One important aspect is the development of an argument-based negotiation protocol, where an agent makes its offers with supporting information such as the related contract clauses, breakdown analysis, or site information. Thus, the system can not only handle the quantitative issues, but also deal with qualitative items. This is essential for claims negotiation.
b) Another important improvement is the adoption of a dual responsiveness model instead of a learning model where each agent not only estimates the opponent's negotiation features, but also traces its own negotiation history. Therefore, an agent's concession history could be an important criterion for its decision of concession rate. A major benefit of this approach is that it allows an agent to change its risk attitudes during the negotiation process.

- **Involvement of a mediator agent:** The involvement of a mediator agent, which can play a role like a human mediator in claims negotiation will be very useful for further development of the prototype, particularly when the system involves many agents. The mediator agent should have the database of project contract documents and various claims cases, thus, it can be able to conduct case-based reasoning to anticipate the possible outcomes of claims. Furthermore, such a mediator agent will play a facilitator role during the prototype implementation process.

- **The empowerment of the MASCOT agents:** The importance and amount of claims varies from case to case. Like human negotiators, it is important to determine the authority of agents in the negotiation. The owner should decide the maximum amount and the type of claims that an agent can handle. This point concerns the industry participants. Similarly, legal issues may be also involved in the agent negotiation system. All these items need to be further investigated in the industry.
9.4.2 Development of MAS for Construction Claims Management

There is a great possibility to extend the MASCOT prototype to facilitate various activities in construction claims management process. To do this, several important aspects of the prototype need to be further developed. For example:

- **Incorporating other legacy systems:** It is expected that the MASCOT system could incorporate other legacy systems of the industry, such as project scheduling systems, costing estimating systems, documentation systems, expert systems, and project databases. By doing so, agents could obtain the necessary information from these systems to support their claims. The capability of the MASCOT system therefore will be considerably enhanced.

- **Improvement of the collaboration mechanism:** The negotiation mechanism of the MASCOT system is particularly designed for construction claims negotiation. To make the system facilitate the whole claims management process, the most important work is to improve the system collaboration (negotiation) mechanism so that it can fulfil various complex tasks at different claims management stages. As a result, the collaboration mechanism should first consider agents’ different requirements and possible strategies at different claims stages. Furthermore, the system architecture may also need to be reconsidered because more agents such as subcontractors may also need to be involved in the system.

9.4.3 Application and Development of MAS in Construction

MAS offer the industry a powerful tool to improve many engineering and management activities, such as: collaborative design, concurrent engineering, project scheduling and control, supply chain management, material supply management, decentralised project management, and so on. Agents can represent different project participants. For example, a few application scenarios are briefly described as follows:

- **Collaborative design,** where agents represent different design groups or some related groups such as material suppliers. Each agent has its minimum requirements
determined either by technical specifications or safety regulations or cost constraints, whilst all the parties have a common objective to reach a design which should satisfy all the party’s minimum requirements. Some of these issues have been addressed in ADLIB project (Anumba et al., 2002).

- **Concurrent engineering**, where agents represent the client, architect, designer, and contractor. Each party has its own knowledge, expertise and constraints, its objective is to inform the other agents of its requirements and to plan its activities according to the other agents’ requirements. The system allows agents to break the time barrier in design and construction process. Therefore, in such a case, the motivation for co-operation is probably stronger than that of competition. In a more general sense, MAS can be used to facilitate the collaboration between project participants in project lifecycle sequence (i.e. the so-called supply chain management). Udeaja and Tah (2001) have developed an agent-based material supply chain integration system, where agents represent different parties in project material procurement system to facilitate their negotiation process, improve the fairness of the process, and reduce the bidding time. Their model addresses some of the key points, and indicates the great potential for the further application of MAS in supply chain management.

- **Project scheduling and monitoring**, where agents represent project management team and site construction groups. The management agent may plan and schedule the project, and monitors the site groups’ progress. Each site group, with its own time constraints, reports to the management agent on its progress. The update on the progress depends on the negotiation between the management agent and all the construction groups. Unlike the previous two cases, the management agent may play a dominant role in this system (i.e. the system has a hierarchical structure).

Although the objectives (e.g. co-operation or competitive), system architectures (e.g. relationships between agents), and co-operation mechanisms are different, properly developed MAS would provide powerful solutions to improving the efficiency and effectiveness of these activities. The development of MAS in these situations needs both construction expertise and computer knowledge. Since claims negotiation represents a
very complex negotiation in construction, the methodology adopted in this research project could also be used for developing other MAS for construction applications. Essentially, several important questions need to be answered during the development of the system, which include:

- How to transfer the complex industry problems into an essential MAS conceptual model. This involves: analysing the particular industry activity to build a generic model that represents the nature and essential characteristics of the industry activity; identifying the key reasoning model for the MAS; and addressing the key problems that need to be resolved by MAS.

- How to develop the system architecture, define agent relationships, and address agent architecture based on the identified industry conceptual model. This also includes how to formulate, describe, decompose, and allocate the industry problems and synthesise results among the agents.

- How to develop the system’s co-operation mechanism (e.g. negotiation protocols and strategies) to ensure that agents act coherently in resolving the industry problems.

- How to enable individual agents to represent and reason about the actions, plans, and knowledge of other agents in order to co-ordinate with them. How to reason about the state of their co-ordinated process.

- What implementation toolkit will be adopted? How to enable agents to communicate and interact? What communication languages and protocols to use? What and when to communicate.

- How to effectively balance local computation and communication. More generally, how to manage the allocation of limited resources. How to avoid or mitigate harmful overall system behaviour, such as chaotic or oscillatory behaviour.

- How to evaluate the developed systems and how to further improve them, and so on.
9.5 CLOSING REMARKS

The continuous rise in the number of construction claims and disputes shows the need to improve various aspects of construction claims management. The research in this thesis has demonstrated how a multi-agent system could be developed to facilitate claims negotiation and resolve the problems that were difficult to resolve through other approaches. Agents' abilities of autonomy, co-operation and learning allow them to work as a virtual society, through a proper co-operation mechanism, to achieve a certain goal that may be for the group’s benefit, for individual’s best benefits or for both. This characteristic provides MAS with considerable potential in resolving the fragmentation problems of the construction industry. Moreover, the Internet facilitates the wide application of MAS. It is expected that MAS applications could enhance the industry’s engineering and management activities and resolve many industry problems that traditional isolated systems fail to resolve.
REFERENCES


APPENDIX A

PUBLICATIONS ARISING FROM THIS RESEARCH

Journal papers:


Conference papers:


APPENDIX A


Proceedings:

QUESTIONNAIRE FOR EVALUATING THE MASCOT PROTOTYPE

The aim of this work is to evaluate a software prototype, the MASCOT model, which was developed to facilitate construction claims negotiation using multi-agent systems. This prototype was developed through the ZUES agent building toolkit, an engineering approach to the design and construction of collaborative agent systems. The completion of the questionnaire will follow a demonstration of the MASCOT prototype.

Information on respondents:

1. The respondent’s experience has involved working as a: Contractor ( ), Engineer ( ), Architecture ( ), Client ( ) or Others ( ).

2. Position of the respondent: ____________________________________________________________.

3. Respondent’s experience in the construction industry _______ years.

4. Is the respondent familiar with construction claims? Yes ( ) / No ( ).

5. Does the respondent have any experience in applying information technology in construction management? Yes ( ) / No ( ).

Evaluation of the MASCOT Prototype

Please complete the following questions based on the demonstration you have just seen. Where a scale is provided, please state the level to which you agree or disagree with the statement. 1 refers to a low ranking, and 5 refers to a high ranking.
## Construction Claims Negotiation

<table>
<thead>
<tr>
<th>Item</th>
<th>Statements</th>
<th>Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Inefficiency is one of the major problems in current construction claims negotiation.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Two factors influence a negotiators' attitude in claims negotiation:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>a) try to maximise his/her own benefits (e.g., more monetary compensation, or long time extension);</td>
<td></td>
</tr>
<tr>
<td></td>
<td>b) try to avoid breaking the negotiation (otherwise, s/he has to give up the claim, or go to arbitration or litigation);</td>
<td></td>
</tr>
<tr>
<td></td>
<td>within these two factors,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- a) is more important than b)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- b) is more important than a)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- a) is as important as b)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- It depends on the claim cases</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>A negotiator can have a certain perception about the opponent's sham offer (exaggerated amount) after s/he negotiates with the opponent for some time in a project.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Two factors may influence a negotiator's decision of concession amount:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>a) the gap between his/her current offer and his/her reservation value;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>b) his/her estimate of the opponent's reservation value and the opponent's current offer;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>within these two factors:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- a) is more important than b)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- b) is more important than a)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- It depends on the claim case</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Negotiation tactics play an important role in deciding the outcome of the claim negotiations</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>The involvement of the client in the claims negotiation will be helpful for the negotiation</td>
<td></td>
</tr>
</tbody>
</table>
7 In your opinion, what is the major barriers for construction claims negotiation:


8 It is possible to adopt a computer-aided approach to improve construction claims negotiation

### Multi-agent System

<table>
<thead>
<tr>
<th>Statement</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Compared with the other information technologies, such as expert systems and decision making systems, multi-agent systems is more suitable to solve the problems caused by the fragmentation of the construction industry</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### MASCOT Model

<table>
<thead>
<tr>
<th>Statement</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 How well does the MASCOT model ensure that all the essential perspectives of construction claims negotiation are represented?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 There are seven input information in the MASCOT model:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the negotiated claim item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the negotiator's reservation value for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the negotiator's optimum value for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the negotiation deadline for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the opponent's reservation value for this item</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the opponent's negotiation habit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• the confidence about these input information</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(1 is poor, 5 is excellent)
To what extent does this input information represent the information needed for real negotiations?

Please specify the problems if you are not satisfied with these input information:

- 
- 
- 

Which kind of information needs to be added or deleted?

- 
- 
- 

3 How effectively does the modified Monotonic concession protocol in the MASCOT model?

4 How effectively do the listed MASCOT negotiation strategies?

- the adoption of the maximum risk acceptability as the criteria to concede
- the learning ability of agents to the claims negotiation
- the time penalty to the claims negotiation
- the involvement of the client agent facilitate claims negotiation
- the extend solution searching facilitate claims negotiation

5 To what extent does the MASCOT system improve the efficiency of the claims negotiation?

6 To what extent does the MASCOT system improve the effectiveness of the claims negotiation?
## APPENDIX B

### General

<table>
<thead>
<tr>
<th>Statement</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  This study has analysed the characteristics of construction claims negotiation, and identified the problems properly.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2  How useful is the MASCOT system to different participants?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Contractor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Engineer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Client</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Others: sub-contractors, suppliers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3  How easy is the MASCOT prototype to use?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4  How useful is the MASCOT system to the overall claims management or project management?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5  What is your overall rating of the system?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6  What is the risk of application of the MASCOT system?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suppose you are using the MASCOT system, to what range will you allow your agent to determine the final negotiation amount?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7  What might discourage people from using it?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## APPENDIX B

### 8 In what ways can MASCOT be improved?

Identify more clearly the characteristics of construction claims negotiation, such as:

- [ ]
- [ ]

Build a more sophisticated negotiation mechanism by considering more human aspects and claim elements

Others, please specify:

- [ ]
- [ ]
- [ ]
- [ ]

### 9 In which areas can the Multi-agent systems be best used in the construction industry?

- a) facilitating engineering design
- b) concurrent engineering
- c) material management
- d) scheduling and control
- e) others, such as:
  - [ ]
  - [ ]
  - [ ]
  - [ ]
The Application of Multi-agent System in Construction Claims Negotiation

(Evaluation Session)

Zhaomin Ren

Outline of Presentation

- Multi-agent System
- Construction Claims Negotiation
- Negotiation Theory
- MASCOT Negotiation Mechanism
- Demonstration
- Evaluation

Multi-Agent System

- A kind of distributed artificial intelligence
- Characteristics: Autonomy, collaboration & corporation, and learning;
- MAS in Negotiation: agents think locally. Agents, on behalf of their owners, negotiate with each other to reach an agreement.

Characteristics of Construction Claims Negotiation

- A bounded self-interested negotiation
- Participant-dependent information
- Strategy-influenced process
- Time - an important factor

Problems of Construction Claims Negotiation

- Inefficiency
  - negotiation preparation
  - negotiation process
- Involvement of the client
- Unhealthy human factors
**Negotiation Theory**

- Rule Based Theory
- Game theory
- Bargaining theory
- Behaviour Based theory
- Learning model
- Psychological Model
- Joint Decision Making Model

**Process Model**

1. Calculate the claim cost
2. Claim items
3. Initial offer
4. Evaluate offer & make counteroffer
5. Conflict deal
6. Expand negotiation searching
7. New offer
8. Agreement
9. New offer

**Concession Mechanism**

- New offer
- Update its belief of the opponent’s reservation amount
- The opponent’s utility function
- Decision of concession (who, how much)
- Calculate and compare the maximum tolerant risk
- Counteroffer
- Terminology:
  - Reservation value
  - Optimum value
  - Time Penalty
  - Negotiation habit

**Further Work**

- Empowerment of agents
- Encoding of domain knowledge
- Further development of negotiation mechanism
- Proper implementation software systems
- ...
Your Comments