Integrating Metacognition Instruction in Interactive Learning Environments

Claudia Amado Gama

Submitted for the degree of D.Phil.

University of Sussex

April, 2004
Declaration

I hereby declare that this thesis has not been submitted, either in the same or different form, to this or any other university for a degree.

Signature:
Acknowledgements

There are several people without whom this thesis would not have been at all possible and whom I’d like to thank:

- First of all, my supervisor, Professor Ben du Boulay for his guidance, understanding, and many useful suggestions on this work.
- Dr. Rosemary Luckin and Dr. Richard Cox for their support and comments on my work.
- The “girls” from COGS School Office for their always prompt help, and specially Aruna Sharma for her good humour.
- Dr. David Weir and Dr. Christopher Mulvey, who gave me opportunity to act as a monitor in their problem solving classes and allowed me to pilot my research ideas with their students.
- The Brazilian Government CNPq for the financial support.
- My family for their support, phone calls, visits, and help with Lucas, specially to Andrea, Marcia, and Thildo.

I would like to acknowledge all that reviewed the thesis document, correcting my brazilian-english writings and making most useful comments on it: Guillaume, Paulo, Gavin, Youli, Kathia, Nicolas, and Carla.

In the School of Cognitive and Computer Science I met many people that in a way or another, taught me a lot about how to do research. Thanks to my PhD mates. We grew together as researchers and as professionals: Fabrice, Nuno, Jorge, Miguel, Anna Lloyd, Genaro, Pablo, and Benjamin. It was great to create the Edutech Study Group together!

Thanks to all members of the IDEAS Lab. The interaction with such a nice group was very rewarding. I will never forget our last Christmas party and the “secret friend” game!

Many thanks to the Brazilian community in Brighton and London - for the parties, gatherings, support, and help: Paulo, Teresa, Carol, Zeca, Andre, Janaina, Rosane, Eduardo, Anabelle, and Luciano. My Brazilian friends Max and Ana gave me all the support I needed in my first months on Campus. Thanks guys!

And my dear Friend Youli: thanks Mini ... we shared the office, the tears, the laughs, the fears, and discovered together “the path”.

A have a special thanks to my husband Guillaume for his endless support and encouragement.

A very very special thought goes to my son Lucas, who shared with me these years in England and discovered with me the happiness and burdens of adapting to a new country and lifestyle. Thanks for your messy and cheerful presence, and most of all, for loving me.
Integrating Metacognition Instruction in Interactive Learning Environments

Claudia Amado Gama

Abstract

Metacognition is a higher order thinking process responsible for active control over cognitive processes. It is an important ingredient for learning as empirical studies have shown that metacognitively aware students perform better than less aware ones. Theories of metacognition emphasize the importance of self-reflection as a means to improve one’s ability to monitor, self-direct and evaluate one’s learning processes.

Some ways of including metacognition instruction in Interactive Learning Environments (ILEs) have been suggested. In practice, however, most ILEs overlook it, not least because it is very difficult to provide adequate guidance on metacognition to students.

This thesis puts forth a metacognition instruction model, named the Reflection Assistant (RA), that focuses on the following metacognitive skills: (1) problem understanding and knowledge monitoring, (2) selection of metacognitive strategies, and (3) evaluation of the learning experience. The RA automatically builds a metacognitive profile of the student based on two measures: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB). The KMA measures the accuracy of the student’s knowledge monitoring. The KMB detects any systematic bias the student might exhibit in her knowledge monitoring, enabling us to categorize students as pessimistic, optimistic, realistic, or random.

We tested the RA model by implementing a full ILE for algebra word problems called MIRA. The experimental version of MIRA included an implementation of the RA while the control version did not. An empirical study conducted with 27 undergraduate students showed that students who performed the reflective activities spent more time on tasks and gave up on fewer problems. Moreover, this group answered significantly more problems correctly than the control group. Evidence of a positive effect of the RA model on the students’ metacognition was observed. These results suggest that the RA model was beneficial for the learning process.

Submitted for the degree of D. Phil.

University of Sussex

April, 2004
Contents

1 Introduction 1
   1.1 Metacognition in Educational Practice ............................ 1
   1.2 Purpose of the Thesis ................................................. 4
   1.3 Research Methodology ................................................. 6
   1.4 Structure of the Thesis ................................................. 7

2 Metacognition and Reflection 9
   2.1 Introduction ......................................................... 9
   2.2 What is Metacognition? .............................................. 9
      2.2.1 Origins ............................................................ 10
      2.2.2 A fuzzy concept ................................................. 10
      2.2.3 Distinguishing metacognition from cognition ................. 11
   2.3 Models of metacognition ............................................ 12
      2.3.1 Flavell’s model of cognitive monitoring ..................... 13
      2.3.2 Brown’s model of metacognition ................................. 14
      2.3.3 Tobias & Everson’s hierarchical model ....................... 15
   2.4 Assessment of metacognition: methods and instruments ........... 17
      2.4.1 The MAI inventory ............................................... 18
      2.4.2 The Knowledge Monitoring Assessment instrument ............ 18
   2.5 Metacognition and Instruction ..................................... 20
      2.5.1 Individual differences in metacognitive development ....... 22
      2.5.2 The importance of the knowledge monitoring skill ........ 23
   2.6 The Role of Reflection in Metacognition ........................... 24
      2.6.1 What is Reflection and Self-reflection ....................... 24
      2.6.2 Self-reflection: an essential part of metacognition ......... 25
   2.7 The design of instructional metacognitive activities ............ 26
      2.7.1 Reflective questions and reflective prompts ............... 26
      2.7.2 Metacognitive scaffolding .................................... 27
      2.7.3 Modelling ....................................................... 28
      2.7.4 Self-questioning ............................................... 28
      2.7.5 Thinking aloud and Self-explanations ....................... 28
      2.7.6 Self-assessment ................................................. 29
      2.7.7 Graphic organizers .............................................. 29
   2.8 Problem solving and metacognition ................................ 29
      2.8.1 The characteristics of problems and how metacognition helps 30
      2.8.2 Heuristics in problem solving .................................. 31
2.8.3 Self-Regulation and goal management ........................................... 33
2.9 Conclusion ......................................................................................... 35

3 Metacognition in Interactive Learning Environments 37
3.1 Introduction ....................................................................................... 37
3.2 Computers as Metacognitive Tools .................................................... 37
3.3 The Design of Metacognition in ILEs .................................................. 38
  3.3.1 A classification schema of metacognition in ILEs ......................... 38
3.4 ILEs that Implement Metacognitive Training ....................................... 40
  3.4.1 Graphical reification in ILEs for metacognitive development .......... 41
  3.4.2 Reflection through follow-up activities ........................................ 46
  3.4.3 Metacognition through collaboration ......................................... 47
  3.4.4 Self-explanations in ILEs triggering metacognition ..................... 48
  3.4.5 Self-assessment in ILEs ............................................................... 49
3.5 Designing automatic measures of metacognition .............................. 51
3.6 Conclusion ......................................................................................... 56

4 The Reflection Assistant Model 58
4.1 Introduction ....................................................................................... 58
4.2 The Objectives of the RA .................................................................. 58
4.3 Theoretical Basis .............................................................................. 60
  4.3.1 Metacognitive skills modelled .................................................... 61
  4.3.2 Domain of application: ILEs for problem solving ....................... 61
4.4 Design of the Reflection Assistant Model ......................................... 64
  4.4.1 The Architecture of the RA Model ............................................. 65
  4.4.2 Metacognitive inference engine ................................................ 66
  4.4.3 Definition of the Knowledge Monitoring Accuracy (KMA) measure . 67
  4.4.4 Definition of the Knowledge Monitoring Bias (KMB) measure ........ 69
  4.4.5 Data repositories and activities ................................................ 71
  4.4.6 Adding the RA to an existing ILE ............................................. 79
4.5 Research Questions and Hypothesis ................................................ 79
4.6 Conclusion ......................................................................................... 80

5 The Domain in MIRA: Algebra Word Problems 82
5.1 Introduction ....................................................................................... 82
5.2 Reasons for Choosing the Domain of Algebra ................................... 83
5.3 Relational Algebra Word Problems .................................................. 83
  5.3.1 The process of solving algebra word problems .......................... 83
  5.3.2 The main misconceptions and difficulties .................................. 85
5.4 Algebra Problem Solving in ILEs ...................................................... 85
5.5 Helping to Translate Problems into Equations .................................. 87
  5.5.1 Definitions of concepts used ..................................................... 88
  5.5.2 A heuristic to help with problem translation ............................... 88
5.6 Designing a Tool to Support Word Problems Translation .......................... 89
5.6.1 Requirements of the PAL Tool .................................................. 89
5.7 A Pilot Study with the PAL Tool ..................................................... 90
5.7.1 Goals of the study .................................................................... 91
5.7.2 Methodology ............................................................................. 92
5.7.3 Discussion .................................................................................. 93
5.8 The Implementation of the PAL Tool .................................................. 93
5.8.1 Interface design ......................................................................... 94
5.8.2 The help features in the PAL Tool .............................................. 95
5.8.3 Limitations of the PAL Tool ....................................................... 96
5.9 Conclusion ...................................................................................... 98

6 A typical learning Session with MIRA ....................................................... 99
6.1 Introduction .................................................................................... 99
6.2 Overview of activities in MIRA ....................................................... 99
6.3 A typical Learning Session .............................................................. 100

7 The design of MIRA ............................................................................ 111
7.1 Introduction .................................................................................... 111
7.2 Main Features ................................................................................ 111
7.3 Principles ........................................................................................ 112
7.4 Algebra Word Problems in MIRA .................................................... 112
7.5 Stages within MIRA ........................................................................ 113
7.6 Design of the Reflective Activities .................................................... 114
7.6.1 Comparison of knowledge monitoring and performance ............... 114
7.6.2 Analysis of knowledge monitoring state (KMA/KMB) ................... 115
7.6.3 Self-assessment of understanding and difficulty ........................... 116
7.6.4 Selection of metacognitive strategies ......................................... 116
7.6.5 Evaluation of problem solving experience .................................. 117
7.7 Design of the Problem Solving Activities ......................................... 119
7.7.1 Problem solving resources ........................................................ 119
7.7.2 Checking answer ......................................................................... 122
7.7.3 Answering quiz ........................................................................... 122
7.7.4 Checking the teacher’s solution ................................................. 122
7.8 Conclusion ....................................................................................... 123

8 Experimental Study of MIRA ................................................................. 125
8.1 Introduction ...................................................................................... 125
8.2 Description of the Experiment .......................................................... 125
8.2.1 Objective .................................................................................... 125
8.2.2 Experimental design ................................................................. 126
8.2.3 Hypotheses ................................................................................ 126
8.2.4 Participants ................................................................................ 126
8.2.5 Tasks .................................................. 127
8.2.6 Materials ............................................. 130
8.3 Preparation for Data Analysis ........................................... 132
8.3.1 Data Normality ......................................... 132
8.3.2 Categorizing data for analysis purposes ...................... 132
8.3.3 Definition of measures of performance ......................... 133
8.4 Exploratory Analysis of the Experiment .............................. 134
8.4.1 General profile of the experimental group ..................... 134
8.4.2 General profile of the control group .......................... 134
8.4.3 Comparison of groups’ general profile ......................... 135
8.4.4 Groups initial profile for algebra problem solving ............ 135
8.5 Testing Hypotheses related to Problem Solving ................... 137
8.5.1 Overall performance ..................................... 137
8.5.2 Perseverance in problem solving ............................ 141
8.6 Testing Hypotheses related to Metacognition ..................... 141
8.6.1 Time management ....................................... 141
8.6.2 Changes in knowledge monitoring accuracy (KMA) ............ 142
8.6.3 Changes in knowledge monitoring bias (KMB) ................. 144
8.7 Observations about Correlations .................................... 145
8.8 Analysis of interaction with MIRA ................................ 146
8.8.1 Evaluation of resources in MIRA ............................ 146
8.8.2 Evidence of Metacognition in the Interviews .................. 148
8.8.3 Use of Strategies ....................................... 150
8.9 Discussion of Results ....................................... 151
8.9.1 Limitations of the Experiment ............................... 152
8.10 Conclusion ................................................. 153

9 Conclusion: Lessons Learned ............................................ 154
9.1 Contributions of this Research ................................. 154
9.1.1 The creation of the Reflection Assistant model for metacognition training 155
9.1.2 The development of an ILE dedicated to algebra problem translation with a reflective approach ................................. 156
9.2 Limitations of the research ..................................... 157
9.3 Future research directions ..................................... 158
9.3.1 Improvements in the RA Model ............................ 158
9.3.2 Improvements in the MIRA Learning Environment .......... 159
9.4 Final Remarks ............................................. 159

A Algebra Word Problems ............................................... 171
A.1 Classification ............................................. 172
A.1.1 Types of problems ....................................... 172
A.1.2 Difficulty level .......................................... 172
A.1.3 Comprehension Level ..................................... 173
Contents ix

A.2 Problems presented in MIRA ................................................. 173

B MIRA: Technical Documentation ........................................... 179
   B.1 User Registration ......................................................... 179
   B.2 Log Files ................................................................. 180
   B.3 Directory Structure and Data Files ................................. 181
   B.4 Sample of Log Files .................................................... 183
   B.5 Main Classes in MIRA .................................................. 188

C Experiment Analysis: Graphs and Tables ............................... 192
   C.1 Normality test .......................................................... 192
   C.2 Initial profile of the groups .......................................... 193
   C.3 Differences in pre-test and post-test scores ..................... 196
   C.4 Group differences in problem solving performance in MIRA 197
   C.5 Analysis of KMA and KMB ............................................. 200
   C.6 Analysis of correlations .............................................. 202

D Experiment Materials ......................................................... 209
List of Figures

2.1 Flavell’s model of metacognition. .................................................. 13
2.2 Brown’s model of metacognition. .................................................. 15
2.3 Tobias & Everson’s hierarchical model of metacognition. ................. 16
2.4 Polya’s Heuristic for Maths Problem Solving ............................... 32
2.5 Comparison of problem solving behaviour in Schoenfeld’s intervention. 36

3.1 Classification Schema for Metacognition Design in ILEs .................. 39
3.2 Snapshot of IQ Form System. ....................................................... 50
3.3 Embedded Metacognitive Assessment Prototype - snapshot 1. .......... 52
3.4 Embedded Metacognitive Assessment Prototype - snapshot 2. ......... 53
3.5 Embedded Metacognitive Assessment Prototype - snapshot 3. ........ 54

4.1 Integration of the RA Model with problem solving learning environments. 59
4.2 Theoretical view of metacognition in the RA Model. ....................... 60
4.3 Problem solving conceptual stages and the RA Model. ..................... 62
4.4 The RA in our classification of metacognitive training ....................... 64
4.5 Architecture of the Reflection Assistant Model. ............................ 65
4.6 Mock-up of the graphic component of the first pre-reflection activity. 73
4.7 Mock-up of graphic component of the second pre-reflection activity. 74
4.8 Mock-up of graphical reification for post-task reflective activity. ....... 77
4.9 Mock-up of optional graph to compare performance between similar problems. 78

5.1 Conceptual stages of algebra problem solving ............................... 84
5.2 Paper prototype of the PAL Tool ................................................... 90
5.3 Student using the paper version of the PAL Tool ............................. 91
5.4 Results of pilot study: ability to identify parts of the problem ......... 92
5.5 Identifying relations in the PAL Tool. .......................................... 95
5.6 The Hints facility in the PAL Tool. .............................................. 96
5.7 The “Give One” facility in the PAL Tool. .................................... 97

6.1 Sequence of activities in MIRA. ................................................... 100
6.2 MIRA: Login screen with navigation sidebar in the background. .... 101
6.3 MIRA: General View Window ..................................................... 102
6.4 MIRA: Reflection on predicted and demonstrated knowledge skills. ... 102
6.5 MIRA: Analysis of knowledge monitoring state (KMA / KMB). ....... 103
6.6 MIRA: Help Window. ............................................................... 104
6.7 MIRA: pre-assessment on problem understanding and difficulty. .... 104
6.8 MIRA: Planning strategies. ....................................................... 105
6.9 MIRA: Problem solving window. .................................................. 106
6.10 MIRA: PAL Tool. ................................................................. 106
6.11 Problem solving resources: Calculator and Answer to Problem. .......... 107
6.12 MIRA: Check Answer. ............................................................ 108
6.13 MIRA: Feedback on Answer Given ............................................. 108
6.14 MIRA: Teacher’s Solution. ....................................................... 109
6.15 MIRA: Post Reflection. ........................................................... 110
7.1 Stages in MIRA. ................................................................. 113
7.2 Instructional messages presented in post-task reflective activity. .......... 120
7.3 MIRA’s resource: Library of Problems ........................................ 121
7.4 The quizzes in MIRA .............................................................. 123
8.1 Room setup for the experiment .................................................... 130
8.2 Comparison of groups’ general profile .......................................... 136
8.3 Graphical view of pre-test and post-test scores for experimental group ...... 138
8.4 Graphical view of pre-test and post-test scores for control group .......... 139
8.5 Graphical view of post-test pre-test differences ............................... 139
8.6 Performance in MIRA ............................................................. 141
8.7 Number of problems in which students have seen the time-up message .... 142
8.8 Graphical view of KMA categories distribution ................................ 143
8.9 Mean values of KMA for the groups ............................................. 144
B.1 Registration in MIRA ............................................................. 180
B.2 Structure of directories of MIRA ................................................. 182
B.3 Diagram of MIRA classes. ......................................................... 191
C.1 Groups’ pre-test performance profile ........................................... 194
C.2 Graph with KMB categories’ distribution ...................................... 203
C.3 Scatterplots involving pre-test, post-test, and KMA ......................... 206
C.4 Scatterplot involving KMA and post-test ..................................... 207
C.5 Scatterplots involving motivation in maths and other variables ............ 208
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Common methods applied in the assessment of metacognition.</td>
<td>17</td>
</tr>
<tr>
<td>2.2</td>
<td>Possible values in the Knowledge Monitoring Assessment Instrument</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>Schoenfeld’s method for metacognition teaching</td>
<td>34</td>
</tr>
<tr>
<td>3.1</td>
<td>Examples of ILEs that support metacognition.</td>
<td>44</td>
</tr>
<tr>
<td>3.2</td>
<td>Experimental Design of the Embedded Assessment System</td>
<td>54</td>
</tr>
<tr>
<td>3.3</td>
<td>Measures for the Embedded Assessment System</td>
<td>55</td>
</tr>
<tr>
<td>4.1</td>
<td>Matrix of values for Knowledge Monitoring Accuracy (KMA)</td>
<td>69</td>
</tr>
<tr>
<td>4.2</td>
<td>KMA classification</td>
<td>69</td>
</tr>
<tr>
<td>4.3</td>
<td>Matrix of values for Knowledge Monitoring Bias (KMB)</td>
<td>70</td>
</tr>
<tr>
<td>4.4</td>
<td>KMB classification</td>
<td>71</td>
</tr>
<tr>
<td>4.5</td>
<td>Data repositories of the RA Model</td>
<td>72</td>
</tr>
<tr>
<td>4.6</td>
<td>Data repositories provided by the ILE</td>
<td>73</td>
</tr>
<tr>
<td>7.1</td>
<td>Pre-task reflection in MIRA: self-assessment questions &amp; answers</td>
<td>116</td>
</tr>
<tr>
<td>7.2</td>
<td>Post-task reflection: variables evaluated in the feedback message</td>
<td>118</td>
</tr>
<tr>
<td>8.1</td>
<td>Experimental Conditions for MIRA's evaluation</td>
<td>126</td>
</tr>
<tr>
<td>8.2</td>
<td>Description of the subject groups</td>
<td>127</td>
</tr>
<tr>
<td>8.3</td>
<td>Organization of the experiment sessions</td>
<td>128</td>
</tr>
<tr>
<td>8.4</td>
<td>Mean KMA Scores</td>
<td>142</td>
</tr>
<tr>
<td>8.5</td>
<td>Table of frequencies: pre-test KMB vs. post-test KMB</td>
<td>145</td>
</tr>
<tr>
<td>8.6</td>
<td>Students’ evaluation of problem solving resources in MIRA.</td>
<td>147</td>
</tr>
<tr>
<td>8.7</td>
<td>Students’ evaluation of the reflective activities</td>
<td>148</td>
</tr>
<tr>
<td>A.1</td>
<td>Algebra word problems presented in MIRA</td>
<td>175</td>
</tr>
<tr>
<td>C.1</td>
<td>Normality test</td>
<td>192</td>
</tr>
<tr>
<td>C.2</td>
<td>General profile of the experimental group</td>
<td>193</td>
</tr>
<tr>
<td>C.3</td>
<td>General profile of the control group</td>
<td>193</td>
</tr>
<tr>
<td>C.4</td>
<td>Testing differences between groups’ pre-test scores</td>
<td>194</td>
</tr>
<tr>
<td>C.5</td>
<td>Testing groups’ GCSE scores differences</td>
<td>195</td>
</tr>
<tr>
<td>C.6</td>
<td>Correlation test: GCSE scores and pre-test</td>
<td>195</td>
</tr>
<tr>
<td>C.7</td>
<td>Descriptive values for pre-test and post-test</td>
<td>196</td>
</tr>
<tr>
<td>C.8</td>
<td>Wilcoxon test for pre- and post-test scores changes</td>
<td>196</td>
</tr>
<tr>
<td>C.9</td>
<td>Mann-Whitney tests for groups’ score differences</td>
<td>197</td>
</tr>
<tr>
<td>C.10</td>
<td>Test for differences of total problems attempted in MIRA</td>
<td>198</td>
</tr>
<tr>
<td>C.11</td>
<td>Test for groups’ differences of types of responses in MIRA</td>
<td>198</td>
</tr>
</tbody>
</table>
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.12</td>
<td>Test for groups’ differences of problems given up in MIRA</td>
<td>199</td>
</tr>
<tr>
<td>C.13</td>
<td>Test for groups’ differences of time ups in MIRA</td>
<td>199</td>
</tr>
<tr>
<td>C.14</td>
<td>Descriptive values of KMA in pre- and post-test</td>
<td>200</td>
</tr>
<tr>
<td>C.15</td>
<td>Frequencies of the categories of KMA</td>
<td>200</td>
</tr>
<tr>
<td>C.16</td>
<td>Wilcoxon test for KMA changes</td>
<td>201</td>
</tr>
<tr>
<td>C.17</td>
<td>Table of frequencies: pre-test KMA vs. post-test KMA</td>
<td>202</td>
</tr>
<tr>
<td>C.18</td>
<td>Mann-Whitney test for ($\Delta KMA$) differences</td>
<td>202</td>
</tr>
<tr>
<td>C.19</td>
<td>Frequency tables with KMB distribution</td>
<td>203</td>
</tr>
<tr>
<td>C.20</td>
<td>Correlation: pre-test vs. post-test scores</td>
<td>204</td>
</tr>
<tr>
<td>C.21</td>
<td>Correlation: KMA in pre-test vs. pre-test scores</td>
<td>205</td>
</tr>
<tr>
<td>C.22</td>
<td>Correlation: motivation in maths vs. pre-test scores</td>
<td>207</td>
</tr>
</tbody>
</table>
List of Abbreviations of Terms Used

AIED  Artificial Intelligence in Education
ILE(s)  Interactive Learning Environment(s)
ITS(s)  Intelligent Tutoring System(s)
KMA  Knowledge Monitoring Accuracy
KMB  Knowledge Monitoring Bias
MAI  Metacognitive Awareness Inventory
MIRA  Metacognitive Instruction using a Reflective Approach
RA  Reflection Assistant (Model)
Chapter 1

Introduction

1.1 Metacognition in Educational Practice

In their analysis of mainstream education (from elementary to postgraduate levels), Koschmann et al. (1996) argue that it is producing individuals who fail to develop a valid, robust knowledge base; who have difficulty reasoning with and applying knowledge; and who lack the ability to reflect upon their performance and continue the process of learning. Among the failures of the educational system Koschmann et al. (1996) single out the fact that students often do not know when they do not know; and when they are aware of their lack of knowledge, they do not know what to do about it. This ability of knowing what one knows and what one does not know, and thinking about one’s own thinking is often described as metacognitive knowledge or metacognitive skill.

Metacognitive knowledge consists primarily of knowledge or beliefs about what factors or variables act and interact in what ways to affect the course and outcome of cognitive enterprises (Flavell, 1979). The acquisition of this type of knowledge in a learning context produces a distinctive awareness of the processes as well as the results of the learning endeavour. Previous research suggests that metacognition has a number of concrete and important effects on learning. It plays an important role in oral comprehension, reading comprehension, problem-solving, attention, memory, social cognition, and various types of self-control and self-instruction. Metacognitive awareness can lead people to select, evaluate, revise, and abandon cognitive tasks, goals, and strategies (Flavell, 1979).

In general, theories of metacognition focus on (a) the role of awareness and executive management of one’s thinking, (b) individual differences in self-appraisal and management of cognitive development and learning, (c) knowledge and executive abilities that develop through experience, and (d) constructive and strategic thinking (Paris and Winograd, 1990). Theoretical models of metacognition have been proposed and provide a general framework to understand the relationship between the different aspects or components of this phenomenon. One such model, devised by Tobias and Everson (2002) perceives metacognition as a compound of skills and knowledge - knowledge of cognition, monitoring of one’s cognitive and learning processes, and control of those processes. Tobias and Everson have investigated largely the monitoring aspect of metacognition,
based on the assumption that accurate monitoring is crucial in learning and training contexts where students have to master a great deal of new knowledge (Tobias et al., 1999). Thus, their model assumes that the ability to differentiate between what is known (learned) and unknown (unlearned) is a prerequisite for the effective self-regulation of learning. They affirm that this aspect of metacognition supports the development of other metacognitive skills, such as comprehension monitoring, help seeking, planning, and revising.

Recently, many studies have examined ways in which theories of metacognition can be applied to education, focusing on the fundamental question “Can explicit instruction of metacognitive processes facilitate learning?” (Hacker et al., 1998; Hartman, 1998). The literature points to several successful examples (e.g. Derry (1992) and Hacker et al. (1998)), such as Schoenfeld’s studies with maths students (Schoenfeld, 1985, 1987). In his work, Schoenfeld illustrates ways that students can be taught to monitor and evaluate their performance on maths problems. For example, students are required to pause frequently during problem solving and ask themselves questions, such as “what I am doing now?”. Learning is both an active and a reflective process. Though we learn by doing, constructing, building, talking, and writing, we also learn by thinking about events, activities, and experiences (Burns et al., 2000). Self-reflective activities encourage students to analyse their performance, contrast their actions to those of others, abstract the actions they used in similar situations, and compare their actions to those of novices and experts (Goodman et al., 1998). So, when students are engaged in metacognitive activities, such as self-assessment, self-explanation, help-seeking, monitoring, or revising, their learning is enhanced.

Recent research indicates that metacognitively aware learners are more strategic and perform better than unaware learners (Garner and Alexander, 1989; Pressley and Ghatala, 1990; Bransford et al., 1999). One explanation is that metacognitive awareness enables individuals to plan, sequence, and monitor their learning in a way that directly improves performance (Schraw and Dennison, 1994). Research also indicates that weaker students benefit even more than stronger students from such activities (Wong et al., 1997; White and Frederiksen, 1998). However, not all students engage spontaneously in metacognitive thinking unless they are explicitly encouraged to do so through carefully designed instructional activities (Bransford et al., 1999; Chi et al., 1989; Lin and Lehman, 1999). It is therefore important to investigate effective ways of including metacognitive support in the design of natural and computer-based learning environments.

Following Davidson et al. (1994) claim for further research on metacognition in instruction, the area of Computers and Education and more specifically of Artificial Intelligence in Education try to address questions like “How can the use of the metacognitive processes be enhanced, both in and out of natural settings?” Designing metacognitive activities in interactive learning environments (ILEs) that focus on improvements at the domain and metacognitive levels is a theoretical and practical challenge. This integrated approach to metacognitive development involves many aspects of student development: on the one hand including academic competence and on the other, fostering knowledge about the self-as-a-learner.

However, most ILEs and Intelligent Tutoring Systems (ITSs) have regarded metacognition training as a by-product, sometimes addressing metacognition in a tangential way, providing embedded reflection tools, but not overtly targeting metacognitive development or analysing the impacts of such tools on students’ metacognition and attitudes towards learning.
Very few attempts have been made to design explicit metacognitive models into ILEs. One example is MIST, a computer-based system that helps students to actively monitor their learning when studying from texts (Puntambekar, 1995). Its design follows a process-based intervention and uses collaboration, reflection, and questioning as tutorial strategies to facilitate students’ planning and monitoring skills. MIST was rather simple from a computational point of view, but demonstrated success in bringing about some changes to the learning activities of students. Another interesting example is the SE-Coach system (Conati and Vanlehn, 2000); it supports students learning from examples through self-explanations. Conati and Vanlehn (2000) devised a student model that integrates information about the students’ actions with a model of correct self-explanations and the students’ domain knowledge. In this way, the SE-Coach system assesses students’ understanding of examples from reading and self-explanation actions and scaffolds improvements in self-explanation.

Researchers in computer-based ILEs have recognized the importance of incorporating metacognitive models into ILE design (Lin, 2001; Aleven and Koedinger, 2000, for example). However, the lack of an operational model of metacognition makes this task a difficult one. Thus, the development of models or frameworks that aim to develop metacognition, cognitive monitoring, and regulation in ILEs is a fruitful and open topic of investigation.

This thesis is concerned with the creation of a computational model of metacognition instruction within ILEs for problem solving. The model defines which metacognitive skills should be modelled, rules for assessing learners’ metacognitive states, and a range of reflective activities. It proposes a new integrated environment that has an action-reflection-action cycle approach to problem solving and metacognitive development. Reflection on the learning processes, on the learner’s abilities, and on the task is viewed here as an approach to improve metacognitive skills.

Despite their general applicability, metacognitive skills may to some extent be domain specific, and their application and benefits may vary across different types of content areas (Davidson et al., 1994; Davidson and Sternberg, 1998). Therefore, it is important to identify the circumstances in which metacognition can be triggered and applied effectively in a particular domain. In problem solving domains there are many metacognitive processes involved in (1) identifying and understanding the problem, (2) mentally representing the problem, (3) planning how to proceed, and (4) evaluating one’s performance. The successful application of these metacognitive processes depends on characteristics of the problem (whether it is a well-structured or an ill-structured problem), the problem solver, and the context in which the problem is presented.

Theories of problem solving have placed a major focus on the role of heuristics. However, it has been shown that extensive domain specific knowledge and a repertoire of heuristics are not sufficient during problem solving. The student must construct some decision mechanism to select from the available heuristics, or to develop new ones, as problem situations are encountered. Furthermore, Polya (1945) argues that the learner must engage in thinking about the various tactics, patterns, techniques, and strategies available to them. In order to do so, a management function must be incorporated to this activity. This executive function of self-reflection is part of the metacognitive skills needed in problem solving. According to John Dewey, philosopher and educational theorist, to be self-reflective is to explore the various external influences affecting consciousness, such as values, conventions, morality, and belief systems. Significantly, there is,
for Dewey (1933), a period of time in which action must be suspended and during which a person makes his or her judgment of action or behaviour, what some have called reflection-in-action and reflection-on-action (Schön, 1987).

We share Polyá’s, Schoenfeld’s, and other researchers’ belief that it is equally necessary to focus on both the cognitive and metacognitive dimensions of problem solving in order to produce good problem solvers. For this reason this thesis investigates how metacognitive skills can be developed in conjunction with problem solving skills. It attempts to ascertain the most relevant metacognitive skills, the best moments in the learning experience to provide metacognitive scaffolding to the student, and the most effective approaches.

1.2 Purpose of the Thesis

This thesis has designed a general computational model of metacognition training for problem-solving learning environments and implemented this model together with a problem-solving ILE. The model, called the Reflection Assistant (RA), uses reflection as a means to improve students’ metacognitive awareness and cognitive monitoring.

The main characteristics of the RA Model are the design of new external representations to reify specific metacognitive skills and the definition of a new automatic metacognitive assessment of knowledge monitoring that is used to build a student metacognitive model.

The RA Model is based on Tobias & Everson’s theoretical model of metacognition, where knowledge monitoring is the primary metacognitive skill to be nourished to achieve a comprehensive metacognitive development. They state that by knowing what one already knows, one becomes aware of the potential knowledge and skills that one can bring to bear, which boosts one’s confidence in one’s own learning. This is an essential attitude for becoming a good maths problem solver. Research in self-regulation also supports this view (Zimmerman, 1998). In our research we follow this assumption and give primary attention to promoting awareness and fostering of knowledge monitoring.

For these reasons we have devised in the RA a mechanism to infer students’ knowledge monitoring levels. In our model two aspects of the learner’s knowledge monitoring skill are modelled: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB). Thus, two scales of measurement for knowledge monitoring are employed:

- The Knowledge Monitoring Accuracy (or KMA) scale provides a profile of the learner’s estimated knowledge of the domain;
- The Knowledge Monitoring Bias (or KMB) scale indicates the bias or inclination in the learner knowledge monitoring ability.

Information about the learner’s prediction of performance compared to her actual performance is used to form beliefs about the current metacognitive conditions and to update the KMA and KMB values.

To achieve its goal, the RA proposes activities that prompt the learner with metacognitive questions, mostly for triggering self-assessment of problem solving and knowledge monitoring. Furthermore, novel graphical reification of metacognitive skills are proposed to provoke self-reflection in order to elicit self-awareness and self-control. The RA Model was used as part of
the design of MIRA, a problem-based interactive learning environment in the domain of algebra word problems.

MIRA is an acronym for “Metacognitive Instruction using a Reflective Approach”. Also, the word *mira* in Spanish is used as an interjection meaning “look!”, “pay attention!”, in the sense of “take time to reflect”. This principle is incorporated in MIRA: it seeks to motivate the learner to observe how she is dealing with the problems and how she is using available resources to complete the learning task.

MIRA was built for the domain of algebra word problems and focuses especially on translating a problem presented in natural language (English) into equations. Thus, at the domain level MIRA provides activities to engage students in developing the skills needed to create linear equations from algebra word problems - it presents a series of algebra word problems and gives students the opportunity to adopt problem-solving strategies such as breaking the problem into individual elements in order to help understanding the task of generating equations. At the metacognitive level, MIRA engages students in reflective activities before and after each problem. The emphasis is on developing students’ metacognitive expertise, particularly their knowledge monitoring skills as well as their ability to reflect on their problem solving processes. The pedagogical strategies include asking students to judge their comprehension and ability to solve the problems proposed, supplying activities to scaffold their reflections, and introducing them to methods for monitoring and reflecting on the learning process. The information generated by the interaction with the problem solving resources available in MIRA and the time spent in activities are used to inform the metacognitive feedback.

The main hypothesis driving the research is that reflection on the metacognitive skills between problem-solving learning activities could enhance:

1. Students’ awareness of their own abilities to solve that type of problem;
2. The selection and use of metacognitive and cognitive strategies;
3. Students’ monitoring process;
4. Students’ performance on the topic in hand.

It was expected that the MIRA system would help students to become more aware of which metacognitive skills they use when they are solving problems and how they use them. By reflecting on these issues it was expected that those students would:

1. Increase their understanding of their own learning processes;
2. Improve their use of metacognitive knowledge and skills;
3. Generate a better understanding of the influence of these skills on the outcomes of the tasks performed.

In summary the goals of this research were:

- To develop a model of metacognitive instruction for interactive learning environments based on problem solving - the Reflection Assistant Model.
- To design and implement an interactive learning environment that incorporated the RA model - the MIRA system.
• To evaluate the effectiveness of the RA and MIRA in fostering metacognitive skills and maths problem-solving performance in students. For this purpose, two versions of MIRA were developed (with and without the reflective activities) and a comparative laboratory evaluation was undertaken.

1.3 Research Methodology

This research used education theories, theoretical constructs in metacognition and research on problem solving, together with empirical observations of real students to inform the design of the RA Model and the MIRA system. We have validated their design with prototypes, which were evaluated by real users. This approach conforms the second and third methodologies for modelling teaching tactics and strategies for tutoring systems as proposed by du Boulay and Luckin (2001). Both methodologies start from a learning theory and derive appropriate teaching tactics and strategies from that theory. The third methodology is also based on observation of students.

Quantitative and qualitative analysis of the evaluation done guided the interpretations of the results and contributions of this research. The “quantitative” aspect stemmed from measuring metacognitive and learning performance variables and analysing the relationships among them with descriptive and inferential statistics. The “qualitative” analysis was based on students’ evaluation of their interaction with MIRA and observations of the students.

Preliminary observations of students in the classroom also supported parts of this research, when it was noted the different ways in which students perceive, experience, and understand their metacognitive skills.

The research was conducted in a phased approach. The phases and main activities are summarised below:

Phase I: Preliminary Studies. The research started in October, 1998. The first year was spent reading the relevant literature in the fields of educational psychology, Interactive Learning Environments (ILEs), and Artificial Intelligence in Education (AIED), and organizing the thesis proposal.

Phase II: Experimentations and Model Building. The second and third years were dedicated to a more in-depth understanding of metacognitive development and problem solving processes, and how students perceived their metacognitive skills in classroom settings. The issue of how metacognition can be taught in an ILE was central to the definition of the Reflection Assistant Model, leading to investigations of appropriate pedagogical approaches to be adopted in the model. As a consequence, some pilot empirical studies were conducted, ranging from questionnaires and observations of classroom activities to the development of a prototype that incorporated a draft version of the Reflection Assistant Model.

Phase III: Implementation and Empirical Study. The first task was to select a domain for the learning topic in the learning environment. The selection of a suitable domain for testing out the model was not straightforward. A shortage of subjects for the main experiment and difficulty accessing literature on problem solving for the domain impacted in the final choice. Initially we tried a topic in the domain of Computer Science (problems about finite states machines). Eventually we chose algebra word problems, as they require less specialized previous knowledge, making it easier to find subjects for the experiment.

The first part of the final period of the research was dedicated to refining the Reflection Assistant Model and to reading the relevant literature in Algebra Problem Solving. Then the
design and implementation of the MIRA learning environment were carried out. The main experiment was conducted and the results were analysed.

1.4 Structure of the Thesis

Chapter 1 introduces the research topic and presents an overview of the research questions, the main hypothesis put forward, and the methodology of the research. It also presents the general organization of the dissertation.

Chapter 2 contains a survey of theories of metacognition, focussing on the role of metacognition in learning through problem solving. It introduces representative theoretical models of metacognition, among which is the componential model of metacognition proposed by Tobias & Everson that is used in this thesis. The issue of measuring metacognitive states and changes is discussed and some methods and instruments are presented. Operational definitions of reflection and self-reflection are discussed. Finally, empirical studies on instruction of metacognition, assessment instruments for metacognition and reflection are used to frame the research questions.

Chapter 3 examines reflective tools and other attempts to design metacognitive activities in Interactive Learning Environments (ILEs) and Intelligent Tutoring Systems (ITSs). A basic design classification is proposed and examples from the literature are organized into this classification.

Chapter 4 discusses the proposed computational framework, which incorporates metacognitive instruction through reflective activities - the Reflection Assistant (RA) Model. It offers a rationale for the RA Model and sets its theoretical bases. The RA Model adapts Tobias & Everson’s hierarchical model of metacognition, and incorporates their knowledge monitoring measurement instrument in the form of rules for an automatic inference engine. The engine produces metacognitive scores and proposes reflective scaffolding according to these scores. The RA architecture and components are presented. It concludes with a more detailed discussion of the research questions and the hypothesis put forward in this thesis.

Prior to the introduction of the MIRA learning environment, Chapter 5 presents the domain of algebra word problems. It focuses on the processes and skills involved in translating a word problem into an equation. Definitions of terms employed in MIRA are presented, along with examples of algebra word problems and some ILEs and ITSs used in this domain. In the MIRA system a specific tool was conceived to help students at the domain level - the PAL Tool. The goals, design features and functionality of this tool are presented.

Chapter 6 and Chapter 7 are dedicated to the MIRA interactive learning environment. Chapter 6 gives an overview of MIRA from the user’s perspective, showing the typical flow of student activities within the system. Chapter 7 describes the design of MIRA. The implementation of the Reflection Assistant model in MIRA is explained together with the adaptations needed to fit the maths topic taught in MIRA.

Chapter 8 presents the experimental study of this research. It was an empirical evaluation of students’ interaction with MIRA focusing mainly on the observation of metacognitive and performance changes. The experiment was undertaken with undergraduate students at the University of Sussex. The design of the experiment is presented, together with the materials and data collected. It also presents the statistical analysis. The observations and analysis are used to formulate more general answers for the research questions proposed. The limitations of the experiment are also
discussed.

Chapter 9 summarises the results and main contributions of this thesis. It considers the validity of the RA Model for promoting the development of metacognitive awareness and growth of reflective behaviour. It re-examines the relationship between metacognitive changes and performance gains. The chapter critiques the approach taken in the thesis and outlines further lines of research.

Parts of the work described here have been presented at the following International Conferences: ED-MEDIA (Gama, 2000b), ITS (Gama, 2000c), AIED (Gama, 2001c), and International Conference on New Technologies in Science Education (Gama, 2001a); and in the Brazilian Symposium of Informatics in Education (Gama, 2003).

Working papers with preliminary results of this research were also presented at the annual Human Centred Technology Postgraduate Workshop at the University of Sussex (Gama, 1999), (Gama, 2000a), (Gama, 2001b), and (Gama, 2002).
Chapter 2
Metacognition and Reflection

2.1 Introduction

This chapter addresses the major theoretical and philosophical underpinnings of the research. It introduces metacognition, examining definitions from different sources which emphasize distinct aspects of this complex phenomenon. It presents models that expand the understanding of what metacognition is and describes its components and their relations. The model of metacognition chosen as the theoretical basis for this research is described - it emphasizes the metacognitive skill called knowledge monitoring. The issue of measuring metacognition is discussed and the methods and instruments that are used in this research are presented.

It then reviews research that shows the role of metacognition in learning and instruction with examples of instructional approaches and methods of helping students acquire metacognitive knowledge and skills to improve learning. Studies concerning metacognition and problem solving are discussed.

Because the approach adopted to foster metacognition is based on reflection on one’s own knowledge and problem-solving skills, definitions of reflection and self-reflection are discussed. We point out that the term reflection is commonly used in different contexts to imply different things in the literature and we define how we use the term reflection in this research.

The chapter ends with a summary of the aspects of metacognition that are emphasized in this research and provides a generic model of metacognition on which the Reflection Assistant Model is built on.

2.2 What is Metacognition?

Metacognition is probably the most actively investigated cognitive process in contemporary research in developmental and instructional psychology (Tobias et al., 1999). Metacognition is a form of cognition, a second or higher order thinking process which involves active control over cognitive processes. It can be simply defined as thinking about thinking or as a “person’s cognition about cognition” (Wellman, 1985, p. 1).
2.2.1 Origins

The term *metacognition* first appeared around 1975 in the work of developmental psychologist John Flavell from Stanford University. He used the term to denote:

“One’s knowledge concerning one’s own cognitive processes and products or anything related to them (...) [and] refers, among other things, to the active monitoring and consequent regulation and orchestration of these processes (...), usually in the service of some concrete goal or objective.” (Flavell, 1976, p. 232).

This definition emphasizes the executive role of metacognition in the overseeing and regulation of cognitive processes. Executive processes are those responsible for the goal-directed processing of information and selection of action, and for the implementation and monitoring of task-specific cognitive processes. Flavell provides some useful examples of metacognition:

“I am engaging in metacognition if I notice that I am having more trouble learning A than B; if it strikes me that I should double-check C before accepting it as a fact; (...) if I become aware that I am not sure what the experimenter really wants me to do; if I sense I had better make a note of D because I may forget I; if I think to ask someone about E to see if I have it right.” (Flavell, 1976, p. 232).

Following Flavell’s idea of metacognition as active monitoring and consequent regulation and orchestration of cognitive processes to achieve cognitive goals, research in the area investigates different forms of monitoring, regulation, and orchestration, such as checking, planning, selecting, and inferring (Brown, 1978; Scardamalia and Bereiter, 1985); self-interrogation and introspection (Chi et al., 1989; Lieberman, 1979); interpretation of ongoing experience (Flavell and Wellman, 1977; Brown and Palinscar, 1982; Whimbey and Lochhead, 1999); or simply making judgments about what a person knows or does not know about how to accomplish a task (Metcalfe and Shimamura, 1994).

Ann Brown, a respected researcher in the area who has done much work on the cognitive processes of reading comprehension, states:

“Metacognition refers to understanding of knowledge, an understanding that can be reflected in either effective use or overt description of the knowledge in question” (Brown, 1987, p. 65).

Her definition calls attention to an important aspect of metacognition: awareness of one’s own knowledge or understanding of knowledge. It can be said that a learner understands a particular cognitive activity if she can use it appropriately and discuss its use. But there are in fact different degrees of understanding, as it is not unusual for learners to use knowledge effectively without being able to explain how they did so. To account for this variability, the term “metacognitive person” is often used to represent a person who is aware of her own knowledge (as well as its limitations). Two other terms are used in our work: high metacognitive and low metacognitive to distinguish learners who are aware of their knowledge and strategies from those who are less.

2.2.2 A fuzzy concept

Research in metacognition has originated independently in different areas which results in some debate over the exact meaning of the word. Not only is metacognition related to different disciplines (cognitive psychology, developmental psychology, philosophy of mind), it has also been
investigated for widely varying purposes. As a result, metacognition is considered by many a fuzzy concept (Flavell, 1981, p.37). Some people go as far as saying that some of the forms of metacognition which have appeared in the literature are puzzling and mysterious (Brown, 1987, p. 65).

Although much work has been done in almost three decades of research in the area, the complexity and multifaceted aspect of metacognition generates difficulties for its researchers to build testable theories, to design complete experiments, and to find definitive evidence (Garner and Alexander, 1989).

Under the umbrella of the term metacognition there are several different research problems to investigate. This corroborates to this idea of fuzziness. Brown (1987) states that the confusion that follows the use of a single term to a multifaceted problem is the inevitable outcome of mixing metaphors. In fact, currently, one can find in the literature several terms used to describe the same basic phenomenon (e.g. self-regulation, self-management, executive control), or an aspect of that phenomenon (e.g. metamemory, metalearning - metacognitive aspects of memory and learning).

2.2.3 Distinguishing metacognition from cognition

Another primary problem with the concept of metacognition is that it is often difficult to distinguish between what is meta and what is cognitive (Brown, 1987, p. 66). For example, the skill needed to read a text differs from the skill of monitoring one’s understanding of the text. The first is an example of a cognitive skill, the second of a metacognitive skill. The knowledge of computer programming is cognitive, the knowledge that you are better at reading than at implementing software is of metacognitive nature. These examples may seem clear, but not everything is so clear cut. Because of the interchangeability of cognitive and metacognitive functions, a particular activity can be seen as the strategy itself (e.g. looking for the main points in a text one is studying from), or as a monitoring function (which is a metacognitive activity), or even a reflection of the knowledge (also metacognitive) that it is an appropriate strategy to employ in a given situation.

This is in part because research in cognitive skills in general includes different tasks, such as memory tasks, reading text, writing, language acquisition, problem solving, but also performing calculations, measurements, mathematical modelling, drawing, etc. Cognition not only includes the observation and manipulation of objects, entities, reality, but also the processing of information, i.e. of signs like words or figures, often coupled to previously learned skills (Vos, 2001).

Flavell (1979), in his model of metacognition, assumes that metacognition and cognition differ in their content and function, but are similar in their form and quality, i.e., both can be acquired, be forgotten, be correct or incorrect, and metacognition can be expressed in external formulations, with said information being either correct or not, subjective, shared, or validated, just like cognition. Hence, we can differentiate cognition and metacognition using two of their basic characteristics: content and function.

The contents of metacognition are the knowledge, skills, and information about cognition (part of the mental world), while cognition is about things in both the real world and mental images thereof (i.e., objects, persons, events, physical phenomena, signs, etc., skills to handle these entities, and information on the tasks). Thus, one way to differentiate metacognitive thinking from other kinds of thinking, is to consider its source. Douglas Hacker says that “metacognitive
thoughts do not spring from a person’s immediate external reality; rather, their source is tied to the person’s own internal mental representations of that reality, which can include what one knows about that internal representation, how it works, and how one feels about it.” (Hacker, 1998, http://www.psyc.memphis.edu/trg/meta.htm).

Cognition and metacognition differ in function in the following way. The function of cognition is to solve problems, to bring cognitive enterprises to a good end. The function of metacognition is to regulate a person’s cognitive operation in solving a problem or executing a task (Vos, 2001), for example, realizing one does not understand, deliberately increasing one’s concentration to block out environmental distractions, consciously using one’s memories to progress toward understanding, etc. (Hacker, 1998).

Broader reviews about the nature of metacognition and its roots can be found in Brown (1987) and Hacker (1998). In the next sections we will concentrate on studies that focused on the role of metacognition in education, on methods for metacognitive assessment and on metacognitive training in the domain of problem solving and maths.

In sum, different avenues of research emerged, focusing on, either specific subcomponents of metacognition, such as metamemory or self-regulation, or on theoretical perspectives of metacognition, or even on the utility of metacognition training for education and general cognitive development. Metacognition is mostly studied in two main areas of learning - the wide domain of studying from texts (Ann Brown and Ruth Garner, for example, have an extensive body of research on that domain; e.g. Brown and Palinscar (1982); Garner (1987)) and problem-solving (Alan Schoenfeld and Janet Davidson, for example, have made important contributions in this area; e.g. Schoenfeld (1987); Davidson et al. (1994)). This research has been concerned with the latter.

Several models of metacognition were defined in order to make the concepts more understandable and to facilitate further inquiries. In the next section three of them will be presented which have been chosen for their relevance to this work.

2.3 Models of metacognition

A number of models have been proposed which are derived from different conceptualizations of metacognition. Some are more general and provide a theoretical framework for metacognition, such as Flavell’s and Brown’s models of metacognition; others concentrate on specific aspects of metacognition, such as knowledge of the structure of language (Tunmer and Bowey, 1984), memory processes and metamemory (Schneider, 1985), studying from texts (e.g. The Tetrahedral Model for Learning from Texts by Brown et al. (1986)), and metacognitive strategies for self-regulation during reading (Scardamalia and Bereiter, 1985).

Next Flavell’s Model is presented as it lays the foundations of the metacognition theory and, for the first time, attempts to define the components of metacognition and the interactions among these components. Similarly, Brown’s Model is also presented because it makes an important distinction of two different categories of metacognition: knowledge of cognition and regulation of cognition, which is taken into account in the design of our proposed model (our model has

---

1We need to make a distinction between “metacognitive strategy” and “metacognitive skill”. We use the term **metacognitive strategy** for the conscious, deliberate use of a specific method, whereas **metacognitive skill** is used to define a refined strategy which is used selectively, automatically and unconsciously as needed.
specific activities that target each of these aspects of metacognition). Finally, we present Tobias & Everson’s Model as they propose a modular model of metacognition and their model will be used as a basis for this study. These three representative general models are detailed below.

2.3.1 Flavell’s model of cognitive monitoring

In his classic article “Metacognition and Cognitive Monitoring”, Flavell (1979) makes the first attempt to define the components of metacognition by creating a model of cognitive monitoring/regulation. His proposal includes four components: (a) metacognitive knowledge, (b) metacognitive experiences, (c) goals or tasks, (d) actions or strategies. A person’s ability to control a wide variety of cognitive enterprises depends on the actions and interactions among these components (Figure 2.1 shows the relations between them).

![Flavell's model of metacognition](image)

Figure 2.1: Flavell’s model of metacognition.

**Metacognitive knowledge** is one’s acquired world knowledge about cognitive processes, a personal perspective of one’s own cognitive abilities as well as others. The statement “I am good at arithmetic, but Bob knows more words than I do” is an example of metacognitive knowledge. Flavell states that metacognitive knowledge consists primarily of knowledge or beliefs about what factors or variables act and interact in what ways to affect the course and outcome of cognitive enterprises (Flavell, 1979, p. 907). He also identifies three general categories of these factors: the person category, the task category, and the strategy category.

The person category comprises everything that one can come to believe about the nature of oneself and others as a cognitive processor. This category relates to knowledge and beliefs one has about differences between individuals or intraindividual differences (e.g. the realization that one is better at calculating than at memorizing history), or universal properties of cognition (e.g. knowledge or belief that there are different degrees and kinds of understanding).

As the name indicates, the task category has in it the information available and demands of the specific cognitive task the person is engaged at the moment. In that category, we would find the understanding of what are the implications of the way information is presented (e.g. the task is well or poorly organized) and the goals set (e.g. recall the gist or recall the wording of a text) to

the path one will choose to manage the cognitive task in the best way to reach the goal and how likely one is to do it successfully.

In the strategy category, we find knowledge about which strategies are likely to be effective for achieving subgoals or goals in various cognitive tasks. Flavell affirms that it is possible to acquire metacognitive strategies as well as cognitive ones (this affirmation is very important to later research developed in the area of metacognition, specially those related to metacognitive instruction).

Flavell, as we remember, argues that metacognitive knowledge does not differ in form and quality from other knowledge stored in long-term memory. As a consequence, it can either be retrieved as a result of a deliberate and conscious memory search, or it can be activated unintentionally and automatically by retrieval clues in the task situation. The latter situation is the most common. Metacognitive knowledge can be used unconsciously. However, it may also rise to consciousness and provoke what he calls a metacognitive experience.

A metacognitive experience is a cognitive or affective experience that accompanies a cognitive action. In other words, it is the conscious consideration of intellectual experiences that accompany any success or failures in learning or other cognitive enterprise (e.g. having a feeling of confusion after reading a text passage). Flavell affirms that many of these experiences have to do with where one is in a task and what sort of progress one is making or is likely to make. An interesting argument put forth by Flavell is that these experiences can also occur at any time before, after, or during a cognitive enterprise. And he concludes stating that these experiences are more likely to happen in situations that demand careful and highly conscious, reflective thinking, for example, in situations that require previous planning, or where decisions and actions are weighty and risky.

The goals or tasks refer to the actual objectives of a cognitive endeavour, such as reading and understanding a passage for an upcoming quiz, which will trigger the use of metacognitive knowledge and lead to new metacognitive experiences. And finally, actions or strategies refer to the utilization of specific techniques that may assist in achieving those goals (e.g. a metacognitive experience could be remembering that outlining the main ideas of a passage on a previous occasion had helped increase comprehension).

This model is important to define what metacognitive knowledge is and what are the main factors that most likely influence its content and development.

2.3.2 Brown’s model of metacognition

Brown (1987) divides metacognition into two broad categories: (1) knowledge of cognition, as activities that involve conscious reflection on one’s cognitive abilities and activities; and (2) regulation of cognition, as activities regarding self-regulatory mechanisms during an ongoing attempt to learn or solve problems. According to Brown, these two forms of metacognition are closely related, each feeding on the other recursively, although they can be readily distinguishable (see Figure 2.2).

Knowledge about cognition refers to the stable, statable, often fallible, and often late developing information that human thinkers have about their own cognitive processes as it requires that learners step back and consider their own cognitive processes as object of thought and reflection;
traditionally this has been referred to as knowing that (Brown, 1987).

**Regulation of cognition** consists of the activities used to regulate and oversee learning. These processes include planning activities (predicting outcomes, scheduling strategies, and various forms of vicarious trial and error, etc) prior to undertaking a problem; monitoring activities (monitoring, testing, revising, and re-scheduling one’s strategies for learning) during learning; and checking outcomes (evaluating the outcome of any strategic actions against criteria of efficiency and effectiveness). It has been assumed that these activities are relatively unstable (although they are ubiquitously employed by adults on simple problems), not necessary statable (knowing how to do something does not necessarily mean that the activities can be brought to the level of conscious awareness and reported on to others), and relatively age independent (i.e., task and situation dependent).

Additionally, Brown introduced the concept of “autopilot state”, arguing that expert learners (e.g. readers) monitor their comprehension and retention and evaluate their own progress in the light of the purposes for which they are learning to the extent that these activities become automatic and learners proceed as if in “automatic pilot”. This concept tries to explain why metacognitive learners (i.e. those who apply metacognitive knowledge and skills in learning situations) sometimes are not conscious of their strategies and can not describe their metacognitive knowledge.

This model emphasizes the **executive processes**, stressing the importance of the control that people bring or fail to bring to cognitive endeavours. Moreover, Brown points to important characteristics of regulation of cognition, that have to be taken into account for those interested in the applications of these concepts into instructional research.
2.3.3 Tobias & Everson’s hierarchical model

Similar to other researchers (Pintrich et al., 2000, for example) Tobias and Everson perceive metacognition as a compound of skills and knowledge - knowledge of cognition, monitoring of one’s cognitive and learning processes, and control of those processes. However, they organize these components into a hierarchical model, where the metacognitive skill of knowledge monitoring is a pre-requisite for activating other metacognitive skills as illustrated in Figure 2.3.

![Figure 2.3: Tobias & Everson’s hierarchical model of metacognition.](image)

They define **knowledge monitoring** (KM) as the ability of knowing what you know and knowing what you don’t know. In their recent research report they affirm:

“We believe that monitoring of prior learning is a fundamental or prerequisite metacognitive process[...]. If students cannot differentiate accurately between what they know and do not know, they can hardly be expected to engage in advanced metacognitive activities such as evaluating their learning realistically, or making plans for effective control of that learning. Learners who accurately differentiate between what has been learned previously and what they have yet to learn are better able to focus attention and other cognitive resources on the material to be learned.” (Tobias and Everson, 2002, p. 1)

Therefore, those who accurately distinguish between what they have already learned and what is yet to be acquired have an important advantage, since they can refrain from studying material that has already been mastered, or merely review it briefly. They assert that these students devote most of their time and energies to new, unfamiliar materials. In contrast, they argue that those students with less effective knowledge monitoring processes are likely to allocate their time and resources less effectively and spend valuable time studying what they already know at the expense of unfamiliar material and, consequently, have greater difficulty mastering new subjects (Tobias et al., 1999).

Tobias and Everson have investigated largely the monitoring aspect of metacognition, based on the assumption that accurate monitoring is crucial in learning and training contexts where students have to master a great deal of new knowledge (Tobias et al., 1999). They have performed a series of empirical studies to investigate this aspect of metacognition and its relationship to learning from instruction in different domains, focusing on issues such as domain specificity of knowledge monitoring, measurement concerns, and the relationship of knowledge monitoring to academic ability. They have developed a metacognitive assessment instrument, which is detailed later in this chapter.
In this research we have adopted this model of metacognition, building our computational framework based on the vision that promoting conscious development of knowledge monitoring, would lead to increasing of attention focus and appropriate allocation of cognitive resources and would, consequently, improve the other components of metacognition, that are placed on the top of Tobias and Everson’s pyramid. This model is particularly suitable for this research because it enables us to focus on specific metacognitive skills relevant for problem solving and provides an assessment instrument for one of these skills (i.e., knowledge monitoring).

2.4 Assessment of metacognition: methods and instruments

Garner and Alexander (1989) stressed the relevance of empirical research on the measurement of metacognition, suggesting that the following questions should be addressed: “How can we measure knowledge about knowledge more accurately?”; “How can we measure the effects of strategy training?”.

Many researchers have attempted to answer these questions, designing instruments and methods to measure metacognition as a whole or components of it; those were then tested with learners in different domains. These methods range from self-questionnaires, where learners themselves rate their metacognitive skills and knowledge, to interviews or verbal-reports, in which the learners recall what they did and what they thought during a learning experience. There is a broad consensus among researchers that all such methods are fallible, not least because measuring metacognition is a very difficult task. Thus, many people have suggested the use of multiple methods that do not share the same source of error to provide a more reliable picture of the phenomena under investigation (Ericsson and Simon, 1980; Garner, 1988; Meichenbaum et al., 1985; Schoenfeld, 1985).

For example, performance measurements can be combined with verbal reports thus combining observable nonverbal data with verbal data. Schoenfeld (1985) argues that any methodology, will illuminate some aspects of behaviour but distort others. Environmental factors, such as the subjects tendency to perform in front of a microphone, their beliefs about the research requirements or about the discipline can affect the results. Thus, he suggests that verbal data should be compared and contrasted with data from other sources.

Table 2.1 presents a summary of the most common techniques that have been employed to assess metacognition. It points out to advantages and drawbacks of these techniques. In addition to these methods, some investigators also devised performance-based measures, using extensive verbal interviews (Artzt and Armour-Thomas, 1992) or questionnaires (Tobias and Everson, 2002) applied prior to a learning task. A comparison between what is said or answered and the performance on task is made in order to make the metacognitive assessment. An extensive review of the types of methods for assessing metacognition and the instruments developed can be found in Meichenbaum et al. (1985), Baker and Cerro (2000) and Pintrich et al. (2000).

In this research we have experimented with some instruments suggested in the literature. We have also created our own methods and adapted some of the existing ones to our purposes. The following sections will discuss the instruments we used, their limitations and usefulness.
Table 2.1: Common methods applied in the assessment of metacognition.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Advantages</th>
<th>Sources of Error and Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>concurrent think-aloud</td>
<td>learner says outloud everything she thinks and everything that occurs to her while performing a task.</td>
<td>get rich data about processes that are “invisible” to other methods.</td>
<td>automated processes remain inaccessible; young children do not sustain verbalizations; reporting may be disruptive of processing; produces masses of data that needs careful analysis.</td>
</tr>
<tr>
<td>post-performance interviews</td>
<td>interview in which learner has to recall what she did and thought during a learning experience.</td>
<td>provides data from responses to specific, direct probes.</td>
<td>lack of awareness of processing for those processes that are automated; failure to remember cognitive events given the interval between processing and reporting; lack of verbal fluency (specially for children) and variation in adult-child use of language; investigator needs to be attentive for not cuing particular responses.</td>
</tr>
<tr>
<td>cross-age tutoring</td>
<td>ask subjects to tutor younger children to solve a problem to observe which strategies and behaviours they will encourage and teach.</td>
<td>non-verbal data; avoid subject guessing what the investigator wants to hear and answering accordingly.</td>
<td>useful in the investigation of specific strategies (e.g. awareness of usefulness of text reinspection strategy).</td>
</tr>
<tr>
<td>self-report inventory</td>
<td>self-questionnaire using Likert or continuous scale with multiple items.</td>
<td>structured and convenient: easy to apply and score.</td>
<td>answers may be given to please the investigator/teacher; difficult to answer about at least partially automated processes.</td>
</tr>
</tbody>
</table>

2.4.1 The MAI inventory

The Metacognitive Awareness Inventory (MAI) is a 52-item self-report with a 100 mm wide rating scale following each item (Schraw and Dennison, 1994). The questionnaire includes multiple items regarding knowledge and regulation of cognition, subdivided into eight component processes (Schraw and Dennison, 1994). Other authors propose similar questionnaires with variations in the quantity or wording of the items (O’Neil and Abedi, 1996; Goos et al., 2000). The MAI has well devised statements. However we observed the following problems: (a) even being elaborated as a domain-independent questionnaire, some of the items can be answered differently, depending on the domain the learner has in mind. For instance, the item “I summarise what I’ve learned after I finish”, may be rated high if the learner thinks of her behaviour in literature or history, or low if she refers to her pattern in maths; (b) similar to other self-report methods, subjects sometimes provide responses that they suppose will please the investigator (e.g. “I know what the teacher expects me to learn”) or, in other circumstances, will hide their weaknesses, because they are reluctant to admit exerting little effort on schoolwork (e.g. “I ask myself if I learned as much as I could have once finished a task”) ; (c) some of the items refer to processes that subjects may not be aware of or never have thought about, either because they are automatic or because they never used them (e.g. “I use my intellectual strengths to compensate for my weaknesses”, “I have control over how well I learn”, etc.).

We have used the MAI inventory in a preliminary study to evaluate its usefulness as a pre-post test instrument to measure metacognitive knowledge and regulation in our main experiment.
We found a low correlation between subjects’ responses and their demonstrated performance, and between the stated use of metacognitive strategies and the actual use of these strategies in a posterior learning task. Moreover, the MAI inventory measures different components of metacognition (e.g. procedural, declarative, and conditional knowledge of cognition; and aspects of regulation of cognition, like planning, monitoring, debugging, etc.), and it would be necessary to reduce the MAI to the items that focus on those aspects of metacognition which are strongly represented in our Reflection Assistant model.

2.4.2 The Knowledge Monitoring Assessment instrument

Tobias and Everson (1996) have created an assessment instrument that focuses on the knowledge monitoring component of metacognition. They wanted a measurement method that could be used across domains and whose results would also be consistent across domains. Another desired feature was for the measure to be objectively scored. The Knowledge Monitoring Assessment instrument has indeed been shown to be quite general in its applicability. It has been used successfully with students ranging from elementary to college students, with normal and learning-disabled students, across racially diverse samples, with economically advantaged samples, and across multiple disciplines (e.g. English, maths, psychology, and oceanography courses). In total 23 studies were performed to date (Tobias and Everson, 2002) which suggest that this is a useful approach to measure metacognitive knowledge monitoring.

How the Knowledge Monitoring Assessment works

The Knowledge Monitoring Assessment technique evaluates a student’s knowledge monitoring ability by first asking her whether she knows something or not and later challenging her to a task or a question which can prove whether she was right in her assessment. For instance a student can be asked whether she knows the meaning of a word and later be prompted for a definition of that word. In a task-oriented scenario, she can be presented with a maths problem, asked whether she thinks she can solve that problem and, at some later point in time, be tested by asking her to solve the problem in question. The student is presented with a battery of such questions and after she has answered them all, she is submitted to one challenge for each of the questions.

By collecting a significant number of elementary assessments for the same student, the Knowledge Monitoring Assessment produces a statistical profile of a student’s awareness of her own knowledge. For each assessment the student is put through, one of the following outcomes must take place:

- student stated she knew and indeed performed accordingly: (a) [+ +]
- student stated she did not know but succeeded when challenged: (b) [- +]
- student stated she knew but failed when challenged: (c) [+ -]
- student stated she did not know and indeed performed accordingly: (d) [- -]

For the purposes of the Knowledge Monitoring Assessment, the first two cases are equivalent since they both show successful knowledge monitoring. The last two represent cases of failed monitoring and must be treated similarly to each other but differently from the first two. For the
Table 2.2: Possible values in the Knowledge Monitoring Assessment Instrument. Types of outcome and their respective counts

<table>
<thead>
<tr>
<th>Actual Performance</th>
<th>Student Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Know</td>
<td>(a) [+ +]</td>
</tr>
<tr>
<td>Do not know</td>
<td>(b) [- +]</td>
</tr>
<tr>
<td>Know</td>
<td>(c) [+ -]</td>
</tr>
<tr>
<td>Do not know</td>
<td>(d) [- -]</td>
</tr>
</tbody>
</table>

same student, the number of times each of these four scenarios happens is counted. The resulting counts for each of the four cases are labelled (a), (b), (c) and (d) as shown in Table 2.2.

The Knowledge Monitoring Assessment score is then obtained using the simple formula:

\[
\frac{(a + d) - (b + c)}{a + b + c + d}
\]

This number is always comprised between -1 and 1. A score of 1 is obtained when b and c are equal to 0, i.e. when the student never fails in her knowledge assessment. It doesn’t matter whether she correctly predicts success or failure as a and d have the same treatment in the formula. Conversely, a score of -1 is obtained for a student who consistently fails in her knowledge assessment. A null score will be obtained when the student fails in her assessment as often as she succeeds. This would be the statistical expectation if she answered randomly when prompted about her knowledge.

**The Knowledge Monitoring Assessment against other measures**

In a comparison of measures of general metacognition, metamemory, and metacomprehension for their psychometric properties and their appropriateness for use by educators (Osborne, 1998), the Knowledge Monitoring Assessment comes at the top of the list. Osborne (1998) concludes that the Knowledge Monitoring Assessment appears to be a naturalistic and robust measure of knowledge monitoring that has good reliability and excellent internal validity.

Likewise, Tobias and Everson (2002) performed studies that investigated the relationship between the Knowledge Monitoring Assessment and other self-report measures of metacognition. It was compared to the “Learning and study strategies inventory - LASSI” (Weinstein et al., 1987), the “Motivated Strategies for Learning Questionnaire - MSLQ” (Pintrich et al., 1991), and the “Metacognitive Awareness Inventory - MAI” (Schraw and Dennison, 1994). One advantage observed in the Knowledge Monitoring Assessment over these self-report instruments is that is it less likely to be affected by the difficulties of abstract recall involved in responding to metacognitive questionnaires. This happens because it does not ask students to report on cognitive processes used while performing a task. For example, research suggested that the Knowledge Monitoring Assessment, compared to self-report scales of test anxiety, is less susceptible to students’ tendency to present themselves in a favourable light (Gerrity and Tobias (1996), described in Tobias and Everson (2002)). Moreover, low-to-moderate relationships were found between those instruments and the Knowledge Monitoring Assessment, showing that those self-reports tended to correlate highly with one another, indicating that they measure similar characteristics using similar methods. Tobias and Everson (2002) noted that in most of the studies the correlations among the Knowledge Monitoring Assessments and measures of scholastic aptitude and learning tended to be higher.
than those for the self-report instruments (p. 17). Their conclusion is that the Knowledge Monitoring Assessment instrument assesses components of metacognition that are more highly related to classroom learning, than those measured by self-report. Another recent review of research of existing metacognitive assessment instruments (Pintrich et al., 2000) confirms that the scores on this instrument have the overall highest relationship with learning outcomes.

Based on these studies and indications, the Knowledge Monitoring Assessment instrument was adapted and used in this research. Since our research pays great attention to the knowledge monitoring component of metacognition and given that the instrument can be scored automatically, it seems highly suitable to be incorporated in the Reflection Assistant Model. Chapter 4 details the adaptations made in order to incorporate it in our computational model.

2.5 Metacognition and Instruction

As early as 1979, Flavell claimed (though with little experimental evidence) that metacognition played an important role in oral communication of information, oral comprehension, reading comprehension, writing, language acquisition, attention, memory, problem solving, social cognition, and various types of self-control and self-instruction (Flavell, 1979, p. 906). He added:

“I think that increasing the quantity and quality of children’s metacognitive knowledge and monitoring skills through systematic training may be feasible as well as desirable” (Flavell, 1979, p. 910)

Since then research on metacognition sought to answer questions, such as: “How does metacognition develop? Can instruction of metacognition make a difference in learning in general? Does direct instruction on metacognition lead to better regulation of one’s cognitive activities?” Empirical research has shown that metacognition has the potential to increase the meaningfulness of students’ learning in different domains (e.g. Brown and Palinscar (1982); Biggs (1986); Paris and Winograd (1990); Pressley and Ghatala (1990); Hartman (2001a)). For example, in an study comparing adults, Corkill and Koshida (1993) reported that metacognitive awareness measured during on-line interviews was related to monitoring accuracy. Ganz and Ganz (1990) also noted that metacognition has the potential to empower students to take charge of their own learning, increase perceived efficacy, and decrease the potential for learned helplessness, all of which are desirable educational goals. They added that developing one’s metacognitive skills might be the route through which affective and attitudinal variables are affected, which suggests that metacognition is a key factor for educational endeavours.

As a consequence of these studies, educational researchers have investigated ways of teaching students to reflect on their knowledge and to use their cognitive resources strategically through metacognitive control. A good example in this direction is the successful reciprocal teaching approach, developed by Palincsar and Brown (1984). The reciprocal teaching approach is a co-operative learning method for improving reading comprehension specifically designed to develop four metacognitive reading skills: questioning, clarifying, summarizing and predicting. In reciprocal teaching a tutor and a group of students (it can also be used in individual tutoring situations) take turns leading discussions about specific segments of text using the metacognitive reading strategies (questioning, clarifying, etc.). The teaching techniques involved in this approach are:
• model and explain
• practice with feedback
• provide students with temporary support (scaffolding)
• take turns leading text dialogues.

The combination of these techniques leads students to, eventually, apply the metacognitive reading strategies on their own, performing self-regulation on their reading comprehension.

Some researchers defend the idea that the curriculum should be oriented towards the development of metacognitive skills via programmes that include the explicit teaching of metacognition (Hartman, 2001a). Such programmes would involve incorporating a range of learning experiences which encourage reflection on learning goals and strategies in relation to course content. The encouragement of student questioning of a reflective kind, the sharing amongst peers of problem solving and study strategies, the use of reflective diaries and concept mapping are amongst the techniques that could be appropriate in such programmes. Hope Hartman advocates that teachers should teach metacognitively. She defines “teaching metacognitively” as teaching with and for metacognition. Teaching with metacognition means that teachers will think about their own thinking regarding instructional goals, teaching strategies, sequence, materials, students’ characteristics and needs, and other issues related to curriculum, instruction and assessment before, during and after lessons in order to maximize their instructional effectiveness. On the other hand, teaching for metacognition means that teachers will think about how their instruction will activate and develop their students’ metacognition (Hartman, 2001b). She also devised some material and activities for training college teachers in several different subject areas and for undergraduate educational psychology students who were preparing to become teachers. The techniques included, among others: (a) advance organizers to provide an overview of the lesson plan; (b) graphic organizers to visually represent course concepts and their relationships; (c) pretests; (d) student journals; and (e) teaching strategy projects. Some instructional techniques and methods for metacognitive development are described below in Section 2.7.

All examples so far propose to foster metacognition in close connection with a domain. At the other end of the spectrum, the idea that metacognition is a domain-independent ability led to the creation of domain-independent approaches such as the Cognitive Strategy Instruction programs. Cognitive Strategy Instruction (CSI) is an instructional approach which emphasizes the development of thinking skills and processes as a means to enhance learning (Livingston, 1997). The objective of CSI is to enable all students to become more strategic, self-reliant, flexible, and productive in their learning endeavours in general (Scheid, 1993). CSI is based on the assumption that there are general identifiable cognitive strategies, previously believed to be utilized by only the best and the brightest students, which can be taught to most students. However, the learning results of these programs are questionable. Many researchers reject the idea that this type of skill can be taught in abstraction of a domain (Laurillard, 1993; Derry, 1992; Garner, 1987; Brown, 1985). For example, Laurillard (1993) defends that teaching must not simply impart decontextualised knowledge, but must emulate the success of everyday learning by contextualising, situating knowledge in real world activity. We share the view that metacognition should be trained and experienced in an embedded way, integrated with a meaningful domain-specific learning activity. As
such, our approach is to promote the development of metacognition in conjunction with problem solving activities in the domain of algebra.

### 2.5.1 Individual differences in metacognitive development

One important issue in metacognition and instruction is the ability of learners to develop their metacognition. Research shows that effective learners are those who are aware of their strengths and limitations and find ways to remedy the latter (Bransford et al. (1999), Chapters 3, 4 and 7).

It is a consensus that learning is effective when it is active, meaningful, retained over time, and transfers to a variety of contexts. However, it is not unusual to find students following instructions or performing tasks without wondering what they are doing or why they are doing it. Those students rarely question themselves about their own learning strategies or evaluate the efficiency of their performance. Some of them have virtually no idea of what to do when confronted with a problem and are unable to explain their strategies for decision making. An extremely important but often neglected aspect of learning is that students often have the requisite knowledge and skills for performing complex tasks but fail to use them: the skills remain inert. Sometimes it is because they are not motivated or confident to apply them, and sometimes because they do not recognize that the situation calls for the use of those skills (Hartman, 2001a). That is, learners may have declarative and procedural knowledge, but not the contextual or conditional knowledge needed for its application and transfer.

Research shows that these students have problems: a) determining the difficulty of a task; b) monitoring their comprehension effectively, i.e. not recognizing when they do not fully understand something (e.g. task directions, information in textbooks); c) planning ahead (e.g. what they need to do and how long each part should take); d) monitoring the success of their performance or determining when they have studied enough to master the material to be learned; e) using all the relevant information; f) using a systematic step-by-step approach; and g) selecting and employing appropriate representations (Hartman, 2001a). All these abilities are metacognitive in nature.

Most of us engage in metacognitive processes when confronted with effortful cognitive tasks, yet some people are more metacognitive than others. Recent research indicates that the more metacognitive one is, the more strategic and successful one is likely to be in cognitive endeavours (Borkowski et al., 1987; Garner and Alexander, 1989; Pressley and Ghatala, 1990). One explanation, suggested by Schraw and Dennison (1994), is that metacognitive awareness allows individuals to plan, sequence, and monitor their learning in a way that directly improves performance. Borkowski et al. (1987) argue that compared to poor learners, good learners have a larger repertoire of strategies, are more flexible in their approaches to problems, have a larger database of knowledge concerning the circumstances that make different strategies appropriate, appreciate the relationship between effort and performance, and do more strategic self-monitoring and regulation of strategy use to ensure that activities are carried out in the appropriate sequence.

Zimmerman (1995) refines the arguments above pointing out that it is not enough for students to have metacognitive knowledge (or skills); they must also regulate its use when confronted with stress, competing attractions and fatigue. Accurate monitoring of new learning enables students with effective metacognitive strategies to concentrate on new content and adjust their learning goals. Zimmerman (2000) characterizes the metacognitive processes of self-regulated
learners in terms of planning, setting goals, organizing, self-monitoring and self-evaluating at various times during the learning process. So, context-dependent motivational issues, such as effort, self-efficacy, persistence and task choice are also important determinants of self-regulation. Metacognition is necessary, but not sufficient, for academic success. The most important point is that through practice of self-regulation, students can develop voluntary control over their own learning. Teachers can enhance students’ awareness and control over learning by teaching them to reflect on how they think, learn, remember and perform academic tasks at all stages before, during and after task execution (Hartman, 2001a).

2.5.2 The importance of the knowledge monitoring skill
Similarly to Tobias and Everson, Lin (2001) believes that knowledge monitoring is a valuable skill that should be fostered in students. She argues that, by identifying what they know and do not know, students can focus their attention and resources more appropriately. Zimmerman (1998) adds that, by knowing what they already know, students become aware of the potential knowledge and skills that they can bring to bear, which boosts their confidence in their own learning.

Tobias and Everson believe that knowledge monitoring is central to learning from instruction in varied domains. They have conducted 23 experiments that support the importance of accurate monitoring of prior knowledge in students’ strategic behaviour during learning. For instance, one of their experiments demonstrated that students who monitor knowledge well generally seek more help on the vocabulary they think is unknown than those not so good at monitoring.

These findings were substantially confirmed in the domain of mathematics. Another experiment found that accurate knowledge monitors required less external feedback regarding the accuracy of their responses. They argue that this happens because they were able to supply their own internal feedback (Tobias and Everson, 2002). Another interesting observation is the positive correlation between knowledge monitoring and academic achievement. They argue that much of the research conducted to date supports this correlation, including their own research on the assessment of knowledge monitoring. Their earlier research, for example, indicated that knowledge monitoring ability was related to academic achievement in college (Tobias and Everson, 1998).

Moreover, the relationship between knowledge monitoring and academic achievement was documented in diverse student populations, including elementary school students, students attending academically oriented high schools, vocational high school students, college freshmen, and those attending college for some time (Tobias and Everson, 2002).

2.6 The Role of Reflection in Metacognition
Before we discuss the importance of reflection in metacognition, we explain what we mean by this term and distinguish between reflection and self-reflection.

2.6.1 What is Reflection and Self-reflection
The Oxford English Dictionary defines to reflect as “to think deeply or carefully about”; semantically it is therefore quite close to the verb to think and, as with thinking, the object of the reflection can be anything. But reflect comes from the Latin reflectere meaning “to bend back”; this origin shows itself in another meaning of the verb illustrated in the sentence “the mirror reflects the
light”. Reflexive, another derivative of reflectere also displays this etymology clearly. By merging those two meanings, we end up with the concept of thinking reflexively or thinking about oneself. Although this meaning of reflection is not reported in most dictionaries, it is a useful one to have for researchers who are interested in self-inspection. Understandably quite a few researchers have adopted this novel and narrower definition of the verb reflect, implying through its use that the object of reflection is the thinker herself. The same transformation of the meaning has also been applied to its derivatives such as reflection and reflective. For example, for Boud et al. (1985) reflection is an activity in which people “recapture their experience, think about it, mull it over and evaluate it” (Ibid, p. 19).

The concept of “reflective thinking” was introduced by John Dewey in 1910 in his book “How We Think” to represent “active, persistent, and careful consideration of any belief or supposed form of knowledge in the light of the grounds that support it and the further conclusion to which it tends” (Dewey, 1933, p.9). Although it is not strictly thinking about oneself, reflection here involves the learner as the object of reflection.

As a follower of Dewey, Donald Schön in his book The Reflective Practitioner states that the reflective thinker is aware of her own knowledge and lack of knowledge and recognizes that there may not be only a single correct solution to a given problem or interpretation of a situation. Schön (1987) creates the terms “reflection-in-action” and “reflecting-on-action”; the former refers to a practitioner’s thinking about a given phenomenon before him, and on the prior understandings which have been implicit in her behaviour and whilst she carries out an experiment which serves to generate both a new understanding of the phenomenon and a change in the situation. The latter enables practitioners to spend time exploring why they acted as they did, and in so doing develop sets of questions and ideas about their activities and practice. We notice that the notion of reflection-on-action implies a reflexive form of thinking.

Those in the area who have stuck with the common definition have had to use more cumbersome terms like self-reflect, self-reflective and self-reflection to refer to reflection where the subject is also the object. For example, the literature concerned with self-regulation and self-regulated learning (e.g. Zimmerman and Schunk (1998)) uses the term self-reflection to define the process that is part of the cyclic self-regulatory process. They state that self-regulation is composed of forethought, performance and self-reflection.

In this thesis we are mostly interested in a subject’s reflection about their own knowledge and learning processes. We decided to stick to the common definition for reflection and indicate its reflexive nature explicitly when needed.

2.6.2 Self-reflection: an essential part of metacognition

Self-reflection is clearly related to metacognition. It is not uncommon in many domains of knowledge that the student have to organize a range of information into a coherent knowledge structure, to analyse situations, generate hypotheses and decide paths for solutions. In this context, it is important that she can study and explore her own learning efforts. Self-reflection provides not only a better understanding of what she knows, but also a way of improving metacognitive strategies, because the learner can examine how she performed a specific learning task. For example, when a learner reflects on an action she has just performed, she is consciously revisiting the information.
Along with the advantage of re-reading, if the student is engaged in an activity that requires her to restructure or explain her thinking, it is expected to result in better learning.

Thus, providing self-reflective activities in a learning environment has the power to enhance the learning benefit of the exercises; it gives the opportunity to review previous actions and decisions before proceeding to a next stage. Consequently it enables students to make more educated decisions later (Goodman et al., 1998).

Self-reflection and self-regulatory skills
Zimmerman (2000) argues that self-reflection has a central role in achieving self-regulation in learning. In his view, it is divided into two components: self-judgement and self-reaction, where self-judgement involves evaluating one’s performance and attributing causal significance to the results, and self-reaction includes satisfaction with one’s performance and adaptive-defensive inferences as conclusions about how one needs to alter her self-regulatory approach during subsequent efforts to learn and perform.

Therefore, researchers in this area believe that developing a self-reflective behaviour is necessary for a student to become a self-regulated learner throughout school years and also as a life-long learner (Zimmerman and Schunk, 1998; Zimmerman, 1998). Schön (1987) asserts that the reflective thinker relies on all available resources to find relevant needed information and opinions in order to come to a personal understanding of a situation, knowing that this understanding may change, as she gains more information and insight into the matter. However, reflexive thinking as the foundation of metacognitive awareness is a complex practice that only begins with one being aware of oneself’s cognitive strategies. Possessing good metacognitive skills consists of more than writing down one’s thoughts on how a process or project is going; it is a dynamic process that occurs while individuals are engaged in activity.

In our research we argue that developing a self-reflective behaviour is a precondition to promote students’ metacognitive awareness and self-monitoring skills, and therefore, it should be part of any instructional program or computer-based environment that intend to promote the development of metacognition. The main hypothesis of this research is that providing reflection on one’s own knowledge monitoring and other self-regulatory skills embedded in problem-solving learning activities can enhance both students’ awareness of their own abilities, and adequate selection of strategies; and, possibly, students’ performance on learning tasks.

2.7 The design of instructional metacognitive activities
While there are several approaches to metacognitive instruction, the most effective ones involve a mixture of theory and practice. The learner must be given some knowledge of cognitive processes and strategies (that will be used as metacognitive knowledge), as well as opportunities to practice both cognitive and metacognitive strategies; evaluation of the outcome of their efforts is also important for the development of metacognitive regulation (Brown, 1987; White et al., 1999). Simply providing knowledge without experience or vice versa does not seem to be sufficient for the development of metacognitive control.

Hence, it is necessary to design metacognitive activities that can be embedded into instructional contexts. In this sense, Lin (2001) suggests that the design of such metacognitive activi-
ties should focus on both cognitive and social aspects of student development, including strategy training and creation of a supportive social environment for the teaching of two kinds of content: knowledge about a specific domain and knowledge about the self-as-learner. She also affirms that this balanced approach is a theoretical and practical challenge for teachers and researchers.

Researchers have identified strategies that teachers can use to promote metacognition in the classroom. An interesting set of activities and tasks that can be adapted and used by teachers in different situations is proposed by Angelo and Cross (1993). It is worth noting that sometimes students apply one or more techniques that are ineffective. Hence it is important to evaluate students’ metacognitive abilities and target instruction to the development of more effective and adequate general learning strategies. We provide below a list of teaching techniques and self-directed strategies that have been commonly applied in both experimental studies and in the classroom.

2.7.1 Reflective questions and reflective prompts
Reflective questions and reflective prompts are simple ways employed by teachers and researchers to promote discussion that begins with revision of the details of the learning experience and moves toward critical thinking and creation of an action plan (Scardamalia and Bereiter, 1985). This can encourage students to reflect on strategies that they use to perform a learning task (such as solving a problem) and explain their reasons for using those strategies.

There is a difference between questions and prompts. Questions are of a more general nature, serving as a way for triggering broad metacognitive monitoring. Examples of questions are: “Now what?” or “So what?”. They may help the student to reflect on the next step and make links to the previous tasks done.

On the other hand, reflective prompts (also called metacognitive prompts) are more focused questions that provide a more directive help on specific aspects of the learning processes. These prompts aim at guiding coherent understanding of the domain tasks at hand and may lead to extensive inference generation (Lin, 2001).

To be effective, the prompts should use open-ended questions. For example, prompts like “Should your goals be reformed?” do not trigger as deep a reflection as “What aspects of your goal setting would you change?”. The prompts should ask for specifics and examples, even if it is the case that the student is not going to write down her reflections (e.g. “What is a new example of ...?”). Another prompting technique consists in paraphrasing and summarizing what the student says when she asks for help; for example: “So what you are concerned about is how you can monitor your problem solving attempt?”. Finally, prompts can also redirect questions to learner, like “Note taking is not occurring in your study from text, should that be the goal of this stage?”. Prompting has also been used to stimulate self-explanation for metacognitive development- eliciting learners’ explanations and justifications through prompting can help them to draw conclusions and make inferences that can lead to increased comprehension (Chi et al., 1989).

Lin and Lehman (1999) conducted a study to see whether different instructional prompts could help students reflect on their understanding of variable control in a biology-based computer environment. Results of the study indicated that students who received metacognitive prompts outperformed the other groups on a far-transfer problem. Similar results have been obtained in different
domains such as maths and science and different tasks, such as learning from expository text (Chi et al., 1989).

The difficulty is to detect the adequate moment to interrupt the student to prompt her to reflect on what she is doing or why she is doing it. Good teachers know when to step in and ask appropriate questions and when to stand back and let the learner figure things out for herself.

2.7.2 Metacognitive scaffolding

Scaffolding means providing support to students to bridge the gap between what they can do on their own and what they can do with guidance from others (Hartman, 2001a). Scaffolding may take the format of models, cues, prompts, hints, partial solutions, etc. The main characteristic of scaffolds is that they have to be regulated according to the amount of help the learner needs, and eventually the help should be not necessary anymore.

Metacognitive scaffolds support the underlying processes associated with individual learning management, providing guidance in how to think during learning. It might remind learners to reflect on the goal(s) or prompt them to relate the use of a given resource or tool to the completion of a task at hand. The scaffolding is intended to serve as an external model of knowledge monitoring behaviour until it is internalized. Therefore, the goal of metacognitive scaffolding is for students to become independent, self-regulating thinkers who are more self-sufficient and less teacher-dependent. It is an especially effective teaching approach for developing higher level cognitive strategies (Hartman, 2001a).

There are two types of metacognitive scaffolding: it can be either domain-specific or more generic. When the problem context is known, scaffolding can emphasize specific ways to think about the problem. In contrast, generic scaffolding focuses on the processes of creating models. This includes finding ways to link models with prior knowledge and experience, linking representational models to current understanding, and enabling learners to manipulate ideas through modelling tools (Hartman, 2001a).

2.7.3 Modelling

Providing models of metacognition in everyday-life and/or school is an important strategy for developing metacognitive knowledge and skills. Teachers can think out loud to externalize their thought processes, serving as an “expert model”, so students can hear effective ways of using metacognitive knowledge and skills. Modelling is often a component of scaffolding. Peer modelling is another possibility. Lin (2001) illustrates this approach with the following example: a student observe a peer engage in effective problem identification and conceptualization of principles for problem solving and by observing their peers, the student may begin to think that she also can be creative and an effective problem solver.

2.7.4 Self-questioning

Self-questioning strategies are effective ways of promoting self-directed learners. Research on self-questioning shows that questions created by the student are much more effective than questions given to the student by someone else. Self-questions such as “Have I left out anything
important?” can help a student self-direct in identifying the omission of important points or examples. The more students practice generating and using self-questions in diverse situations the more likely they are to develop the habit of self-questioning so that it becomes a skill, which is used automatically and unconsciously as the situation requires. It is important to regularly have students adapt their self-questions to the needs of the specific subject and task. Self-questioning can guide the learner’s performance before, during, and after task performance; it can improve self-awareness and control over thinking and thereby improve performance; self-questioning can improve long-term retention of knowledge and skills; it can improve the ability to apply and transfer knowledge and skills students learn; and finally, it can improve attitudes and motivation as a result of improved performance (Hartman, 2001a).

2.7.5 Thinking aloud and Self-explanations

Thinking aloud is a technique of externalizing one’s thought processes as one is engaged in a task that requires thinking. The thinker says out loud all of the thoughts and feelings that occur when performing a task (e.g. solving a problem, answering a question, conducting an experiment, reading through textbook notes, etc.). It is a method that can be used either by the teacher or tutor, or by two students working together, or by a student working alone. Teachers can use the think-aloud method to serve as expert models showing students how to use metacognitive knowledge and strategies when working on a variety of tasks; for example, to let students see and hear how they plan, monitor, and evaluate their work. When the thinker-talker is the subject-matter expert, the process allows the expert to model their own thinking for students. This modelling shows how to think about the material (knowledge, skills, procedures, etc.). It lets students hear what goes on in an expert’s head when a text is read, a homework assignment is attacked, study for a test is planned, an essay is written, an error is found, or a problem is solved. Also, when modelling academic performance, it is important to intentionally make occasional mistakes, so that students can observe and become aware of them and also of strategies for recovering from them and self-correcting (Hartman, 2001a). Meichenbaum and Biemiller (1998) state that think-aloud modelling may be in the form of self-questions (e.g. “Did I carefully check my work?”) or self-instructional directive statements (e.g. “That’s not what I expected. I’ll have to retrace my path”). They recommend that teachers use think-aloud modelling for showing students how to: summarise, access prior knowledge, self-monitor, obtain help, and self-reinforce. This modelling should involve communicating with students so that the lesson is an interactive dialogue instead of a monologue, and modelling should be gradually phased out as student competence and responsibility increase.

Self-explanation is the process of clarifying and making more complete to oneself the content of an exercise, a text, an example, etc. Several studies in cognitive science point that students who spontaneously self-explain when they study learn more (Chi et al., 1989). Moreover, self-explanations are usually more effective than explanations provided by others, because they require students to actively elaborate their existing knowledge. Besides that, the student naturally addresses her specific problems in understanding the content when self-explaining, what leads to a more constructive learning (Chi, 2000). However, studies show that most students do not spontaneously engage in self-explanation and often need guidance to do it (Bielaczyc et al., 1995) or need just to be prompted to do it (Chi et al., 1989).
2.7.6 Self-assessment

Students who observe and evaluate their performance accurately may react appropriately by keeping and/or changing their study strategies to achieve the goal of, for example, maximizing their grade in a course or on a test. Thus, it is desirable to engage students in activities that will help them to assess themselves and to explain explicitly what they know and do not know.

2.7.7 Graphic organizers

Graphic representations can be used to understand text and to solve a variety of problems. Graphic organizer techniques can help students analyze text and see how it is structured. Some graphic organizers that can be used to understand text are flow charts, concept maps, Venn diagrams and tree diagrams. Other graphic organizers identified found to be useful for reading text: network trees, fishbone maps, cycles, spider webs, continua/scales, series of events chains, compare/contrast matrices and a problem/solution outlines. Hartman (2001a,b) describes experiences in using a scaffolding approach to teach students how to create graphic organizers as learning strategies. Hartman (2001b) affirms that scaffolding is much more effective for developing this metacognitive strategy than simply modelling, which has yielded only moderate success in her experience.

2.8 Problem solving and metacognition

Resnick and Glaser (1976) argue that the ability to solve problems is a major aspect of intelligence. We solve problems every time we achieve something without having known beforehand how to do so. What constitutes problem solving varies from person to person (Schoenfeld, 1987). What is challenging and demanding to one person might be easy and straightforward to another, and thus not constitute a problem solving situation. Problem solving involves an interaction of a person’s experience and the demands of the task. There is much evidence to demonstrate that those who are flexible and perseverant in problem solving, and who consciously apply their intellectual skills, are those who possess well-developed metacognitive abilities (Artzt and Armour-Thomas, 1992; Swanson, 1990; Whimbey and Lochhead, 1999). The literature also points to several metacognitive regulatory skills important for problem solving, such as planning and monitoring (Mayer, 1999).

Among the learning activities students perform, we have chosen problem solving as the type of learning activity around which to build the Reflection Assistant Model proposed in this thesis. As such, we focused on those aspects of metacognition which are most used in problem solving: knowledge monitoring, including problem comprehension; planning general heuristics and strategies; and evaluating the problem solving process.

An extensive body of research has been carried out on problem solving (for a good overview see Eysenck (1991) and for a more in depth review see Halpern (1996)). We do not intend to make a review of this area of inquiry. Instead, our purpose here is to discuss only those aspects which are relevant to the development of a framework for metacognition instruction in the domain of maths problem solving. We focus on topics that provide the building blocks to our model: (1) stages and characteristics of (maths) problem solving, (2) general heuristics for problem solving, and (3) self-regulatory skills for problem solving. And from the existing conceptualizations of the cognitive processes in problem solving, we have selected three that are used as the conceptual
basis for our proposed environment: (1) George Polya’s general heuristics for problem solving, (2) Alan Schoenfeld’s framework for self-regulatory skills development, and (3) Richard Mayer’s stages and strategies for maths word problems (this one is presented in Chapter 3, as part of the specific literature review in algebra word problems). There are similarities in the conceptualization of maths problem solving by Polya (1945), Schoenfeld (1985, 1987, 1992) and Mayer (1999) since all of them emphasize metacognition and self-regulation as core elements for developing expertise in solving non-routine problems.

This discussion aims at eliciting the requirements for a framework that provides a balanced approach for problem solving, highlighting important metacognitive strategies and knowledge to foster competence of learners as problem solvers and providing sound cognitive strategies for maths problem solving.

2.8.1 The characteristics of problems and how metacognition helps

All problems contain three important characteristics: givens, a goal, and obstacles. The givens are the elements, their relations, and the conditions that compose the initial state of the problem situation. The obstacles are the characteristics of both the problem solver and the problem situation that make it difficult for the solver to transform the initial state of the problem into the desired state. The goal is the solution or desired outcome of the problem. Problem solving is the active process of trying to transform the initial state of a problem into the desired one (Davidson et al., 1994; Eysenck, 1991). From a general perspective metacognition helps the problem solver to:

1. recognize that there is a problem to be solved;
2. figure out what exactly the problem is;
3. understand how to reach the solution.

Several metacognitive processes help individuals identify the givens, goals and obstacles of problem solving. They guide the active process of transforming the initial state of a problem into the desired state, and in this manner they are responsible for: problem identification and representation, planning how to proceed, and evaluation of the solution obtained (Davidson et al., 1994).

Conceptual stages of problem solving

An assumption underlying research in mathematical problem solving is that expert problem solvers engage in both cognitive and metacognitive behaviours as they attempt to solve the problem in all three cyclic stages of problem solving (Garofalo and Lester, 1985; Halpern, 1996; Artzt and Armour-Thomas, 1998):

Preparation to solve the problem This first stage is spent understanding the nature of the problem: its desired goal, and the givens. This is a crucial part in problem solving because a correct solution cannot be generated without an adequate understanding of the problem. It is also called familiarization stage (Halpern, 1996, p. 320).

Actual problem solving This stage is also called production stage (Halpern, 1996). During this stage the problem solver produces the solution paths that define the problem space.

Verification of problem solving Also called judgement or evaluation stage (Halpern, 1996). During this stage the problem solver evaluates the solution paths in order to select the best one.
2.8.2 Heuristics in problem solving

Problem solving is the process of moving toward a goal when the path to that goal is uncertain. Normally progress towards the goal is incremental: the problem solver goes step by step, making some false moves but gradually getting closer to the end point. Having something to guide the problem solver to make the choices that yield in progress is very important, and for that, one can use heuristics, which are powerful general strategies.

Therefore, the learner should know what heuristics are and should be aware of their power. Moreover, the learner should have both general and specific heuristics at her disposal. General heuristics can be applied in a variety of problems; they are usually content-free and thus apply across many different situations. Researchers agree that students who know about the different kinds of heuristics, including general ones, will be more likely to use them when confronting different classroom tasks (Bransford et al., 1999; Whimbey and Lochhead, 1999; Schoenfeld, 1992).

However, in order to apply a heuristic, one needs to engage in a process of heuristic search. In this context it is necessary to monitor the progress continually and switch strategies whenever necessary. This monitoring process requires metacognitive knowledge and self-regulatory skills. For example, Artzt and Armour-Thomas (1992) found that metacognitive awareness of mathematical problem solving strategies among grade school children was associated with the increased use of problem-solving heuristics and higher-level group responses to problems.

Polya’s heuristic for maths problem solving

The mathematician George Polya produced an influential piece of work on problem solving Polya (1945). Although he does not use the term “metacognition” in his work, he gives primary importance to “thinking about the process” in his notion of successful problem solving. He envisions mathematical problem solving as a task that has a specific goal and that provides some challenge to the problem solver. Progress toward such a goal is incremental and each move should be informed by a repertoire of strategies and heuristics. Because false moves may occur, the problem solver needs to monitor her progress continuously and go back to previous moves and change strategies if necessary. He advocates for the problem solver to be aware of her heuristics and consciously apply them.

Polya (1945) describes a general heuristic for problem solving that breaks down the problem into pieces and creates various subgoals, making each of those pieces manageable. However, having several subgoals creates the need of monitoring those subgoals. Polya suggests self-directed questions that help the problem solver to monitor their progress, what is commonly called “Polya’s heuristic” (see Figure 2.4). His method is a four-step process: first, one needs to understand the problem at hand, trying to break it into smaller parts and identify the data and unknowns; then, one goes on to creating a plan that aims at linking the data and unknowns previously identified. One may be obliged to consider auxiliary problems if an immediate connection cannot be found. Eventually, one should obtain a plan of the solution; then, one carries out the plan, monitoring and checking each step; and, finally, after building the whole solution one should look back and check the results.

Of course, this process is not linear; the learner occasionally needs to go back and do the previous step again and again, until the goal is reached. For example, in the attempt of making a
good plan the student may discover a need to understand the problem better. So, the student has
to go back to develop a new understanding of the problem.

Polya also suggests some general strategies or actions, such as “draw a figure” for understanding
the problem. This example also shows his acknowledgement of the necessity of constructing
an external representation of the understanding of the problem. Other possible representations that
he suggests are: “list the parts of the problem”, “make a diagram”, and “paraphrase the problem”.

“Polya’s heuristic” is dynamic and iterative in its nature, because the problem solver should
go back at any moment to a previous stage whenever it is necessary. Many later problem solving
frameworks in maths textbooks acknowledge some relationship with Polya’s problem solving
heuristic.

2.8.3 Self-Regulation and goal management

When someone is solving a problem one way of reaching the goal is to break it down into subgoals
and work on those. At that point, the cognitive challenge becomes goal management - keeping
track of what to do and when. Researchers regard goal management as a central feature of problem
solving (Zimmerman, 2000). This management involves the capacity to examine and control one’s
own thoughts. It requires one to be aware of the current activity and of the overall goal, the
strategies used to attain the goal, and the effectiveness of those strategies. We could summarise
this self-monitoring in one important question: “What should I do now, given my goal, my current
position, and the resources available to me?”. Self-regulatory behaviours in mathematics problem solving include clarifying problem goals, understanding concepts, applying knowledge to reach goals, and monitoring progress toward a
solution (Gourgey, 1998). These are essential skills that students should acquire and develop. In instructional settings, teachers should use problem solving as a learning activity and help students become self-regulators of their own problem solving process (De Corte et al., 2000).

De Corte et al. (2000) present a good summary of theoretical frameworks, studies and powerful learning environments that emphasize self-regulation as an essential part and also a goal of mathematics education. Below we make a summary of Schoenfeld’s method, which inspired our own framework.

**Schoenfeld’s method for metacognitive and heuristic training in (maths) problem solving**

Alan Schoenfeld has done a great deal of research on metacognition and maths problem solving primarily at college level. He puts forth the view that understanding and teaching mathematics should be approached from a problem solving perspective and that mathematics should be studied to help students learn “how to think” (Schoenfeld, 1982, p. 32). He describes the problem solving process as a dialogue between the problem solver’s prior knowledge, his attempts, and his thoughts along the way; thus, to accomplish this task successfully he suggests that explicit teaching of self-regulation skills should be an objective of maths problem solving classes (Schoenfeld, 1992).

According to Schoenfeld (1987) there are three distinct aspects of metacognition that are relevant in the learning of mathematics: knowledge, beliefs (and intuitions), and self-regulation.

- **The metacognitive knowledge component relevant for problem solving** is one’s knowledge of one’s own thought processes. Schoenfeld affirms that one’s approach to a task and one’s understanding of how to solve that task are affected by the extent to which one can realistically assess what one is capable of learning.

- **Beliefs and intuitions** are ideas about a learning topic one brings to work. They function as interpretative filters and implicit assumptions that shape the way one interprets the learning task. Schoenfeld states that students build their mathematical frameworks from their beliefs, intuitions, and past experiences trying to understand and make sense of the world. For example, one such belief could be that classroom mathematics is formulaic, non-negotiable, and not related to the outside world.

- **Self-regulation** refers to how well one keeps track of what one is doing, and how well one uses the input from this monitoring to guide one’s problem solving actions. Another way to think about this is as awareness of one’s thinking and one’s progress in solving a problem. Schoenfeld suggests a management approach to promote self-regulation; aspects of this management include: (a) making sure that one understands what a problem is all about before hastily attempting to solve it; (b) planning; (c) keeping track of how well things are going during a solution; and (d) allocating resources, or deciding what to do, and for how long, as one works on the problem (Schoenfeld, 1987, p. 190-191).

Based on this, Schoenfeld concludes that successful solving of mathematical problems depends up on a combination of four categories of knowledge/skills: (1) resource knowledge, including procedural knowledge of mathematics; (2) heuristics; (3) control processes, i.e. decisions about when and what resources and strategies to use; and (4) beliefs.

He affirms that it is not sufficient to teach students isolated heuristics procedures; those heuristics will remain inert and students will be unable to decide which one to use. Heuristics should be taught in the context of a metacognitive strategy that supports the learner in selecting the right one to solve a problem. Schoenfeld elaborated a regulatory strategy teaching method consisting of five
stages (Table 2.3) together with the most important heuristics to be used in the stages of Analysis, Exploration and Verification.

<table>
<thead>
<tr>
<th>Stages of the Problem-Solving Regulatory Strategy</th>
<th>Most important heuristics suggested</th>
</tr>
</thead>
</table>
| **Analysis** of the problem aimed at constructing an adequate representation | ▶ Draw a diagram if at all possible  
▶ Examine special cases of the problem  
▶ Try to simplify the problem |
| Design of solution plan | ▶ Try to keep the goal perspective on what you are doing and proceed hierarchically; i.e. monitor your actions and adapt them according to your goals |
| **Exploration** oriented toward transforming the problem into a routine task | ▶ Consider equivalent problems by replacing conditions by equivalent ones, by recombining the elements of the problem in different ways, by introducing auxiliary elements, or by reformulating the problem  
▶ Try to decompose the problem in subgoals, and work on them one by one  
▶ Look for a related or analogous problem where the solution method may be useful with a view to solving the new problem |
| Implementation of the solution plan | |
| **Verification** of the solution | ▶ Check whether all pertinent data were used and whether the solution conforms to reasonable estimates, predictions, or both  
▶ Check whether the solution could have been obtained differently |

Table 2.3: Schoenfeld’s method for metacognition teaching. It is based on explicit metacognitive and heuristics strategy teaching in problem solving.

According to Schoenfeld (1985) the Design Stage (2) is not really a separate phase, but something that pervades the entire solution process; its function being to ensure that the learner is engaged in activities most likely to be profitable at any moment. Most often, this means keeping a goal perspective on what one is doing and proceeding hierarchically. So, it is a self-regulatory component of his method.

Schoenfeld’s theory and instructional method is supported by extensive protocol analysis of students solving unfamiliar maths problems. He found that around 60% of the college students tend to read the problem and quickly choose an approach to it, then they pursue this direction, even if they have clear evidence that they are not making progress. As a consequence of this first, quick, wrong decision, they fail to solve the problem Schoenfeld (1992). Moreover, at the end of the attempt they are not able to explain how that approach could have helped them solve the problem. In his experiments, the instructor conducted problem solving classes in which students
spent one third of their time solving problems in small groups, following his method. Initially, the instructor explained the regulatory method and demonstrated how an expert selects and applies the heuristics. Afterwards students were given opportunities to practice the method and the instructor took the role of a consultant, providing external regulation in the form of hints, prompts, feedback, etc. He prompted the students from time to time with the following questions:

- What exactly are you doing? (Can you describe it precisely?)
- Why are you doing it? (How does it fit into the solution?)
- How does it help you? (What will you do with the outcome when you obtain it?)

Asking these questions serves two purposes: it encourages students to articulate their problem solving strategies and it induces reflection on those activities. The ultimate goal of this intervention is that students spontaneously and individually ask themselves the three questions, and in doing so regulate and monitor their own cognitive processes. An interesting point is that, at the beginning of the training, students felt uncomfortable with these questions and were lost as to how to answer them; as the instructor continued prompting them, they started to discuss these questions in their small groups in advance. By the end of the training (which took one whole term) this behaviour had become habitual. Schoenfeld stresses that it is not a question of being able to solve the problem, for to a significant degree solving non-standard problems is a matter of luck and prior knowledge (Schoenfeld, 1992). The point is that, through self-regulation, students gave themselves the opportunity to solve the problem by analysing their attempts and giving up unproductive paths. After the training, fewer than 20% of the students followed the original “jump into one solution attempt and pursue it no matter what” approach (see Figure 2.5).

### 2.9 Conclusion

From the review of literature in previous sections it becomes clear that metacognition is a multifaceted topic of research.

The empirical studies lead us to conclude that metacognitive training is a challenging task, involving more than just adding a few new activities in learning environments. In order to achieve observable improvements it is necessary to tailor the metacognitive message to the domain and blend it seamlessly into the teaching.

Our research has benefited from the literature that has identified the metacognitive skills that play an important role in problem solving and from empirical studies that successfully trained students with a metacognitive method. However, transposing these findings in a computational environment still poses big challenges. The main ones are doing without the teacher’s ability to adapt the timing and amount of metacognitive scaffolding to individual needs.

Although, there have been some attempts of helping students develop metacognition in the areas of Computers in Education and Intelligent Tutoring Systems, only a few have target explicitly metacognitive training as the central goal or have obtained significant results. A review of the most relevant and recent computer-based learning systems that incorporate some kind of metacognitive training is presented in the next chapter.
Figure 2.5: Comparison of problem solving behaviour in Schoenfeld’s intervention. Before instruction 60% of the solution processes resembled graph (A); by the end of the instruction many students demonstrated an expert-like problem-solving behaviour (B & C). (Schoenfeld, 1985)
Chapter 3

Metacognition in Interactive Learning Environments

3.1 Introduction

The previous chapter reviewed research on metacognition in education. This chapter focuses on attempts to incorporate metacognition instruction into Interactive Learning Environments (ILEs) and Intelligent Tutoring Systems (ITSs).

Computers enable new and unique ways of capturing and representing metacognitive knowledge and metacognitive skills, such as graphical reification, and (abstracted) replays and reviews. They can also improve on traditional metacognitive approaches, like self-explanation, and collaboration by presenting them in a more dynamic and interesting form.

The main issues regarding the design of such metacognitive activities in ILEs are discussed in this chapter and a classification schema of the ways in which ILEs can target metacognition is proposed. Examples from the literature are brought together to illustrate the use of this schema and to give a summary of previous research on metacognition instruction and self-reflection in computer-based learning systems.

3.2 Computers as Metacognitive Tools

Computers have a lot of potential as metacognitive tools. For example, through their ability to record interactions with users, they can become powerful reflection tools. Having captured the actions of the student carrying out a task, these can be played back to her, properly abstracted and structured. This will help the student to become aware of her processes and help her improve performance on the task in question through reflection on the how’s and why’s of the chosen problem solving paths.

As collaborative learning devices, they can be programmed to support group planning, monitoring and evaluation of the learning process. Students in a small learning group can, for example, look back over their solution paths and compare them with other members of the group. This should trigger reflection on which changes could be improved (Reusser, 1993). Another interesting possibility is that of simulated “Learning Companions” acting as peers who encourage the student to reflect and articulate her actions (Goodman et al., 1998).
Despite this potential, the majority of computer-based learning environments to date have focused on supporting students develop domain-related skills and knowledge. Some attempts have been made to incorporate metacognitive components, mostly in the form of embedded reflection on the learning task or processes. And a very small number of systems have included explicit metacognition training as their main target. Detecting, tracing, modelling, and fostering students metacognitive and self-regulatory behaviours during learning in ILEs is a research challenge that is almost intact. Nevertheless, the examples from existing literature shows the prospects of this important area.

3.3 The Design of Metacognition in ILEs

Major instructional and design issues arise when learning systems intend to promote metacognition. The instructional issues are the creation of activities that are sensitive to the context and complexity of the learning task, and that take into account the student’s competence and previous metacognitive behaviours. Also, the definition of metacognitive mechanisms and components involves the careful design of their interface, the language used to talk about metacognitive topics, the timing for metacognitive instruction, etc.

The criteria used to decide what is the most suitable combination of the possible options can vary from domain to domain and depend on the kind of task proposed. But there are nonetheless two basic requirements which the designer should always take into account:

- be careful not to increase the student’s cognitive load;
- get students to recognize the importance of the metacognitive activities.

From the beginning, the design decisions have to match the goals of the metacognitive instruction within the ILE. The metacognition may have a limited role in the system, providing only an underlying frame for developing domain-related skills. In this case it can appear as a single component related to a specific task. It can for instance take the form of an animated assistant who reminds students to use the on-line help. At the other extreme, explicit scaffolding of metacognition may be the main goal of the ILE. In this case, most of the system’s components have to be designed to promote metacognitive development (e.g. a system that teaches students to plan and monitor their study time).

In the next section, we propose a classification schema that contemplates important issues that have to be considered in the design of metacognitive training in ILE.

3.3.1 A classification schema of metacognition in ILEs

We have organized what we consider the major generic issues that have to be taken into account in the design of metacognition instruction in ILEs into a classification schema. This schema can be used to guide decisions regarding both individual metacognitive components to be incorporated into the ILE as well as those relevant to the whole metacognitive module.

As shown in Figure 3.1, the classification schema has three dimensions: timing, target level, and instructional approach. Each dimension is independent from the others so that the option chosen in one dimension does not affect the possible choices in the other dimensions. The possible
values for the timing are not necessarily mutually exclusive. For example, the designer may decide to create a metacognitive activity that aims at fostering the planning of heuristics before and after a problem solving task, both activities targeting heuristics that are domain-dependent (i.e. normally useful for that particular domain) and decide to implement such activity using a collaboration approach.

Figure 3.1: Classification Schema for Metacognition Design in ILEs. The diagram shows the independent dimensions and representative values (not mutually exclusive) for each dimension.

Timing
This dimension refers to the moment chosen for promoting training on metacognitive skills and self-reflection. According to Flavell (1979) metacognitive experiences can occur at any time before, after, or during a cognitive enterprise. Thus, the moment in time where the system provides metacognitive instruction is an important issue to consider. As mentioned before one basic requirement is to not increase the student’s cognitive load. In a learning environment the learner is presented with new information and is asked to perform tasks of different complexity. Balancing the amount and complexity of tasks is important and the instructional designer has to be aware that metacognitive activities are probably new and demanding to students.

Rutz et al. (1999) affirm that questions of a generical and strategic nature are not as sensitive to timing as task specific ones. They developed an animated reflective assistant and tested the trust students had for it as a means to trigger reflection. They remarked that the hardest design issue with respect to trust is timing. Knowing when to interrupt the student, and having a good match between the content of the comment and the current task are key to capturing students’ confidence and consequently making them reflect.

The designer may choose to present the metacognitive activities at one or more of the following points in time:

Before the learner starts a new learning task: it can put the learner in the correct frame of mind for the task (before a new problem, before a new lesson, etc.);

During the learning task: it can support the student in the self-monitoring process, but the danger is cognitive overload;

After the completion of the learning task: it is a natural time to get the student to reflect on her performance and learning process.
Target Level

This dimension refers to the way the metacognitive component is embedded in the learning system. Since the same metacognitive skills are used in different contexts, they can be worked on at different levels of integration with the domain of study. Specifically, this classification considers the following levels of instruction:

**General level:** the metacognitive component works on metacognitive skills which are not specific to any domain.

**Domain-related level:** the metacognitive component targets metacognitive skills most useful to the domain e.g. planning reading goals for the domain of learning from texts, or monitoring step-checking if the domain is maths equations.

**Task-specific level:** the metacognitive component focusses on a task the student has to perform (solve an equation, read a narrative text, etc.) making her reflect on the steps she is taking to accomplish that task and on her overall performance.

In the last two cases, there is an additional distinction that can be drawn. In both these cases metacognition is not the only subject of attention of the student given that there is a context which is either a task or the domain. Thus either the student’s attention is drawn on the fact that extra work is being done on metacognition or the work on metacognition takes place but nothing is done to bring this fact to the attention of the student. We call the first case *explicit* metacognitive instruction and the second one *implicit*. Explicit metacognitive instruction creates more load on the student but it has the potential of having a more lasting effect since the student is aware of the work she is doing on her metacognition.

### Instructional approaches of the metacognitive component

To implement metacognitive activities in computer-based learning environments, several representation mechanisms have been adopted. The most common ones are collaboration (like peer collaboration and menu-driven dialogues), graphical reification, task replay and review (reflective follow-up) and self-reflective activities. The next section explains each one of these approaches and presents examples of existing training systems that incorporate each of those mechanisms.

Computer-based learning environments typically support metacognitive development by providing students either with graphical and representational tools for reflection and interpretations (Reusser, 1993; Schaubüle et al., 1993), or facilities for prompting self-reflection or help-seeking behaviour (Aleven and Koedinger, 2000). Examples of self-reflective activities in such environments include: reflective prompts or guided questioning that requires learners to justify their ideas and make them explicit (Lin and Lehman, 1999; Goodman et al., 1998), prompts to self-explain (Conati and Vanlehn, 2000), and features that support self-assessment (Schauble et al., 1993; Katz et al., 1998; Niemi and Latva-Karjanmaa, 2001; Katz, 2000). Some ILEs may combine more than one approach to deliver metacognitive training. MIST (Puntambekar and du Boulay, 1997), for example, combines reflective prompts, self-questioning and peer collaboration.

The approaches are detailed in the next section together with illustrative examples of ILEs that incorporate them.
3.4 ILEs that Implement Metacognitive Training

Different combinations of timing, target level and representation approach were used in previous ILEs to define the metacognitive activities. Table 3.1 shows representative examples of these systems organized according to our classification schema. One could say that there are many others ILEs which implicitly support metacognition or where the interaction with the system eventually generates metacognitive development (e.g. using collaborative systems). We only show here those that either explicitly train metacognition, or have metacognition training as part of their project goal.

3.4.1 Graphical reification in ILEs for metacognitive development

Reification is the operation by which something that was previously implicit and unexpressed or even possibly inexpressible (like an abstract concept, an idea, or a series of actions) is explicitly formulated and made available to conceptual manipulation.

Many of the actions a learner performs to solve a problem in an ILE can be reified. For example, imagine that during a problem solving session the student opens a help system twice, she accesses textual explanations about the type of problem, she sends a message to the tutor asking for clarification, etc. These actions are normally implicit to the process and may never be used as a source of study by the learner or the tutor. Applying reification to these actions, using tree or network structures, allows the ILE to present the trace of the learner’s actions. It is a pedagogical device that enables reflection on the problem solving and reasoning process.

The effects of reification could be enhanced with an active coaching strategy and scaffolding. This is a less attempted approach, not least because it is a very difficult one.

Early examples from research on tutoring systems already present graphical reification as a means to improve reflection on the problem-solving process. The Geometry tutor (Anderson et al., 1985) and its second improved version, the ANGLE tutor (Koedinger and Anderson, 1993), and Algebraland (Brown, 1985) are the best known examples of such systems. They provide a meta-level representation of the problem solution (also called “problem spaces” by Collins and Brown (1988)), making it possible for students to interact with the representation and to observe or change the path of the solution.

Later on, Derry and Hawkes (1993) developed TAPS, that besides providing graphical reification of the process of solving algebraic word-problems, incorporates a diagnostic model of errors using fuzzy logic that interprets patterns of errors in problem-solving performance. Another example is the Heron system (also called Heron-WATGRAF) (Reusser, 1993), a learning environment designed to help children in understanding and solving a wide class of complex mathematical word problems. It provides tools for assisting students to comprehend the language of a problem, select the main components that will be used in the solution, and construct an explicit and reified mathematical problem model from which a linear equation can be derived. Apart from the Geometry tutor (and ANGLE), all the others mentioned above are problem-solving environments in the domain of algebra.

It is consensual that through the use of structured visual displays of reasoning processes, monitoring and regulation are reinforced. Nevertheless, none of the systems above incorporated explicit scaffolding on planning, monitoring or regulation. Furthermore, from the examples above, only
### ILEs that include Metacognition Training

<table>
<thead>
<tr>
<th>System</th>
<th>Domain</th>
<th>Metacognition Development Intended</th>
<th>Timing</th>
<th>Target Level</th>
<th>Approach</th>
<th>Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algebraland (Brown, 1985)</td>
<td>Algebra equations</td>
<td>Monitoring Regulation Reflection on problem-solving process</td>
<td>During learning task</td>
<td>Task-specific skills</td>
<td>Graphical Reification</td>
<td>Limited empirical studies: indicated that subjects that acquired error-detection skills using the system also used these skills in problem solving done without the system.</td>
</tr>
<tr>
<td>Sherlock 2 (Katz et al., 1992)</td>
<td>Faults diagnosis in avionics</td>
<td>Regulation Evaluation</td>
<td>After learning task</td>
<td>Domain-related skills</td>
<td>Reflective follow-up activities</td>
<td>Simulated environment: learning by doing. Experiments show technicians improved troubleshooting skills and transferred them to other tasks.</td>
</tr>
<tr>
<td>TAPS (Derry and Hawkes, 1993)</td>
<td>Algebra word-problems</td>
<td>Planning Monitoring Error-recovery skills</td>
<td>During learning task</td>
<td>Task-specific skills</td>
<td>Graphical Reification</td>
<td>Uses Fuzzy Logic to match students’ solution to systems’ set of possible solutions. Planning is facilitated with menu of problem “schemas”. No experimental studies reported.</td>
</tr>
<tr>
<td>Heron (Reusser, 1993)</td>
<td>Complex algebra word-problems</td>
<td>Self-directed problem comprehension and conceptualization Planning</td>
<td>During learning task</td>
<td>Task-specific skills</td>
<td>Graphical Reification</td>
<td>Support problem understanding with explanatory help on text vocabulary. Strong use of the mouse to perform tasks. Uses cognitive task analysis (instead of student modelling) to perform behavioural diagnosis.</td>
</tr>
<tr>
<td>System</td>
<td>Domain</td>
<td>Metacognition Development Intended</td>
<td>Timing</td>
<td>Target Level</td>
<td>Approach</td>
<td>Observations</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------</td>
<td>-----------------------------------</td>
<td>-------------------------</td>
<td>-----------------------</td>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HyperClinic (Gama et al., 1997)</td>
<td>Cardiology</td>
<td>Reflection on prior knowledge Planning</td>
<td>Before learning task</td>
<td>Domain-related skills</td>
<td>Peer collaboration Reflective prompts</td>
<td>Case-based system. Used in real settings. No statistical analysis done.</td>
</tr>
<tr>
<td>LuCy (Goodman et al., 1998)</td>
<td>Satellite activity</td>
<td>Reflect on knowledge Articulate knowledge Review</td>
<td>After learning task</td>
<td>Task-specific skills</td>
<td>Collaboration: Learning Companion (LC)</td>
<td>The LC acts as an advisor and has expert knowledge. LC asks questions in natural language.</td>
</tr>
<tr>
<td>The Pedagogical Assistant (Rutz et al., 1999)</td>
<td>Object-oriented analysis</td>
<td>Monitoring Regulation</td>
<td>During learning task</td>
<td>Task-specific skills</td>
<td>Collaboration: Learning Companion (LC)</td>
<td>Animated LC in the format of a parrot Study conducted to check the degree of “trust” on the LC</td>
</tr>
<tr>
<td>System</td>
<td>Domain</td>
<td>Metacognition Development Intended</td>
<td>Timing</td>
<td>Target Level</td>
<td>Approach</td>
<td>Observations</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
<td>-------------------------------</td>
<td>-------------------</td>
<td>---------------------------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IQ Form (Niemi and Latvakanmaa, 2001)</td>
<td>Domain-independent</td>
<td>Awareness of self-as-learner Evaluation of learning strategies and skills</td>
<td>Before and during learning task</td>
<td>General skills</td>
<td>Self-assessment Personnal diaries</td>
<td>The tool creates students’ profiles. Based on previous self-assessment questionnaires. Tested with 4 different subject groups from distinct domain areas from distance higher education courses.</td>
</tr>
<tr>
<td>Embedded Assessment System (Katz et al., 2001)</td>
<td>Hypothesis testing in statistics</td>
<td>Knowledge monitoring Regulation Evaluation</td>
<td>During and After learning task</td>
<td>Task-specific skills</td>
<td>Self-reflection Self-assessment</td>
<td>Used to create a automatic assessment of metacognitive behaviours in ILEs. Tested experimentally (but with few subjects). Uses the Knowledge Monitoring Assessment as a comparison measure test.</td>
</tr>
<tr>
<td>ECOLAB II (Luckin and Hammerton, 2002)</td>
<td>Science: food chains and webs</td>
<td>Help-seeking awareness task selection skills</td>
<td>During learning task</td>
<td>General skills</td>
<td>Metacognitive prompts and messages</td>
<td>Scaffolds children to select optimally useful help Creates a metacognitive ability tag about the user (i.e., a simple metacognitive model)</td>
</tr>
<tr>
<td>Basic Inquiry Platform (Woolf et al., 2002)</td>
<td>Geology, biology, civil engineering, etc.</td>
<td>Inquiry Skills Ability to make predictions and test ideas Knowledge monitoring</td>
<td>During learning task</td>
<td>General Skills</td>
<td>Questioning Inquiry learning Cases</td>
<td>Complex software with different tools, phases and resources Domain independent infrastructure Main goal is to develop inquiry skills and self-critical learners</td>
</tr>
</tbody>
</table>

Table 3.1: Examples of ILEs that support metacognition.
the Geometry Tutor and Angle were used in real classrooms settings, with high-schools students.

All these systems use reification for slightly different purposes. In *Algebraland* the student is presented with an algebraic equation and a choice of basic operations she can perform to simplify the equation. The system structures the audit trail of the search space explored by the student as she applies the operations offered to her. So the student does not interact directly with the reification; instead she uses it as a visual display of her path towards the solution, as it reveals dead ends, and the multiplicity of correct solutions paths. The student can see exactly where she backed up, where she reached the same state twice, where she was getting farther away from a solution, and so on. Thus, the student is encouraged to generate multiple problem-solving trees. Also, the structured representation of partial solution paths provides an opportunity to reflect on problem-solving and evaluation strategies in the context of use (Collins and Brown, 1988). Studies conducted with the system investigated whether students who reviewed their problem-solving traces acquired general skills for avoiding errors or for detecting errors and recovering from them (Foss, 1987). However, the empirical studies were limited and indicated that the desired effect hardly ever occurred (Foss, 1987). But whenever the students acquired the error-detection skills it was positively associated with subsequent successful use of those skills in problem solving done without the system.

The *Geometry Tutor* provides a representation of the concept of proof. Using the reification it is shown that a proof is a path between axioms and theorem, and certain actions are shown to be fruitless because their results lie on incomplete paths. Students have to use both forward and backwards inferences to connect the premisses and the conclusions of the proof. This graphical visualisation of geometry proofs emphasizes the overall process of completing the proof rather than the final product. This points to Brown (1985) who says that by placing the emphasis on the entire thought process, reification favours deeper forms of learning rooted in reflexive recapitulations of mental processes. Later on Koedinger and Anderson (1993) designed the *ANGLE* tutor (A New Geometry Learning Environment), a second generation of the Geometry Tutor. They intended to further address the issue of “implicit planning”, i.e. the thinking process behind competent performance that is under the surface and not revealed in the actions performed, nor made explicit in textbooks or typical instruction. So, they created interface notations to allow students to make a plan of the proof, not worrying about the details of the proof (e.g. icons for representing generic schema categories). After a proof plan has been discovered the student could fill in the details, adding rules and statements to the solution.

The *TAPS* system (Teaching Arithmetic Problem Solving) is an ITS system, which was designed to help students solve arithmetic word problems (Derry and Hawkes, 1993). Its major cognitive tool is a graphical interface that facilitates the construction of problems trees. The student chooses from a menu of blank subtree diagrams (called “schemas”) and fills the blank nodes of the tree with labels, values and operators. An interesting and innovative feature is the use of fuzzy logic to diagnose errors and patterns of errors, enabling the system to carry out on-line performance monitoring and error recognition. For this purpose the system has a knowledge base with possible sub-trees for each problem. However, TAPS does not use any measure of student’s progress for choosing future activities. Although the error recognition could trigger local tutoring decisions, a tutoring module was not implemented. In sum, the system has great potential to encourage students to engage in planning and self-monitoring, but experimental studies need to be
conducted to confirm it. As far as we know, no such experimental studies have been reported.

Heron goes beyond the problem space reification and combines it with tools for understanding the description of complex algebra word-problems. Two kinds of strategy support for text comprehension are proposed: (1) explanatory help with respect to the vocabulary, syntaxes and semantics of the problem, and (2) aid with identification and conceptualization of problem components together with support to build the solution tree (or planning tree, as he calls it). This graphical reification in the format of a solution tree bridges the gap between the text of the problem and the linear equation that has to be produced. Heron was explicitly designed as a non-directive cognitive tool with high level of learner control.

An empirical study with four pairs of students was conducted and the subjects’ dialogues were videotaped (Reusser et al., 1996). An analysis of the dialogues was done focusing on Heron’s assistance on the qualitative understanding of the problem situation. The results provide evidence of quantitative and qualitative changes in the conversation, related to both the communicative and the task-related aspect of peer collaboration. The results also clearly show that for some students there is a need for more help or control by either the system or a participating teacher or expert. As a consequence of the results of this initial analysis, a new version of Heron was developed including a feedback component that provides some degree of help. Unfortunately, we found no indication that another, more extensive, and conclusive evaluation was conducted.

3.4.2 Reflection through follow-up activities

Using the recorded process trace, or audit trail to encourage reflection on problem-solving strategies is another useful approach. This kind of post-task reflection is called reflective follow-up (Katz et al., 1992) and provide students with capabilities for seeing replays or reviewing their performance and actions. The difficulty of this approach is to design an appropriate abstraction of the audit trail. The Sherlock 2 tutor (Katz et al., 1998) is a good example of a tutoring system that uses reflective follow-up activities.

Sherlock 2 is one of a series of systems developed as tools for training avionics technicians in diagnosing faults and repairing faulty aircraft and the systems used to maintain them. It is a realistic computer simulation of the actual job environment. Besides training in the domain, it aims to develop general diagnostic skills that can be readily transferred to related job specialties. Sherlock 2 presents trainees with a series of exercises of increasing difficulty. There are two main phases of a problem in the system: problem solving and review. The review is called reflective follow-up (RFU); it is a post-problem reflective phase where students reflect on their problem-solving performance and receive feedback from Sherlock 2. More specifically, during the RFU, students can step through their solution and receive feedback from the computer coach, replay the simulated expert’s solution, review the instructional goals of the tutor, ask questions about their own or the simulated expert’s actions, and get suggestions about how to improve in the next session.

The basic idea of Sherlock’s coaching scheme is to give students control over their own learning and to help them develop metacognitive skills by requiring them to figure out for themselves what type of information they need. Likewise, the reflective follow-up activity promotes learning by helping the student see another viewpoint on solving the problem grounded in the context of
Chapter 3. Metacognition in Interactive Learning Environments

the student’s own solution. This approach has been demonstrated to be effective on the hardest tasks.

Studies showed that learners who used Sherlock 2 improved dramatically in their troubleshooting skills, during training as well as on a post-test (Gott et al., 1996).

3.4.3 Metacognition through collaboration

Based on the Vygotskian concept that understanding one’s own learning is enhanced by interaction with others during learning, some systems have adopted collaboration as a basis to promote reflection (Vygotsky, 1978).

Systems, like LuCy (Goodman et al., 1998) and the Animated Pedagogical Assistant prototype (Rutz et al., 1999), use peer collaboration in the form of learning companions. Goodman et al. (1998) justify this approach stating that reflection and articulation are two key pedagogical functions underlying student-learning companion dialogues. A learning companion can help a student reflect on her thinking by critiquing, questioning, or evaluating particular steps. Similarly, students and learning companions may articulate those steps through further explanation or elaboration.

Other systems combine reflective prompts with collaboration, triggering students’ dialogues through system’s questioning. MIST (Puntambekar and du Boulay, 1997, 1999) for example uses menu-driven system-student dialogues with students working in pairs. MIST encourages students to think about what they have been doing and what they want to do using an active strategy of questioning.

The HyperClinic system (Gama et al., 1997) is another example of a collaborative environment with reflective prompts. It is a web-based case-based application in the domain of cardiology that proposes reflective questions to a group of students on a specific clinical case as a preparation to solving the case. HyperClinic provides a newsgroup tool to promote discussion about clinical knowledge relevant to the case and possible strategies to elaborate an appropriate diagnosis. The main goal of the discussion is to activate students’ prior knowledge and help on articulating their thinking. Trials showed that the participation of senior physicians in the discussion, offering tips and advices, helped students to concentrate on the hard aspects of the case and articulate their thinking better. One interesting result was that, even being an informal discussion, some (not so novice) students felt embarrassed expressing their confusion or lack of knowledge in the discussion list that included the experts. So we removed student identification which generated an increased participation by the students.

MIST (Metacognition in Studying from Texts) is designed to help students develop a systematic approach to learning from texts by supporting a range of planning and monitoring activities. It helps students become aware of the processes they might engage in when learning from texts. The emphasis is on the process and not on the product of learning. Thus, the interface presents opportunities to discuss and reflect on the different types of activities students might engage in when learning from texts rather than on the specific text they are learning. Therefore, the system addresses the metacognitive skills relevant for the domain, focusing on the students’ repertoire of strategies and how they will use them during the study session. MIST deals with the major goals of learning from texts, which are: work out the core information contained in the texts, reduce it to its main ideas, comprehend it, and integrate it with the learners’ existing prior knowledge.
Chapter 3. Metacognition in Interactive Learning Environments

The first part of the system helps students become aware of the various strategies that can be used as well as where, when, and how they should be used; it also teaches students to pay attention to the different types of tasks and texts and activate any prior knowledge that they may have (Puntambekar and du Boulay, 1999, p. 249). In the second part, students can choose options relating to planning, text processing or reading, and memory enhancing. Because MIST has knowledge about the types of activities that students might engage in during learning from texts, rather than specific knowledge about the texts that students wish to study from, intervention by the system is necessarily at the meta level and is meant to help students think about what they are doing and why they are doing it. Students use MIST in pairs and the system’s questions and activities trigger a collaborative reflection on the metacognitive skills used when learning from texts. MIST is one of the few examples of systems whose primary objective is to train metacognition.

LuCy is a simulated learning companion, acting as a peer in an intelligent tutoring system, called PROPA. LuCy was developed to encourage the student to reflect on and articulate her past actions, and to discuss her future intentions, like alternative courses of action, and their consequences. LuCy has an inquiry interface. Goodman et al. (1998) state that peers encourage each other to reflect on what they are learning and to articulate their thinking, which enhances the learning process. So LuCy aims to promote more effective instructional exchanges between students and the intelligent tutoring system. By initiating a menu-driven dialogue with a student, the system forces reflection and articulation on the student’s learning.

3.4.4 Self-explanations in ILEs triggering metacognition

Self-explanations encourage the student to reflect on and articulate her past actions (Chi et al., 1989). However, as already mentioned in the previous chapter, studies show that students do not spontaneously engage in self-explanation and need guidance to do it (Bielaczyc et al., 1995). One issue is how to motivate students in an ILE to do so and how the self-explanations can be used for tutoring purposes.

Conati and Vanlehn (2000) developed the SE-Coach, a system that support students learning from examples through self-explanations. The system acts within Andes, a system that tutors on Newtonian mechanics. The SE-Coach includes an interface to scaffold students’ self-explanations, a probabilistic student model and a coaching component. The student model integrates information on the students actions, the model of correct self-explanation and the students domain knowledge and thus enables the assessment of example understanding from reading and self-explanation actions. Based on these evaluations the SE-Coach changes the level of scaffolding by eliciting further self-explanations whenever necessary. There are three different and incremental levels of self-explanation scaffolding.

The SE-Coach interface uses a clever masking mechanism that requires the student to explicitly move the mouse over the part of the example she wants to read to uncover it. This makes it possible for the system to know which parts of the example the student has seen and for how long. Unfortunately, the system does not allow self-explanations in natural language (not least because it would be a very complex task). Instead, the student builds a sentence through a series of choices in lists of options. The student is not verbalizing using her own words thus not guaranteeing that true self-explanation is occurring.
Another limitation is that the system only provides guidance to self-explanations that are local to a certain solution step, asking students to justify the step based on inferences of previous knowledge demonstrated. It is not possible then to ask students to explain steps that are missing, which is common in experts’ solutions.

A pre-post test experiment was conducted with 56 college students who were taking introductory physics classes. The results suggest that in the experimental condition (N=29) (i.e. students using a fully functioning version of SE-Coach) students at an early learning stage were the ones who benefited best from the structured scaffolding. The explanation was that because they were still unfamiliar with the subject matter, they are more motivated to put substantial effort in exploiting the help at best. They did not find similar results with the more proficient students. Although very interesting and promising, the SE-Coach needs more conclusive indications of its benefits for developing students’ self-explanations and monitoring of comprehension.

3.4.5 Self-assessment in ILEs

A more recent attempt to develop an environment focused on metacognition is the **IQ FORM** (Intelligent Questionnaire Platform). It is part of a research project conducted at the University of Helsinki by an interdisciplinary research group (Niemi and Latva-Karjanmaa, 2001; Virtanen et al., 2003). Differently from all the others examples in this chapter, this learning environment is not attached to any specific domain and, therefore, targets metacognition development for learning in general.

The IQ FORM is a set of interactive Web-based learning tools for promoting higher education learners’ self-regulation, learning skills and strategies, and supporting collaborative processes in virtual courses for distance education. The tool set has two parts: (1) a tool for assessing a learner’s individual qualities and learning skills (**The IQ Learn**) and (2) a tool that offers information about group processes to promote students participatory skills, collaborative learning and knowledge creation in e-learning (**The IQ Team**). Both parts consist of interactive questionnaires for self-assessment and tutorial packages that provide information and concrete advice about how one can develop metacognitive skills and behaviour. The system also includes a learning diary where the learner can record, for example, the ideas inspired by the assignments of the tutorial package. In addition, the system offers the teachers a special tutoring package with information and advice about how to make the learning easier and more meaningful for learners studying on a Web course.

One of the tools provides reference information about the study group and each student may compare her learning skills to those of other students in the group.

**The IQ Learn** tool consists of three elements:

1. An interactive test bank with three questionnaire sets for students’ self-evaluation (Figure 3.2) and graphic tests result:

   - **forethought of learning**: expectations of success, performance anxiety, meaning of studies, self-efficacy, and self-confidence.
   - **strategies in learning**: time management, self-management, persistency, and help-seeking strategies.
   - **learning skills**: rehearsal, critical thinking, finding essential points, connecting new and old knowledge, keywords and advanced organizers, application of theories, and self-assessment.
2. The tutoring sets, with a hypertext structure for each sub-component of the tests. It has texts directed to the learner and the teacher.

3. A learning diary for the reflection on profiles and learning experiences.

Figure 3.2: Snapshot of IQ Form SystemExample of the IQ Learn interactive self-questionnaire.

The IQ-Team tool is still under development and consists of:

1. An interactive test bank with three questionnaire sets:

   **Group Roles**: rejection, dominance, encouraging, conforming, sharing know how, avoidance.

   **Social Interdependence**: individualistic, competitive, cooperative.

   **Group Processes and Knowledge Creating Process**: atmosphere, goal orientation and commitment, innovation and creativity, the benefits of doing together, the utilization of differences, the role of the tutor.

2. The tutoring sets.

3. A joint learning diary called “the Log book” for reflections, discussion and knowledge creation.

The IQ Learn component of the IQ Form was used and evaluated in a pilot study with 4 different groups of higher education students from different subject areas. Students were asked to use it freely, according to their own interests, while studying on an on-line distance course and to write in their learning diaries how they used the IQ Learn questionnaires and tutoring tool. At the
end of the course, they were asked to compare their comments in their diaries with their IQ Learn results.

This pilot study revealed that some students benefit from the virtual tutoring, mainly those who are at an early stage of their studies, or have difficulties in learning. For example, some younger students on one subject group reported that they felt that the IQ Learn had helped them to find the reason for their lack in learning (e.g. lack of time management and confidence). The concrete issues of self-regulation such as time-management and tips to cope with test anxiety were considered as very useful by the subjects. Students with stable and effective learning strategies reported that the system strengthened their understanding of their learning. However, many students only used the tool superficially, and only gained most from the system when they were guided in to use it, by giving clear assignments.

As general as it is, the IQ Form resembles the Cognitive Strategy Instruction (CSI) programs (mentioned in Section 2.5) and faces the danger of becoming abstract and disconnected from real learning experiences. Students may acquire new information about themselves as a learner and possible remedial strategies but may not detect when and how to use them. Moreover the tutorial texts are static and do not adapt to students’ needs or courses’ needs. So, the tips and orientations may not all be relevant to the learning task at hand.

Nevertheless, the IQ Form may have interesting uses, if it is tailored to specific instructional goals of a given course. The questionnaires tackle many important issues, but perhaps too many issues for a student to explore in a single course. Selecting sub-components of the questionnaires, adapting the tutoring texts to content areas, and combining them with the course exercises could provide a powerful environment for metacognitive development. Also, it is necessary to develop tools for monitoring students’ progress. It would be interesting for the student to gain awareness of which activities suggested provided best learning improvements when used and which ones did not help them.

Our guess is that IQ-Team tool, when developed and tested, may prove to be more effective than the IQ-Learn, because in distance learning courses students need a good framework for group interactions.

Next, we present a system called “Embedded Assessment System”. It is an interesting example of an attempt to create measures for assessing metacognition in an ILE.

### 3.5 Designing automatic measures of metacognition

Katz et al. (2001) have developed a web-based prototype called **Embedded Assessment System** for problem solving in the domain of statistics. The topic chosen was hypothesis testing because it was identified as particularly challenging for students to learn and for instructors to teach.

The prototype system is part of a larger research project that aims at designing automatic models for adjusting metacognitive scaffolding in computer-based learning environments. As such, this prototype aims at detecting types of metacognitive behaviour students engage while solving problems in order to identify measures that have general applicability and the potential to be incorporated into computerized learning environments.

The prototype is structured as a scaffolded learning environment for problem solving. Figures 3.3, 3.4 and 3.5 show snapshots of the prototype. For each problem proposed the student
is expected to perform four types of activities: problem categorization, construction of problem solution, reflection, and self-assessment. The components and activities are summarized below.

![Figure 3.3: Embedded Metacognitive Assessment Prototype- snapshot 1. Start new problem: problem categorization.](image)

The problem solving activity is organized as a sequence of goals that need to be accomplished to solve the hypothesis-testing problem. So, the solution to the problem is obtained by answering separate questions. Students have to fill 6 sub-solutions, starting with categorizing the problem, and justifying their choice as shown in Figure 3.3. Then, the student has to establish the hypotheses, select a statistical test, justify the statistical test, perform the hypothesis test, and finally draw conclusions - Figure 3.4.

Next to each guided question there is a reflective prompt (e.g. “What information led you to these hypotheses? Why did you choose a two-sided solution?”) as shown in Figure 3.4. Students are expected to answer the goals questions and the prompting questions. The purpose of embedding these prompts is to encourage students to reflect on and explain their problem solving as they work towards a solution, i.e. generate explanations for each solution given. As the answering order is not controlled by the system and it is possible for students to respond to the solution steps and reflective prompts in the desired order.

The prototype incorporates a help system. For each question students have to answer there are two kinds of help: guiding questions, that aim at focusing learners on important parts of the problem; and textbook, which provides more detailed information on how to accomplish the goal.

After solving the problem, students have to assess their own solution. For that purpose, the student’s solution is presented (both solution steps and explanations) together with an expert’s solution (both solution steps and explanations). Students are requested to judge whether their solution is equivalent or not to the expert’s; if not, learners are prompted to compare their solutions and explanations with those of experts, providing additional opportunities for self-explanation and remediation - Figure 3.5.

An experiment was performed using twelve students who had just completed the yearlong AP Statistics class. The study was conducted across three sessions and all students participated in the
same way in the sessions. Table 3.2 summarises the experiment design.

All participants worked on the same problem, which was of medium-to-low difficulty. Some students attempted additional problems, but because these extra problems differed among the students, the experiment analysis focuses on student performance only on the first problem.

One interesting aspect of this study are the measures created to analyse metacognitive behaviour. Metacognition was measured in this experiment through students’ interaction with the metacognitive prompts, help system, and self-assessment. The level and quality of student engagement in the proposed activities in the prototype were indicative of their knowledge monitoring behaviour. All nine measures are summarised in Table 3.3 below. As an external validity check for the measures they assessed students’ knowledge monitoring skill in statistics independent of the learning environment. For this purpose they used the Knowledge Monitoring Assessment instrument created by Tobias and Everson (as defined in Section 2.4.2) adapted to statistics problem solving (they called it StatKMA). The intent was to determine the extent to which any of the metacognitive online measures matched the metacognitive skill as assessed by the StatKMA.

The interaction with the prototype was recorded and the log was used to detect patterns of problem solving, omission of responses to solution steps, metacognitive prompts, self-assessment activities, editing behaviour students might have engaged in and their use of help.

In their analysis Katz et al. (2001) found a strong relation between SAT-M and Knowledge
Chapter 3. Metacognition in Interactive Learning Environments

Figure 3.5: Embedded Metacognitive Assessment Prototype - snapshot 3. Self-assessment.

<table>
<thead>
<tr>
<th>1st Session</th>
<th>2nd Session (in computer lab)</th>
<th>3rd Session (in computer lab)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administer StatKMA: it is the external measure of knowledge monitoring and measure of statistics knowledge.</td>
<td>Present tutorial on overhead projector to explain the system</td>
<td>Use prototype</td>
</tr>
<tr>
<td></td>
<td>Fill in biographical questionnaire</td>
<td>Group debriefing</td>
</tr>
<tr>
<td></td>
<td>Use of prototype to solve problem</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.2: Experimental Design of the Embedded Assessment System: sessions’ organization.

Monitoring Assessment scores thus confirming results from previous research (Tobias and Eversen, 2002). Interestingly the Knowledge Monitoring Assessment score did not correlate with the problem score. Moreover, contrary to their expectations, they found little relation between the knowledge monitoring assessment measured with the StatKMA and the online metacognitive measures. However, the relation between reflective prompts scores and the Knowledge Monitoring Assessment were stronger than those between the Knowledge Monitoring Assessment and the self-assessment scores. This result suggests that scoring responses to the reflective prompts may be a better reflection of knowledge monitoring skill compared with responses to the self-assessment task.

The online metacognitive scores were more closely associated with overall proficiency; i.e. the students who knew statistics well (looking at Statistics Score and Problem Score) tended to do well on the reflective prompts.
Table 3.3: Measures for the Embedded Assessment System: metacognitive and domain-related measures.

Hence, of the two hypotheses put forward none was sustained by the experiment.

The first hypothesis was that the explicit presentation of steps and sub-goals of the problem would prompt students to monitor and recheck earlier solution steps. Actually, from the 12 students, 11 followed the structured sequence imposed by the system, first elaborating a response for a solution step, then generating the reflective explanation and then moving on to the next step. Only six students engaged in editing behaviours, backtracking to earlier steps to make changes to their responses. This could suggest that these students engaged in reflective thinking, where the changes in their responses were as a result of responding to the reflective prompts. Unfortunately, this conclusion is not confirmed by their statistical analysis. In order to draw a conclusion it would be necessary in our opinion to ask these students to perform a later experiment where they had to solve another similar problem without the reflective prompts and observe their behaviour. If they did not replicate their editing behaviour, then the claim that the reflective prompts were the causal factors for the change of pattern would be strengthened.
The other hypothesis put forward was based on the idea that the use of help is indicative of a student’s knowledge monitoring ability and strategic action to bridge the knowledge gap. It was hypothesized that students with high metacognitive awareness would be sensitive to their lack of knowledge and so would seek help more consistently than their less aware counterparts. Surprisingly, they observed that students rarely used help even in situations when they were not accurately completing the solution. From the twelve participants, only five students accessed the help system, typically after entering a solution step but before responding to the corresponding reflective prompt. Thus, the reflective prompts did not trigger further help seeking. Instead, students used help to check their responses to a problem solving step. The researchers suggest that one explanation for that is that during the experiment students collaborated with each other as they worked on the system.

The lack of significant results was partially be attributed to the small sample size (N=12). One criticism that can be made to this study is that, besides the attempt of creating interesting measures for metacognition some designs decisions corroborate making the claims weak:

- the analysis was made based on only one problem which is not enough to observe metacognitive patterns of behaviour;
- students were not able to practice with the system before attempting their first problem;
- problems were not tailored to be challenging for each student, making the help and reflections less necessary; A Medium-low difficulty problem does not provide the trigger for metacognitive skills to take over.
- prompts were always visible preventing the researchers to observe when and if students look for further aid for understanding - similarly to what happened with the help system;
- participants should have been instructed to work individually on the problem.
- the StatKMA presented questions with content somewhat different than that specifically assessed by the prototype; also it was a multiple-choice test, whereas the system asked for written answers. Moreover, students did not replicate prediction actions in the system. How to compare this measure with the other measures in the system taking that the Knowledge Monitoring Assessment measures specifically the knowledge monitoring skill?

In conclusion, the inclusion of metacognitive prompts to elicit explanations is insufficient if the students are unaware of the expectations of the task. Moreover, the mere presence of the prompts is not enough to improve learning unless students are motivated to use them. Additional modelling might be necessary to facilitate the generation of appropriate explanations.

### 3.6 Conclusion

In this chapter we have presented a review of research on metacognition in ILEs, which is one of the contributions of this thesis. The review used a classification schema for the major issues concerning the design of metacognition instruction. Some representative examples of interactive learning environments that incorporate one or more components dedicated to metacognition instruction were offered, together with the description of experiments conducted with some of these ILEs that shade light to our own research.
As we have seen, only two of these have been developed with the specific goal of teaching metacognition: MIST and IQ Form. In all the other systems, researchers have sought improvements in performance in the domain or on cognitive skills and metacognitive training is only a means to that end. Only those two explicitly help students to learn about the process involved in learning and the role of metacognition on this process. On the other hand, the Embedded Assessment System tries to do something different from most others: to create online measures of metacognition from students’ behaviours. Unfortunately, it did not have much success.

Of all the systems reviewed, only SE-Coach uses what can be called a student model to direct the behaviour of the system with respect to metacognitive instruction; however, only the self-explanation ability is modelled. We believe that although it not an easy task, this is an effort that must be emulated. Such a model must be built taking as input any aspect of the student-system interaction from which information on the metacognitive level of the student can be extracted. Then, using the model, it becomes possible to adjust the quantity, timing, and content of the metacognitive training according to student’s needs.

Another challenge for ILEs is the design of adequate external representation to represent the “invisible” metacognitive skills the learner possess, such as monitoring and regulatory behaviours during learning tasks. The SE-Coach devised a clever way of modifying the user interface to detect which parts of the problem the student is looking at. Similar novel mechanisms to reify other subtle behaviours need to be designed.

In the next chapter we present the Reflective Assistant Model. It is a framework for explicitly fostering metacognitive skills that suggests specific timing, target level and approach metacognitive scaffolding in problem-solving learning environments for maths and includes a rule-based algorithm to assess students’ metacognitive state.
Chapter 4

The Reflection Assistant Model

4.1 Introduction

This research has been concerned with the creation of a reflective tool that explores new instructional designs for metacognition instruction in problem solving environments. It puts forth a model that contemplates specific metacognitive skills to be trained, proposes novel external representations of these skills, and defines diagnostic rules to assess students’ knowledge monitoring ability within ILEs. The model intends to enhance problem solving environments, reifying the metacognitive skills involved in such activities. This chapter defines the model proposed in this research, which is called the Reflection Assistant Model (or simply the RA).

Firstly, the chapter introduces the objectives of the RA. Then, it discusses the theoretical basis and the metacognitive skills modelled, and how the RA matches the domain of problem solving. The educational principles held by the RA Model are presented as well.

Next, a detailed description of the architecture and main components of the RA is offered, including the mechanics of the knowledge monitoring inference engine proposed and how the yielded results are applied to scaffolding the learning experience.

The chapter ends with a list of research questions and hypotheses put forward in this research and how we intend to investigate them.

4.2 The Objectives of the RA

The Reflection Assistant is a computational framework for metacognition instruction in problem solving ILEs. It aims at structuring the metacognitive training by defining: (i) specific metacognitive skills to be targeted, (ii) appropriate timings for this training, (iii) suitable approaches to promote the development of these skills, and (iv) mechanisms for assessing students’ metacognitive changes.

Hence, the RA intends to create appropriate conditions for students to think about themselves as problem solvers and reflect on their ongoing learning experience. This is achieved through reflective activities on those metacognitive skills considered important to attain success in problem solving.
Chapter 4. The Reflection Assistant Model

While the RA Model is dedicated to the metacognitive aspects of learning, it has been designed to be used in conjunction with problem-solving learning environments. So, it should not be used in isolation; instead it has to be coupled to some ILE, acting as an assistant to the learning process as shown in Figure 4.1. The goals of this new integrated environment are:

1. Elicit a connection in the student’s mind between her metacognitive skills and her domain-level actions, as well as the products or results she generates.

2. Emphasize the importance of having an accurate judgment of the understanding of the problems to be solved as a means to improve attention and to allocate cognitive resources appropriately.

3. Demonstrate to the student means to assess, reflect on, and improve her problem solving process.

![Figure 4.1: Integration of the RA Model with problem solving learning environments. From the user’s perspective it is an integrated environment. However, the Reflection Assistant is still detached from the problem solving ILE.](image)

The Reflection Assistant Model addresses mainly the following difficulties students normally exhibit in learning situations (Hartman, 2001a): a) determining the difficulty of a task; b) monitoring their problem comprehension effectively; c) planning ahead (the RA focuses on planning metacognitive strategies); and, d) monitoring the success of their performance and their metacognitive skills employed. The RA does not focus on some of the difficulties Hartman (2001a) describes, such as: planning ahead the time they need to perform a learning task; determining when they have studied enough to master the material to be learned; using all the relevant information;
using a systematic step-by-step approach; and selecting and employing appropriate representations (see the complete list of difficulties in section 2.5.1).

### 4.3 Theoretical Basis

We have adopted Tobias & Everson’s (2002) hierarchical model of metacognition (presented earlier in Section 2.3.3) as the theoretical foundation for the RA. Their model differs from other theoretical models because it supports the view that metacognition skills could be developed in an incremental way; this makes it more appropriate for training purposes. In their pyramidal model, *knowledge monitoring* is viewed as the most basic metacognitive skill, supporting the development of other metacognitive skills shown as upper layers of the model, such as evaluation of learning, selection of strategies, and planning.

Following Tobias & Everson’s formulation of an interdependent and hierarchic relation between metacognitive skills, the RA Model proposes an incremental focus on metacognitive skills in the same order they propose. Therefore, it is primarily directed to learners’ acquisition and improvement of knowledge monitoring skill. Supported by this, it focuses on evaluation of the learning process; then on top of those two, it works on the selection of metacognitive strategies. Figure 4.2 shows our adapted theoretical model and the desired effects of training the selected metacognitive skills in a computer-based learning environment. In the next section we elaborate on the skills trained within the RA Model.

Tobias & Everson have also developed an empirically validated instrument for measuring knowledge monitoring accuracy (from now on, we call it KMA). This instrument was adapted and augmented for the purposes of our computational model and is the basis for the continuous and automatic assessment of knowledge monitoring.

**Figure 4.2:** Theoretical view of metacognition in the RA Model. Adapted hierarchical model from Tobias & Everson with the metacognitive abilities that are emphasized in our research.
4.3.1 Metacognitive skills modelled

Our computational framework focuses on the following metacognitive skills: knowledge monitoring, evaluation of the learning experience, and selection of metacognitive strategies.

The RA does not include “planning”, located at the top of Tobias & Everson’s model, because we preferred to limit the scope of the framework. We can hope, however, that the training on those three fundamental skills would trigger the autonomous development of planning and other metacognitive skills.

Modelling knowledge monitoring

Scaffolding knowledge monitoring is the prime target of the RA Model. Knowledge monitoring is the ability to assess one’s knowledge or, by extension, one’s understanding.

The RA is built on the assumption that promoting awareness of one’s level of knowledge monitoring accuracy is the first step to foster this skill. Improvements in knowledge monitoring ability in turn triggers selective attention and facilitates better allocation of cognitive resources. That is, having identified what one does not know but should know, help to shift one’s attention to it. In a problem solving scenario, knowledge monitoring targets the understanding of the givens and the goals of the problem. An effective assessment of one’s understanding of these elements may lead to further readings of the problem description for a deeper understanding of what exactly is being asked and what information is provided in the problem.

Modelling evaluation of learning

As seen in Chapter 2, research on problem solving supports the idea that when the student steps back and reflects on how she actually solved a problem, she is putting attention directly on her own thought process (Brown, 1985). Thus, the activities proposed in our model aim to develop awareness of how one behaves during problem solving, which resources one uses, how long one spends on each task, and which decisions one makes in the process of solving a problem.

As a result it is expected that the student will make better plans in future learning interactions, based on conscious knowledge of what works best for her and what could be improved. For example, the student could approach future problems knowing that her habit of taking her time reading the problem is a good thing, while she needs to spend more time checking the solution once it is shown to her.

Modelling selection of metacognitive strategies

Finally at the top level of our model comes the ability to select good strategies to be applied during the learning process. The focus here are the metacognitive strategies, i.e. general heuristics or strategies that are loosely connected to the domain and the task. The RA Model focuses on developing students’ awareness of three kinds of metacognitive strategies: strategies for monitoring understanding, strategies for monitoring the problem solving process and controlling errors, and strategies for revising.

4.3.2 Domain of application: ILEs for problem solving

The RA is designed to be used in conjunction with problem solving learning environments. Thus, it needs to integrate quite tightly with this kind of environment in order to adapt the learning process, create new meaningful activities and learning targets.
As presented previously in Section 2.8, the problem solving activity can be divided into three conceptual stages: (a) preparation to solve the problem or familiarization stage, (b) production stage, and (c) evaluation or judgement stage. Therefore, the RA Model is organized around these stages, matching specific metacognition instruction to the characteristics of each of these stages. Figure 4.3 shows a high level diagram of the association of the RA Model with the problem solving stages.

Figure 4.3: Problem solving conceptual stages and the RA Model. Two new stages are proposed: pre-task reflection and post-task reflection.

At the top of the diagram is a timeline representing a typical problem solving episode broken down into conceptual stages. The layer in the middle of the diagram shows the cognitive skills that are brought to bear on the process as time goes by. The layer at the bottom represents the metacognitive skills which are active along the way.

Considering these conceptual stages, we have set specific moments where each of the selected metacognitive skills shall be trained. As such, knowledge monitoring and selection of metacognitive strategies (skills defined in Section 4.3.1) are mainly explored in the familiarization stage, when the learner should naturally spend some time understanding the nature of the problem, recognizing the goals, and identifying the givens of the problem.

We believe that cognitive load is higher at the production stage. Moreover, as observed in previous empirical studies (Rutz et al., 1999, for example) knowing when to interrupt students during problem solving, and having a good match between the content of the reflection and the current task are key to make them reflect. Unfortunately that kind of match is impossible to do given that we will not have access to the thoughts of the student.

Therefore, the design of the RA Model does not include any major interference during the production stage. Instead, two new conceptual stages in problem solving were created, which are called pre-task reflection and post-task reflection.
These stages can be seen as extensions of, respectively, the familiarization stage and the evaluation stage (look again at the timeline depicted in the diagram). At these new stages the cognitive load is lower, because the student is not engaged in actually solving the problem, but still has the problem fresh on her mind (in the case of pre-task reflection, this will be a problem solved earlier).

Thus, the RA Model uses this “cognitive space” to promote reflection on knowledge monitoring and evaluation of learning, in this case of the problem solving experience (see definition of evaluation of learning in Section 4.3.1).

As such, the metacognitive skills come to the foreground, becoming the object of reflection.

**Pre-task reflection stage**

Self-reflection is normally regarded as an after task activity. In the RA self-reflection also happens before the learning task. It covers the metacognitive aspects necessary to start the new problem; it provides suitable conditions for the student to become aware of useful general strategies, resources available, and also the degree of attention necessary to succeed in solving the problem.

Collins and Brown (1988) affirm that sometimes self-reflection does not occur naturally, in part because students are not really motivated to perform reflection after solving the problem and in part because the current problem solving medium (paper and pencil) does not really lend itself to this activity. By placing self-reflection before the learning activity, we expect to reduce problems of low motivation.

**Post-task reflection stage**

Self-reflection after problem solving has a different goal. It involves mainly the most recently problem attempted, but it is also linked to the student’s reflections before the beginning of the problem. In other words, the post-task reflection stage creates a space where the student thinks about her actions during the past problem solving activity, comparing them with her reflections expressed just before the beginning of that problem. Using the task in hand the student abstracts her thoughts to a metacognitive level, thinking about herself as a problem solver and about her general learning processes. The goal is to trigger self-questioning about the learning experience such as: *Did I use all the problem solving resources available? How well did I manage my time? Did I spend enough time reading the problem and making a suitable plan of action? What could I have done differently?*

**Exemplifying the matching of the RA Model with a problem solving activity**

We regard maths problem solving environments as the main candidate for adopting the RA Model. Accurate knowledge monitoring, attentive evaluation of learning and a good selection of metacognitive strategies play important roles in helping students to become good maths problem solvers.

Understanding mathematical text problems is a complex and knowledge-intensive inferential and highly constructive process that requires skillful interaction of more than one kind of knowledge, including linguistic, situational, as well as mathematical knowledge (Reusser, 1993). Knowledge monitoring provides awareness of problem comprehension and it is the first step towards the search for a deeper understanding.

Polya (1945) states that progress toward a maths problem goal is incremental and each move should be informed by a repertoire of strategies and heuristics. Furthermore, Schoenfeld (1992) says that heuristics should be taught in the context of a metacognitive strategy that supports the
learner in selecting the right one to solve a problem. We believe that being aware of the heuristics one possesses and consciously applying them is key to improve learning. Thus, the selection of metacognitive strategies in the RA follows Polya and Schoenfeld’s principles to improve heuristic search.

Finally, research on maths problem solving affirms that it is important to reflect on the steps taken in order to evaluate the use of cognitive and instructional resources. Reflection on adequate allocation of time, effort, usage of prior knowledge and heuristics learnt are important objects of evaluation of the problem solving experience. Promoting awareness on the whole experience makes students realize that solving a problem is not a linear activity, but a complex knowledge building endeavour.

4.4 Design of the Reflection Assistant Model

One possible general view of the RA Model is through the classification schema we have introduced in Chapter 3. The dimensions are: timing, that refers to the moment chosen for promoting training on metacognitive skills; target level, the refers to the degree of connection of the metacognitive skills trained with the domain; and instructional approach, which refers to the representation mechanisms chosen to provide the training. The possible values for this classification were presented before and are summarised in Figure 3.1. According to this classification, the RA Model can be categorize as shown in Figure 4.4 below.

![Figure 4.4: The RA in our classification of metacognitive training.](image)

The RA proposes reflective activities both before and after the problem solving task, and during the familiarization stage of problem solving, but not during the production and evaluation stages. The target level is on domain-related skills (skills that are relevant to problem solving in general) and task-specific skills (skills that are relevant to solve the specific problems presented). Three instructional approaches were adopted: graphical reification combined with metacognitive scaffolding, and self-assessment.

The RA combines the elements in these dimensions in different ways, depending on the situation. For example, the reflective activity related to knowledge monitoring is proposed before the problem solving task and uses graphical reification of student’s assessment of her understanding.
of previous problems performed, together with metacognitive scaffolding messages.

4.4.1 The Architecture of the RA Model

The RA is kept as general as possible so that it can be adapted according to the specific domain and ILE to which it will be incorporated. Figure 4.5 presents the general architecture of the Reflection Assistant Model. Similarly to Figure 4.1 shown before, this diagram depicts the communication of the RA Model with the problem solving learning environment and the interaction of the user with both environments.

The RA is divided into two main modules, which are called: pre-task reflection and familiarization assistant and post-task reflection assistant. Students interact with the activities proposed within the modules, which aim to provide structured support for metacognition development.

The pre-task reflection and familiarization assistant is used in the newly created pre-task reflection stage and also in the familiarization stage of problem solving. The aim of this module is to prepare the student for the problem solving activity, promoting reflection on knowledge monitoring, assessment of the difficulty and understanding of the problem to be attempted, and awareness of useful metacognitive strategies.

The activities labelled as “Comparison of knowledge monitoring and performance” and “Analysis of knowledge monitoring state (KMA/KMB)” are not related to a specific problem; rather they refer to the overall experience of the student with the problem solving environment, and uses information from previous interactions with the RA Model and previous problems performed. The
activities labelled as “Assessment of problem comprehension and difficulty” and “Selection of metacognitive strategies” are related to the current problem the student is about to attempt. They take place in the familiarization stage of problem solving, just before the student starts to actually solving the problem.

The post-task reflection assistant presents activities related to the evaluation of problem solving. As the name suggests, it takes place in the post-task reflection stage, just after the student finishes a problem. It involves two activities labelled “Evaluation of problem solving experience” and “Self-assessment of strategies usage”. After concluding a problem the student interacts with this assistant to analyse her performance, her use of problem solving resources, time management, and use of strategies.

Besides the modules, the RA includes an inference engine to assess students’ level of metacognition. The engine is an algorithm that uses formulas and rules to infer knowledge monitoring ability. The engine uses information from students’ interaction with the problem solving environment and from the RA Model and generates two measures: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB). These concepts and their measures will be presented in detail in the next section.

Finally, the RA incorporates a series of data repositories. The “library of metacognitive feedback” and the “library of metacognitive strategies” contain general knowledge about metacognition, and the “student KMA/KMB model”, the “student’s metacognitive strategies”, and the “history of student’s reflective actions” are repositories of information about students’ demonstrated or inferred metacognition.

As seen in the architecture diagram, the communication between the RA Model and the ILE is through two data repositories in the ILE side: “student’s problem solving performance” and “history of student’s problem solving actions”. So, even detached from the problem solving environment, the RA needs information about students’ actions that comes from the ILE. Thus, in order to use the RA Model it is essential that the ILE implements these repositories. In the section about the repositories we detail all pieces of information that have to be provided for both the RA and ILE.

From the user’s perspective, the RA Model may appear either separated from the problem solving environment (e.g. in specific and separated windows) or it can be embedded in the ILE. Independently of the choice, the interaction takes place in the following sequence: (1) the student starts by performing the first two activities proposed in the pre-task and familiarization assistant, then the ILE presents a new problem and she proceeds to the other two activities of the the pre-task and familiarization assistant; (2) she solves the problem with the aid of the problem solving tools provided by the ILE; and (3) after finishing the problem, she performs the activities proposed by the post-task reflection assistant.

4.4.2 Metacognitive inference engine

The RA Model incorporates an assessment mechanism of students’ metacognition and uses it to maintain a student metacognitive model. It is a simple inference engine, since it infers only one metacognitive skill: knowledge monitoring.

The student’s knowledge monitoring ability is inferred from the ILE (using the student’s per-
formance on problem solving) and from the RA (using the student’s own prediction of her understanding of problems and ability to solve them).

The inference mechanism is applied every time the student attempts a new problem and the student model is updated as a result. The information comprised in the student model is then used in the pre-task reflection activities provided by the RA.

Two aspects of knowledge monitoring ability are inferred: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB).

The Knowledge Monitoring Accuracy (KMA) refers to how skillful a student is at predicting how she will perform on a learning task; it reflects her awareness of the knowledge she possesses.

The Knowledge Monitoring Bias (KMB) provides a statistical measure of any tendency or bias in the learner knowledge monitoring ability.

The assessment mechanism of these measures and rules for classifying students’ KMA and KMB scores are shown next.

4.4.3 Definition of the Knowledge Monitoring Accuracy (KMA) measure

We have developed a formula for measuring knowledge monitoring accuracy (KMA) based on Tobias & Everson’s assessment instrument (presented in Section 2.4.2). Their original instrument evaluates the student’s knowledge monitoring ability by first asking her whether she is able to solve a problem and later asking her to solve that problem. The KMA results from the match between these two pieces of information.

By collecting a significant number of elementary assessments for the same student, their instrument produces a statistical profile of the student’s awareness of her own knowledge. The scale of possible scores for KMA varies between -1 and 1, where -1 signifies very low accuracy of knowledge monitoring, and 1 means complete accuracy. Values around 0 signify a random assessment; in other words, that the student failed in her assessment as often as she succeeded. They present the following formula for scoring the final KMA:

\[
\frac{(a+d) - (b+c)}{a+b+c+d}
\]

where \(a\) is the number of times the student predicted success and succeeded, \(b\) is the number of times the student predicted failure and succeeded, \(c\) is the number of times the student predicted success and failed, and \(d\) is the number of times the student predicted failure and failed (for further explanation, please refer back to Section 2.4.2).

To build our KMA measure we reinterpreted this formula as the mean of individual scores evaluated for each prediction/performance cycle. Each such score, which we call \(kma\), represents the correctness of the prediction on that particular cycle.

As there are two possible values for the evaluation and two for the performance, \(kma\) has to be defined for a total of four cases. Tobias & Everson’s formula is obtained by giving \(kma\) the following values:

<table>
<thead>
<tr>
<th>Performance</th>
<th>Estimation</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>correct</td>
<td>will solve</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>will not solve</td>
<td>-1</td>
</tr>
<tr>
<td>not correct</td>
<td>will solve</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>will not solve</td>
<td>1</td>
</tr>
</tbody>
</table>
We see that $kma$ is 1 when the assessment is correct (i.e. predicted that was not able to solve the problem and indeed did not solve it correctly, or predicted that was able to solve the problem and solved it correctly) and -1 when the assessment is incorrect (i.e. predicted that was able to solve the problem, but did not solve it correctly, or predicted that was not able to solve the problem, but solved it correctly).

If we average $kma$ over a series of problems, we end up with the KMA score as proposed by Tobias & Everson.

Seeing this formula as an average, we notice that in its original form, the whole KMA scheme is a binary one. Prediction and performance are binary and the variable $kma$ also ends up taking only two distinct values. We wanted to have a bit more flexibility and be able to differentiate between students who did not solve the problem correctly at all from those who presented a partially correct solution path. In order to do that, the minimum change we could do was to allow both prediction and performance to take a third value representing some intermediary state.

**Defining our new measure of KMA**

In the dimension of prediction, we added the possibility for the student to predict that she would partially solve the problem or that she partially understood it. In the dimension of performance, we now treat partially correct answers as a meaningful case. The 2x2 matrix needs to become a 3x3 matrix to accommodate all possible combinations as shown in Table 4.1.

We then had to decide which values $kma$ would take for each of the 9 cases so defined. The four corners of the new matrix are equivalent to the original four cases of Tobias & Everson’s table. It was thus natural to use the same values for them.

The point in the middle is similar to the top left and bottom right corners in that the prediction and the performance are matched. It therefore makes sense to assign 1 to $kma$ in this case as well.

All the other cases represent mismatches between what was predicted and what happened. However these mismatches are not as large as the ones corresponding to the bottom left and top right corners of the matrix. For instance, when the student predicts that she will solve the problem partially and solves it fully, this is a knowledge monitoring error. However it is not as bad an error as if she had predicted that she would not solve the problem at all. It therefore makes sense to associate with such mistakes a $kma$ value larger than -1, which represents the full mistake.

At first glance, a good candidate would be 0 which is just half way between -1 (being completely wrong) and 1 (being completely right). However, doing so would mean that students that always make such small mistakes in their assessments remain with a null value of KMA. Only one correct assessment would then push their KMA to a positive value. It seemed more appropriate to have two such small mistakes cancel out one accurate assessment. For that to happen, we decided to assign -0.5 to all those 4 cases.

Thus, one requisite for the ILE is that it analyzes and breaks down the student’s performance into the three possible values: correct, partially correct, or incorrect. Our metacognitive inference engine will need such a categorization to function. Similarly, the RA has to gather estimates of performance with the three possible values: able to solve completely, able to solve partially, not able to solve the problem.

The mean of $kma$ scores over all problems solved so far yields the current KMA state of the student. The more the student interacts with the ILE and RA, the more reliable becomes the RA’s
Chapter 4. The Reflection Assistant Model

4.4.3 Prediction → Performance

<table>
<thead>
<tr>
<th>Prediction → Performance</th>
<th>estimated will solve</th>
<th>estimated will not solve</th>
<th>estimated will solve partially</th>
</tr>
</thead>
<tbody>
<tr>
<td>demonstrated not correct</td>
<td>1</td>
<td>-0.5</td>
<td>-1</td>
</tr>
<tr>
<td>demonstrated partially correct</td>
<td>-0.5</td>
<td>1</td>
<td>-0.5</td>
</tr>
<tr>
<td>demonstrated correct</td>
<td>-1</td>
<td>-0.5</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4.1: Matrix of values for Knowledge Monitoring Accuracy (KMA).

assessment of her KMA.

Students’ Classification of KMA

The score inferred for the KMA is shown to students in the reflective activities. For this purpose the numeric values are converted into qualitative ones. The classification summarises scores by mapping them in three categories: low KMA, average KMA, and high KMA. Table 4.2 shows the exact ranges of this mapping. So, if a student has a KMA of 0.8, she will be assigned a high KMA profile, which means that most of the time she demonstrates accurate estimations of her knowledge.

Notice that we have voluntarily made the high range smaller than the lower ranges, because we decided that we would acknowledge as high KMA only those students with a very consistent accurate behaviour.

<table>
<thead>
<tr>
<th>KMA Value</th>
<th>Classification</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-1, -0.25)</td>
<td>Low KMA</td>
<td>Learner doesn’t estimate correctly her knowledge in the majority of situations</td>
</tr>
<tr>
<td>[-0.25, 0.5)</td>
<td>Average KMA</td>
<td>Learner sometimes estimates correctly, but makes frequent slightly wrong or completely wrong estimations</td>
</tr>
<tr>
<td>[0.5, 1]</td>
<td>High KMA</td>
<td>Learner most of the time makes correct estimation of her knowledge</td>
</tr>
</tbody>
</table>

Table 4.2: KMA classification. Students’ categories according to their KMA scores.

4.4.4 Definition of the Knowledge Monitoring Bias (KMB) measure

The Knowledge Monitoring Bias (KMB) measure was created since the KMA does not provide a detailed account about the type of inaccuracies the student may show. For example, imagine a student that was assigned a low KMA profile, because she is constantly predicting that she will solve the problems correctly, but her problem solutions are invariably wrong. This case is different from the one of another student that tends to estimate that she will not solve the problems
completely correct, but then most of the time she reaches a correct solution. So, in order to describe the pattern or bias in the learner’s knowledge monitoring ability we developed the KMB measure.

The KMB takes into account the way student deviate from an accurate assessment of her knowledge monitoring. If there is no deviation, we say that the student is accurate in her assessment of her knowledge or realistic about it. Otherwise, three cases are possible: (i) the student often predicts she will solve the problems but she does not succeed, demonstrating through this an optimistic assessment of her knowledge; (ii) the student often predicts she will not solve the problems, but then she succeeds in solving them, demonstrating through this a pessimistic assessment of her knowledge; and (iii) she is sometimes optimistic in her assessment of her knowledge as she is pessimistic, in a random pattern.

The KMB scores are dependent of the KMA demonstrated. So, the possible range of values for the KMB is also from -1 to 1. Similarly to the KMA, a 3x3 matrix was also created for providing the possible scores for each individual kmb (see Table 4.3). The possible kmb scores have a different meaning from the kma. Accurate estimations receive the score 0, meaning a null bias; 1 is assigned to the highest optimistic estimation; likewise -1 is assigned to the highest pessimistic estimation; -0.5 is assigned to a minor pessimistic estimation and 0.5 to a minor optimistic estimation.

<table>
<thead>
<tr>
<th>Prediction → Performance</th>
<th>estimated will not solve</th>
<th>estimated will solve partially</th>
<th>estimated will solve</th>
</tr>
</thead>
<tbody>
<tr>
<td>demonstrated not correct</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>demonstrated partially correct</td>
<td>-0.5</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>demonstrated correct</td>
<td>-1</td>
<td>-0.5</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4.3: Matrix of values for Knowledge Monitoring Bias (KMB).

Students’ Classification of KMB

A classification of student’s current KMB state in respect to her predictions deviations is made based on the mean of kmb scores over all problems solved so far.

As the score inferred for the KMB is used in the reflective activities of the RA Model, a classification that converts the numeric values into qualitative ones was defined to make it easier to the student to understand her bias. The classification summarises scores by mapping them in four categories: realistic, optimistic, pessimistic, and random. Table 4.4 shows the exact ranges of this mapping.

So, if the learner was assigned a value that represents a “high KMA”, she is then classified as realistic and it means that she has no bias in her judgement of her knowledge. However, if she has obtained an “average KMA” or a “low KMA”, it means that she has some kind of bias in her judgement and a KMB value is calculated. For these cases, the KMB value is matched to the
corresponding interval. For example, if the student has a low KMA (meaning a low accuracy of her knowledge monitoring) and has a KMB equals to - 0.7, then it is said that the student tends to have a pessimistic assessment of her knowledge, often saying that she is not able to solve the problems, but actually succeeding in solving them or partially solving them.

<table>
<thead>
<tr>
<th>KMB Value</th>
<th>Classification</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>High KMA</td>
<td>Realistic</td>
<td>Learner makes an accurate estimation of her knowledge, having a High KMA</td>
</tr>
<tr>
<td>[0.25, 1]</td>
<td>Optimistic</td>
<td>Learner tends to estimate she can solve the problems, but she does not succeed in the majority of situations</td>
</tr>
<tr>
<td>[-1, -0.25]</td>
<td>Pessimistic</td>
<td>Learner tends to estimate she can not solve the problems, but then she succeeds</td>
</tr>
<tr>
<td>(-0.25, 0.25)</td>
<td>Random</td>
<td>Learner estimates of her knowledge are as often optimistic as pessimistic</td>
</tr>
</tbody>
</table>

Table 4.4: KMB classification. Students’ categories according to their KMB.

By exhibiting to the student her KMA and KMB levels the RA intends to promote awareness and development of knowledge monitoring skill.

4.4.5 Data repositories and activities

This section presents the other components of the RA: data repositories and activities within modules. The description of the activities was organized into the following rubrics: what: shows the objectives of the activity; how: presents the approach devised with the interface mock-ups suggested; input/output: presents the input data used and the output data generated by the activity.

The reflective activities use graphical representations associated to textual explanations to convey information about metacognitive skills and the learning processes. The diagrams and graphs proposed are believed to transform the reflective activities into easier, perceptual ones. Considerable research has taken place on the different mechanisms through which graphical representations assist users in drawing inferences, solving problems, comprehending computer programs, etc. (Larkin and Simon, 1987; Tufte, 1990; Cox, 1999). In general it has been accepted that a graphical method of representing information and solution improves performance, individual understanding, and solution quality (Larkin and Simon, 1987). Hence, we use graphical representations of the learners’ metacognitive abilities and learning processes to help them see patterns, express abstractions in concrete form, and discover new relationships.

Data repositories

A list of all pieces of information that is stored in the RA repositories are presented in the Table 4.5. The data is presented in a general degree of detail.

Table 4.6 enumerates the repositories that have to be created in the ILE in order to provide information about students’ actions and performance to the RA.
**Student KMA/KMB Model**

For each problem:
- Assessment of the difficulty to perform it
- Prediction of solving it correctly
- KMA score calculated after problem performed
- KMB score calculated after problem performed
- Student’s reflections on her metacognitive progress and tendency

**Student’s Metacognitive Strategies**

For each problem:
- List of strategies selected (they can be created or edited from the library of metacognitive strategies)
- Student’s reflections on the use of the strategies selected

**Library of Metacognitive Feedback**

- Texts with instructional scaffolding messages related to each metacognitive value for the KMA and KMB
- Texts with instructional scaffolding messages related to strategies selection
- Texts with instructional scaffolding messages related to evaluation of the learning experience

**Library of Metacognitive Strategies**

- List of strategies for monitoring understanding
- List of strategies for controlling errors
- List of strategies for revising

**History of Student’s Reflective Actions**

For each problem, time spent on reflective activities both before and after the problem

| **Table 4.5: Data repositories of the RA Model.** |

---

**The RA reflective activities**

**Activity 1: Comparison of knowledge monitoring and performance**

**What:** This activity aims to trigger reflection on student’s trend and progress of her knowledge monitoring ability. It focuses on past problems, showing to the student her previous performance and comparing them to her judgements of her knowledge and understanding of those problems.

**How:** It replays to the student her previous demonstrated performance (summarised into the following categories: low, average, and high) for all past problems solved together with the her prediction of solving those problems (also summarised into: low, average, and high). It uses a graphical reification such that the student can compare the two pieces of information. Figure 4.6 shows the mock-up made for this graphical reification; it depicts bar-type graphs showing the assessment of problem understanding next to the performance for all past problems. Beside the graphical display, textual explanations are necessary to provoke appropriate interpretations of
Chapter 4. The Reflection Assistant Model

### Student’s Problem Solving Actions
For each problem, whether the solution given was correct, partially correct or not correct.

### History of Student’s Problem Solving Actions
For each problem, the time spent on each problem solving resource (date and time of the beginning and date and time of the end).

**Table 4.6: Data repositories provided by the ILE.**

The graph, asking students to look for trends, to check for improvements, and to self-explain the reasons for the differences shown (if they exist).

Larkin and Simon (1987) describe the benefits of using both diagrams and text to describe a scene, a 3-dimensional object, etc. We believe that a combination of visualizations and text is a fruitful way to facilitate learning of metacognition. For example, the text would be helpful at a very early point when a student is just learning to interpret the graphical representation of her predicted and demonstrated performance and infer patterns from that.

![Graphical reification of the comparison of predicted knowledge and performance](image)

**Figure 4.6: Mock-up of the graphic component of the first pre-reflection activity. Graphical reification of the comparison of predicted knowledge and performance.**

**Input:**
- Prediction of solving each problem correctly.
- Performance in problem solving.

**Output:**
- Student’s reflections on her metacognitive progress and tendency.
- Time spent on the reflective activity.
- Identification of the reflective activity.
**Activity 2: Analysis of knowledge monitoring state (KMA / KMB)**

**What:** This activity also focuses on knowledge monitoring ability. Differently from the previous activity, it does not show past problems performance; it refers to problem solving in general and presents the RA's inferred assessment of student’s level of KMA and KMB. In this way, it aims to foster accuracy of knowledge monitoring by raising awareness of and initiate discussion on knowledge monitoring ability overtly.

**How:** The values calculated for the KMA and KMB by the inference engine are converted into degrees (KMA= high, average, low / KMB= realistic, random, optimistic, pessimistic) and a visual display shows these levels in a coloured and easy to grasp manner. We have devised graphical widgets similar in form to a car speedometer and which we call **reflectometers**.

Similarly to the “skillometers” of Anderson et al. (1992) that were designed to present learners with a general view of the state of their knowledge of the domain, the reflectometers present learners with a view of their knowledge monitoring ability.

Figure 4.7 shows the mock-up made for the reflectometers and other possible interface functionalities. The first reflectometer indicates the accuracy of the student estimate of her own knowledge (KMA) and the second shows any bias that exists in that estimation (KMB).

![Figure 4.7: Mock-up of graphic component of the second pre-reflection activity. Graphical reification for the analysis of knowledge monitoring state.](image)

Textual explanations and scaffolding are very important in this activity, since the student needs some help to interpret the information provided in the reflectometers and guidance to adjust her future problem solving actions to improve her knowledge monitoring scores. The scaffolding should be tailored to the values shown and also based on previous actions the student has taken (e.g. if the time spent on assessing problem comprehension is generally low, one possible instructional scaffold could be “try to spend more time reading the problem and carefully evaluating your understanding of the concepts involved, before stating your understanding of the problem”).

**Input:**

- mean of the KMA using all KMA scores calculated after each problem performed.
- mean of the KMB using all KMB scores calculated after each problem performed.
• Texts with instructional scaffolding messages related to the value of the KMA/KMB.

Output:

• Time spent on the reflective activity.
• Identification of the reflective activity.

**Activity 3: Self-assessment of problem comprehension and difficulty**

**What:** This activity is related to the self-assessment of the current problem still to be solved. It aims to make the student reflect on her understanding of the concepts and components of the problem and on her confidence to solve it correctly. It also gathers information that will be used later on the future pre-reflective activities (Activities 1 and 2).

**How:** Questions about the student’s level of comprehension of the problem and difficulty to solve it are presented. Discrete scales should be used. The number of questions depends on the domain and level of detail it will be necessary to have a good picture of the student’s understanding of the problem. Examples of questions are: *Do you recognize the type of the problem? Do you understand the goals of the problem? Have you done a similar problem before?* etc. It is important that the student provides all the information requested prior to starting to attempt to solve the problem. There is no right or wrong answer; the answers must reflect the student’s judgement of her understanding.

**Input:** This activity does not use any input data from the RA Model. It only uses the problem description that is provided by the problem solving ILE.

**Output:**

• Assessment of the difficulty to perform the current problem.
• Prediction of solving the current problem.
• Time spent on the reflective activity.
• Identification of the reflective activity.

**Activity 4: Selection of metacognitive strategies**

**What:** The goal of this activity is to make students reflect on strategies that can be useful to solve the problem at hand. It focuses on metacognitive strategies related to the three metacognitive processes that should take place during problem solving: monitoring understanding, monitoring progress and controlling errors, and revising solution paths. Thus, this activity helps students to think of relevant strategies, their purposes and appropriate moments to apply them.

**How:** Lists of pre-defined strategies are presented to the students and they are asked to select those they think may help them to solve the current problem. Besides that, students should be allowed to create their own strategies and also to edit the existing ones. Extra functionalities can be incorporated, such as ordering the strategies by importance or relevance, identification of strategies that are always used for all problems, etc.
The RA does not interfere with the problem solving stage. However, one possible option is to present the list of selected and created strategies in a separate window during problem solving. This will create minimal interference and may serve as a reminder to the student or a trigger to make them apply those strategies. Another possibility is to offer the list whenever the student asks for help. The designer is free to implement the solution that appears to be more convenient to the domain and goals of the problem solving environment.

**Input:**

The activity uses as input data the information in the library of metacognitive strategies. The library contains specific strategies, organized into three groups: strategies for monitoring understanding, strategies for controlling errors, and strategies for revising. Below we suggest some strategies to be included in the library. They were taken or adapted from the literature on metacognition assessment (Schraw and Dennison, 1994; Pintrich et al., 1993).

- **Strategies for Monitoring Understanding:**
  1. Read the problem more than once.
  2. Read the problem to separate important parts or identify components.
  3. Think of a related problem you have already done and use it as a model.
  4. Before starting to solve the problem, think what you are supposed to learn from it.
  5. Read the problem and determine which parts you don’t understand well.
  6. Review the basic concepts that are not clear before attacking the problem.
  7. Set a goal to yourself and plan the steps to reach this goal.

- **Strategies for Controlling Errors:**
  1. Stop and review each step to see if you made a mistake.
  2. Reread the problem from time to time to check for forgotten important parts.
  3. Stop and change strategies if you get lost and confused and don’t seem to move anywhere.

- **Strategies for Revising:**
  1. Think about a way of checking to see if your solution is correct.
  2. Review all you did to make sure you are not forgetting anything.
  3. Reread the task description and ask yourself if your solution really meets the task goal.

**Output:**

- Strategies selected, edited or created by the student for the problem.
- Time spent on the reflective activity.
- Identification of the reflective activity.

**Activity 5: Evaluation of problem solving experience**

**What:** Similarly to the reflection-on-action proposed by Schön (1987), this activity aims to give an opportunity to the student to review her most recent experience, exploring why she acted as
she did, what happened during problem solving, etc. The focus is on helping her to reflect on the “causes of her mistakes” related to the process, her use of resources, and time management issues. In so doing she can develop a better understanding of her problem solving experience and practice.

**How:** We have developed a graphical reification of the student’s interaction with the problem solving activity. Figure 4.8 shows the mock-up for the graphical component of the activity. The graphical reification proposed was inspired in Schoenfeld’s timeline graph originally created for analysing students’ problem solving behaviours (as seen in Figure 2.5). We envisioned a timeline graph that shows the problem solving activities and the reflective activities.

![Graphical Reification](image)

**Figure 4.8:** Mock-up of graphical reification for post-task reflective activity. Evaluation of problem solving experience.

Looking at the graph the student can observe when (and if) she used the main problem solving resources available, the number of times the resources were used for a given problem, the time spent on reflective activities, etc. Another possible option is to show the moments when she asked for help. A textual explanation of the information depicted by the graph is also needed to trigger reflection on the main “conflicts” that may exist on the student’s performance (e.g. the student said the problem was difficult, but then she did not use any of the resources provided by the problem solving environment to help on solving the problem, or she spent little time selecting strategies to solve the problem, etc.). Finally, it could also be beneficial to provide a comparison of the student’s performance in the last problem solved to the one of a more experienced person for the same problem, or to compare the problem to a previous similar problem done by the student. Figure 4.9 presents an initial mock-up for this last option.

**Input:**

- History of student’s reflective actions (duration of each activity).
- History of student’s problem solving actions (duration of each activity).
- Student’s problem solving performance.
Assessment of difficulty to perform each problem.

- Student’s prediction of solving each problem.
- Metacognitive texts - feedback.

**Output:**

- Time spent on the reflective activity.
- Identification of the reflective activity.

**Activity 6: Self-assessment of strategies usage**

**What:** The goal of this activity is also to reflect on the most recent problem solving experience, but this time the focus of the reflection are the metacognitive strategies the student planned to use. The activity helps the student to review the experience from this point of view and to analyse whether she applied or not a strategic behaviour during problem solving.

**How:** The list of metacognitive strategies selected or created by the student at the beginning of the problem is presented again and some reflective questions are posed concerning the use of those strategies during problem solving. The questions serve as a trigger for making the student self-explain if and when she used the strategies and whether they were useful or not during problem solving. Questions like: Did you use the strategies during problem solving? Which ones were most useful? When did you use them? What would you do differently next time? are helpful to guide student’s self-explanations.

**Input:**

- Strategies selected, edited or created by the student for the problem.
Output:

- Student’s reflections on the use of the strategies selected for the problem.
- Time spent on the reflective activity.
- Identification of the reflective activity.

4.4.6 Adding the RA to an existing ILE

The Reflection Assistant model is a generic framework whose principles should be applicable to any kind of problem solving ILE.

In order to facilitate integration with existing ILEs, we have minimized the coupling between the RA and the ILE. Information flows only one way from the ILE to the RA. As shown on Figure 4.5, the RA only needs access to two well defined data repositories: one keeps information on the student’s performance, the other on the history of the interaction between the student and the resources of the problem solving environment. If the ILE keeps track of the interaction with the student in some form, it ought to be easy to add a wrapper around the existing datastore to get it to conform to the interface needed by the RA. More work will be needed in the case where the ILE doesn’t keep any history of its interaction with the student. In such case, this datastore will have to be built from the ground up.

In terms of user interface, the mock-ups presented are only a guideline for the final graphical components. They would have to be adapted so that the phrasing of messages is relevant to the ILE. Whether they are included in existing windows or included in new additional windows will also have to be decided on an ad-hoc basis.

4.5 Research Questions and Hypothesis

The RA model is based on the notion that focusing on metacognitive skills as object of reflection triggers the development of these skills and has the potential to improve learning. Hence, it intends to make students aware of the importance of metacognition for the success of the learning endeavour.

More specifically, the RA promotes the idea that by reflecting on one’s own knowledge monitoring skills, one increases one’s own understanding of the influence of these skills in the outcomes of the problem solving experience.

Therefore, our main research question can be expressed as: Does the Reflection Assistant Model help students to improve their metacognition? And, if so, does it lead to improvements in problem solving performance?

We expect that the interaction with the reflective activities proposed by the RA encourages students to become more conscious about their learning processes and skills. Thus, the hypotheses driving this research are that reflection on the metacognitive skills between problem-solving learning activities has the potential to:

(i) enhance students’ knowledge monitoring skills and in turn this helps them to become more aware of how to allocate their cognitive resources to solve problems,

(ii) encourage students to think of appropriate metacognitive strategies for problem solving,
motivate students to monitor the use of these metacognitive strategies during problem solving, and

boost students’ performance on the topic in hand.

In order to test these hypotheses we needed to implement the model within a problem solving ILE. Instead of using an existing environment, we decided to build an ILE for algebra word problems and incorporate the RA Model in it. We called this system MIRA (Metacognitive Instruction using a Reflection Approach). All the activities, data repositories, and inference engine proposed in the RA were implemented.

To investigate the power of this new integrated environment (MIRA with RA) we conducted an experiment. A controlled study was devised and the data it provided was used to draw conclusions about the effectiveness of the RA Model for metacognition training.

4.6 Conclusion

In this chapter we have presented the Reflection Assistant model, a generic computational model for metacognitive instruction in problem solving learning environments. The architecture of the RA was presented and details about its components were discussed.

The ultimate goal is to create a comprehensive problem solving environment that provides activities that serve to anchor new concepts into the learner’s existing cognitive knowledge to make them retrievable. We believe that the metacognitive training proposed by the Reflection Assistant model can help with that.

One important innovation introduced by the RA Model is the idea that it is necessary to create purposefully and separate situations where self-reflection can occur. In other words, that it is necessary to conceive specific moments and activities to promote awareness of aspects of the problem solving process. Therefore, the RA is designed as a separate component from the problem solving environment with its own specialized activities.

Moreover, two new stages in problem solving activity are proposed: pre-task reflection stage and post-task reflection stage. During these stages the student interacts solely with the reflective activities proposed in the RA Model, which focus on her metacognitive skills and her problem solving experience.

The RA Model follows principles of reflective tools as facilitators, i.e. tools that aim to help student becoming self-regulators of their own learning (Reusser, 1993; Collins and Brown, 1988). As such, the RA adopts the following instructional principles:

1. The RA intends to promote students’ sense of autonomous thinking. Therefore, it is not directive and it does not tell the learner what to do. It suggests, encourages, and reifies students actions.

2. The RA provides feedback on strategies used, actions taken, and performance from the metacognitive perspective.

3. The RA tries to avoid cognitive overload. It lets students perform their problem solving activity with minimal interference during the production phase.
4. The RA coaches the metacognitive skills overtly and explicitly, but always contextualized in the domain; keeping the balance between domain general activities and task activities is essential to the success of the metacognitive training.

As the RA is a generic model for problem solving environments its activities and components have to be adjusted and implemented according to the specific domains and environments it will be used with. Prior to implementing the RA for a specific learning environment the instructional designer has to review the reflective activities proposed and adapt their language to the domain at hand. Besides that, the programmer has to implement the modules and data repositories of the RA. Additionally, new data repositories may have to be included in the problem solving learning environment, which will capture the history of the students’ interactions and problem solving performance. These data are used by the RA, following the information provided in table 4.5. In Section 4.4.6 we have presented the communication points between the RA and the learning environment in detail.

In order to test the RA model we have implemented it into a new learning environment for algebra word problems, called the MIRA System. So, we had to implement both the environment and the RA modules and data repositories.

The next chapter presents the domain of algebra word problems used in the MIRA system. After that, the other two chapters are dedicated to show the design and functionality of MIRA, and how MIRA incorporates the RA Model.
Chapter 5

The Domain in MIRA: Algebra Word Problems

5.1 Introduction

This Chapter presents the domain of Algebra and how it is incorporated into the MIRA learning environment.

Despite being mainly interested in metacognitive scaffolding, our research proposed to implement the general Reflection Assistant model into a learning environment to be used in real classroom situations - the learning environment is called MIRA. Therefore, we had to decide upon an appropriate domain for the learning content presented in MIRA and we chose relational algebra word problems. Algebra is a central topic in school mathematics and it is a suitable domain for using a problem solving approach for learning, which is an important requirement for MIRA.

First, this chapter presents the characteristics of the domain of relational algebra problem solving, the skills needed to successfully translate a word problem into equations and the common misconceptions students have. These characteristics were taken into account in the development of MIRA.

Then, it presents illustrative examples of other ILEs for algebra problem solving. There is quite an extensive research agenda in AIED and Interactive Learning Environments involving algebra teaching and learning. Different goals have motivated previous researchers who investigated algebra word problems, ranging from developing new tutoring methods, to testing cognitive theories and computational modelling techniques. These were also used to inform the design of the problem solving activity in MIRA.

Next, we introduce the heuristic devised, which aims at helping students to comprehend the mathematical components in a word problem and translate them into equations.

Finally, the chapter presents a tool implemented and incorporated in MIRA, which helps students to identify and organize the relevant parts of a problem. This organizing tool is called PAL Tool. A pilot study was conducted to inform the interface design and features of the tool. This study is reported with a discussion of the results. The final interface design implemented is then presented, and the limitations of the tool are discussed.
5.2 Reasons for Choosing the Domain of Algebra

It was desirable to choose a domain where the problem solving activity is commonly used as an approach for teaching and learning. The reasons for choosing Algebra as the topic to be taught within MIRA were twofold:

i. Algebra is an important subject in the Mathematics Curriculum.

The National Council of Teachers of Mathematics (NCTM), which is an international professional organization committed to excellence in mathematics teaching and learning, calls for an increased focus on problem solving in the K-12 mathematics curriculum and suggests that algebra should play an especially important role for students in the middle and high school grades. NCTM recommends increased attention to the use of real-world problems, use of computer utilities, and mathematical communication.

ii. Algebra problem solving involves a wide range of cognitive skills, which creates a suitable environment for developing metacognition.

According to the Principles and Standards for School Mathematics (NCTM, 2000) a good algebra problem solver needs to master a range of skills that includes, amongst others (Chapter 6, p.222):

- Understand relationships, being able to relate and compare different forms of representation for a relationship.
- Represent and analyse mathematical situations and structures using algebraic symbols. In order to do this, students need to develop a conceptual understanding of different uses of variables, to use symbolic algebra to represent situations and to solve problems, and to recognize and generate equivalent forms for simple algebraic expressions and linear equations.
- Use mathematical models to represent and understand quantitative relationships. Students need to model and solve contextualized problems using various representations, such as graphs, tables, and equations.

5.3 Relational Algebra Word Problems

An algebraic word problem consists of one or more sentences having some known or unknown values with underlying algebraic equations (Aziz et al., 1995). The problems present a situation or a story, where one needs to understand the elements that make up the situation to generate a mathematical model to represent it. The model consists of one or more equations; solving the equations yields numerical values that solve the problem. A relational algebra word problem is the one with relational propositions that connect the various elements presented in the story. The problem presented below is a typical example of a relational algebra word problem and the sentences in bold are examples of relational statements.

In April, Jill planted 40 roses in her yard. Then in May, she decides she needed more flowers and she planted some tulips and some poppies. She planted twice as many tulips as she had planted roses and one fourth as many poppies as tulips. How many flowers had she planted all together?

5.3.1 The process of solving algebra word problems

Researchers on maths problem solving divide the activity of solving word problems into two stages that comprise distinct cognitive processes (Mayer, 1999; Whimbey and Lochhead, 1999): Problem
Chapter 5. The Domain in MIRA: Algebra Word Problems

Representation and Problem Solution. Solvers may iterate between these two stages.

Each stage is composed of specific activities, which use distinctive skills and knowledge. Figure 5.1 shows the stages and knowledge involved.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Knowledge Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem Representation</td>
<td>Linguistic</td>
</tr>
<tr>
<td></td>
<td>Semantic</td>
</tr>
<tr>
<td></td>
<td>Schematic</td>
</tr>
<tr>
<td>Problem Solution</td>
<td>Strategic</td>
</tr>
<tr>
<td></td>
<td>Procedural</td>
</tr>
</tbody>
</table>

![Figure 5.1: Conceptual stages of algebra problem solving according to Mayer (1999).](image)

**Stage 1: Problem Representation**

According to Mayer (1999) this stage is composed of two process: translation and integration. These processes encompass the transformation of algebra word problems into a system of equations.

In the literature there are some variations in the terms used. Heffernan (2001) uses the term *symbolization* to denote the translation process; yet, Whimbey and Lochhead (1999) use *translation* to indicate both the translation and integration processes. Similarly, we will adopt the word translation to signify both sub-processes of the Problem Representation stage.

The knowledge and skills necessary to master the stage of Problem Representation are three-fold:

- The learner has to be able to translate each statement of the problem into some internal representation in memory. This process requires that the student understands the sentences used to express the problem situation (i.e., the English sentences). It implies that the student needs *linguistic knowledge*.

- The learner also needs *factual knowledge*. This means that the problem solver needs some knowledge related to the situation. For example, knowing that one hour has 60 minutes. In some cases this factual knowledge is more specific conceptual knowledge. For example, knowing that all sides of a square are equal in length is more than observational knowledge. It is knowledge of a specific mathematical concept.

- In order to build an accurate mathematical representation of the story problem the learner needs to integrate the information in the problem in a coherent way. So, she needs to understand how the statements in the problem fit together and to have knowledge of problem types (i.e., *schematic knowledge*).

**Stage 2: Problem Solution**

This stage involves breaking down the problem representation devised in the previous stage to depict a numeric solution, and checking this solution against the original statement. The learner has
to deploy algebraic and arithmetical rules and a wide range of mathematical skills, such as understanding of variable, equality, and symbol manipulation (for example: substitution, simplification, etc.).

5.3.2 The main misconceptions and difficulties

Previous research investigated the main misconceptions students have regarding algebra problems and pointed out the translation stage or sub-processes in the translation stage as the source for most of them (Hinsley et al., 1977; Heffernan and Koedinger, 1998; Mayer, 1999; Whimbey and Lochhead, 1999). Even when students master the basic operations and can manipulate equations, they have problems in creating the equations and understanding what they have to do and why they are doing it (Hinsley et al., 1977; Derry and Hawkes, 1993).

Heffernan (2001) argues that the ability to take a problem situation and formulate a mathematical model is the single most important skill needed to succeed in algebra problem solving:

“Even if a student has access to a spreadsheet or graphing calculator, the student will still need to be able to translate a problem into the language of mathematics (i.e., an algebraic expression)” (Heffernan, 2001, p. 3).

Derry and Hawkes (1993) conducted research with 244 college students and 81 7th graders to observe them solving algebraic word problems. They found that virtually all of them were able to solve simple problems, such as ones with a single schema in it, indicating basic understanding of algebraic concepts. However, the students experienced increasingly greater difficulty as the complexity of the problems or the number of steps to reach the solution increased. They concluded that even among college students there is great difficulty in solving complex, multistep algebra problems. They believe that the nature of the difficulties exhibited by the participants in their experiments is related to misconceptions in comprehension of multiple relational schemas to create problem representations. In their opinion, metacognitive knowledge is what enables some students (and not others) to build complex representations and devise solutions for difficult problems. They highlight planning and monitoring as important metacognitive abilities for this purpose.

In our research we focus exclusively on students’ abilities to translate algebra word problems into equations because we believe that this stage is the most suitable for experimenting with metacognitive instruction. Thus, at the domain level, MIRA adopts an instructional method and includes learning resources specifically built to help students in the translation of verbal representations into algebraic representations.

5.4 Algebra Problem Solving in ILEs

There has been great interest in the domain of algebra in AIED research community. One motivation seems to be the degree of difficulty students have in mastering this domain. Some researchers characterize algebra word problems as an enigmatic domain (Koedinger and Anderson, 1993); i.e, part of a class of domains that students find particularly mysterious such that even when well-motivated they have difficulty (another example of such domain is geometry proofs).

---

1Derry and Hawkes use the term “schema” to refer to mental abstractions that allow set-relationship patterns in problem situations to be recognized and that indicate actions appropriate for those patterns (Derry and Hawkes, 1993).
Some projects focused on the problem solution stage, presenting equations and training skills such as symbol manipulation and simplifications. Algebraland (Brown, 1985) is an early example. As presented in Chapter 3 this system exhibit equations in a graphical form and has a menu with a choice of operations to simplify the equations.

Other projects involved both stages: problem representation and problem solution. The most extensive work in this direction is from Anderson’s group at the Carnegie Mellon University with The Carnegie Learning Tutors. In the early 1990s, the Pittsburgh Advanced Cognitive Tutor (PACT) Center at Carnegie Mellon University developed their first intelligent algebra word problem tutor, initially known as the Pump Algebra Tutor (PAT) (Koedinger et al., 1997). New, refined versions, named Algebra I Cognitive Tutor and Algebra II Cognitive Tutor, were built and now they are marketed under the name “Cognitive Tutor”.

The algebra tutors were constructed around a cognitive model that reflects the ACT-R theory. This theory considers that problem solving skills can be modelled as a set of independent production rules. The cognitive model enables the tutor to trace the student’s solution path through a complex problem solving space, providing feedback on each problem solving action and advice on problem solving as needed (Corbett et al., 2000). So, the Cognitive Tutor keeps track of the students’ progress, both for each problem and across problems in a lesson and chooses problems based on the students’ strengths and weaknesses. The Tutor controls the advance to a next lesson, and this happens only when the student reaches mastery level in all skill areas. They also support reasoning among multiple representations: tables, graphs, symbolic expressions, and natural language. Various versions of the Cognitive Tutor have been used by school children in USA.

A more recent and interesting project from the Carnegie Group is Ms. Lindquist (Heffernan, 2001). It differs from the other tutors, because it focuses essentially on the problem representation stage, helping children to build algebraic expressions from the textual description of problems. Therefore, it does not addresses the teaching and learning of many algebra skills, like graphing and equation solving. Instead it concentrates on developing in the student a repertoire of strategies for the translation of problems into expressions.

Other examples that focus more on the translation stage are the Heron-WATGRAF (Reusser, 1993) and TAPS (Derry and Hawkes, 1993). We have already commented on them in Chapter 3. Both provide a graphical tool for organizing the components of the problem in a tree that links them and makes the relations and operations explicit. Heron also uses a glossary with words presented in the problems to help with the linguistic and factual knowledge.

The TAPS project involved an important analysis of the kinds of errors students typically make while solving algebra word problems. Eleven general categories of deviation errors were defined, and further divided into smaller subcategories. The main categories are summarised below (Derry and Hawkes, 1993, p.118, 119):

1. bad value assignments, which can be (among others) due to a transfer mistake (copying wrong values from the problem description to the solution);
2. bad labels to components of the problem, what may signify that students overlook important details or overgeneralize the information provided in the text;
3. bad schema; i.e., students construct schemas that are not in the current solution path;
4. arithmetic errors: wrong operator or computing errors;
Chapter 5. The Domain in MIRA: Algebra Word Problems

5. omit errors: omit steps or branch of solution;
6. isolated bits presented as solution;
7. hesitancy, with long idle period with no action;
8. wild goose chase with the connection of three of more bad schemas as a solution strategy;
9. given up; i.e., students quit or present an answer before completing a solution strategy;
10. unnecessary strategy shift, abandoning promising path of solution;
11. time expires.

The prototype of TAPS has the capability of recognizing and interpreting some of these errors, using a pattern-matching algorithm based on a fuzzy logic technique. Unfortunately, a tutoring and remedial help was not implemented in TAPS.

Another tutoring system, also called TAPS (Aziz et al., 1995), focuses on misconceptions related to the reversal error (an error often observed in expressions like “At this university there are six times as many students as professors”, where many students answer \( 6S=P \) instead of \( S=6P \)).

A different approach was adopted by the ANIMATE system (Nathan et al., 1992). ANIMATE is a tool students can use to describe equations for word problems, which can then be “animated”. The system does not tell the students when they make a mistake; instead, the student observes the behaviour of the animation to figure out what is wrong.

All these projects made some kind of contribution to a better understanding of problem solving strategies, students’ misconceptions, and possible ways of fostering the learning process in this domain.

5.5 Helping to Translate Problems into Equations

Polya’s work sheds light into the approach we have adopted for helping students translate algebra problems. We have envisioned an heuristic that follows Polya’s principles (Polya, 1945) for the types of problems we are using.

Our heuristic involves the identification and understanding of the main elements of algebra word problems: \textit{givens}, \textit{goals}, \textit{unknowns}, and \textit{relations}. We believe that a declarative understanding of these elements can be a powerful resource leading to a full understanding of the problem. It enables the organization of the problem into a format to which the student can apply some strategies in order to successfully translate the problem into equations.

Below we define the elements and the heuristic developed. We use the following relational word problem to give a practical example along the explanations:

\textit{Paul is forty years old. He has two sons, Mat and Kevin, and one daughter, Emily. Mat is one quarter of Paul’s age. Kevin is two years younger than Mat. How old is Kevin?}
5.5.1 Definitions of concepts used

The Givens
The givens are the values that are known and that will be used to solve the problem. In the example above there are two givens: Paul’s age (40) and the number of children Paul has and their genders (2 sons and 1 daughter). However, not all givens are relevant to the problem. In this case, the number of children is not needed to solve the problem.

The Goals
The goals of a problem are what we have to find. The values of the goals are unknowns (we find them out by working out the equations). In the example above, finding Kevin’s age is the goal of the problem.

The Unknowns
The unknowns are quantities in the problem that one refers to, but their value are not explicitly stated. They need to be worked out in order to reach the goal. In the example above, Mat’s age is an unknown; we need to find it out in order to work out Kevin’s age. Note that Emily’s age is also unknown but this turns out not to be needed.

The Relations
The relations of a problem are the sentences connecting the givens, the unknowns, and the goals. They are the sentences that describe links between elements. In the example above, the sentence “Mat is one quarter of Paul’s age” states a relation between Mat’s age and Paul’s age. The sentence “Kevin is two years younger than Mat” is another relation, connecting Kevin’s age and Mat’s age.

Equations
Equations are mathematical expressions of the relations. Arithmetic operations (+ - * /) are used to represent the type of relation, while variables represent the unknowns. In the example we have the following equations:

Paul is forty years old. Let \( P \) be Paul’s age.
   \[ P = 40. \]
Mat is one third of Paul’s age. Let \( M \) be Mat’s age.
   \[ M = \frac{1}{3}P \text{ or } M = (1/3)\times40 \]
Kevin is 2 years younger than Mat. Let \( K \) be Kevin’s age.
   \[ K = M-2 \]

5.5.2 A heuristic to help with problem translation

The heuristic is composed by a series of actions that can be performed repeatedly, until the student is satisfied with the solution. Sometimes the student only recognizes an important aspect of the problem after some unsuccessful attempts to generate the equations. In these cases, it is necessary to go back to a previous step and review it. The basic steps are:

1. Read the problem description and identify the goals. State each goal in your own words and create variables to represent them.
2. Read the problem description and identify the givens of the problem. Create a variable for each given and assign the known value to this variable.

3. Identify the unknowns that are explicitly described in the problem statement and define variables to represent them.

4. Select from the problem description the sentences that define relationships between two or more elements.

5. Rewrite each relation substituting the parts of the sentence that refer to the components by their variables names. If no previously identified component matches any of the components in the relation, identify its type (givens, goals, or unknowns) and create an appropriate variable for it.

6. Analyse the mathematical meaning of the relation and decide which arithmetic operators should be applied (+, -, *, /, =).

7. Translate the relation into an equation, using the variables and the operators organized in the form that best matches the meaning of the original sentence.

8. Read each equation produced (in more than one way if possible) to verify that it translates back to the original sentence.

9. Check carefully if any equation has been omitted and if all variables previously identified have been used in the equations.

5.6 Designing a Tool to Support Word Problems Translation

In order to enable students to use the heuristic proposed, we have designed a tool to support the translation of word problems into equations. As explained in the previous sections, one of the hardest problems in translating algebra problems into equations is comprehending the situation and constructing representations of it. We would like to develop an instructional approach that gives students control over their process while helping them to build this representation.

The PAL Tool was designed to meet this requirement. It is the tool in MIRA that helps students to organize and represent the problem into its smaller components (the givens, the goals, the unknowns and relations). The PAL Tool implements a specific strategy for translating algebra word problems. This strategy is in line with strategies proposed by Polya (1945), Mayer (1999) and many algebra courses (e.g. the Math Forum). It motivates the students to look for specific elements in an algebra problem, labelling these elements as givens, goals, and unknowns. Also, it helps in making the connections between the relations and the elements previously identified. It provides an interface for describing the components of the problem, creating variable names, identifying the relations and the components that are involved in each relation. It focuses exclusively on the translation stage of algebra problem solving and is intended to provide limited help on the task, letting students figure out the equations on their own.

5.6.1 Requirements of the PAL Tool

The conditions or requirements the PAL tool had to meet in its design to achieve the goals were:

1. Help students to isolate and categorize parts of the problem;

---

2The Math Forum is an Internet web site for mathematics. http://mathforum.org/
2. Make students think about implicit information (sub-goals, unknowns, etc.);
3. Help students to think about the relations in the problem and their conditions;
4. Assist students in making explicit the relations identified;
5. Help students to create a basic notation for thinking about the problem;
6. Provide input information to the metacognitive engine that is used to infer about students’ understanding of the problem;

5.7 A Pilot Study with the PAL Tool

Prior to the implementation of the PAL Tool a low fidelity prototype built on paper (Preece et al., 2002) with the features being considered for the final software was designed to inform the final design. A summary of the pilot study using this prototype and its results were published in Gama (2002).

The conceptual design of the prototype was simple, with some tables in which each line stood for an element of the problem. Figure 5.2 shows this initial mock-up of the tool.

The buttons on the paper served to trigger help on the task. There were two types of help: Show Hint and Give me One. The hint consisted of a multiple choice question with three possible answers about the givens, goals, unknowns, or parts related. When students asked for a hint, a question with the possible answers was shown, but no feedback was provided regarding which answer was the correct one. They had to figure it out by themselves. The give me one button, on the other hand, provided complete information about one part of the problem (givens, goals, etc.). For example, if the student placed a rubber (this was the way to indicate the action of “clicking on a button”) on top of the give me one button next to the “goal table” the experimenter would produce a piece of paper containing one goal description, together with an identifier and the corresponding unit (i.e. a complete row of the goal table).

The “fade-out” function was provided as a pen that students used to cross out irrelevant information. Figure 5.3 shows a participant using the paper version of the Pal tool - notice the rubber, pen for “fade-out” functionality and pieces of paper with hint messages.

The conceptual model of the low-fidelity prototype aimed at helping learners to perform the following activities:

1. find and label relevant elements of the problem.
2. classify the elements in three distinct categories: givens, goals, and unknowns.
3. write down the relationships identified in the problem, with the parts related (givens, goals, unknowns).
4. write down any assumption they made to solve the problem.

On a separate piece of paper the students had to write down the equations and solve the problem. Some students were asked to try to write down the equations before using the Pal Tool. After a few minutes, they were given the PAL Tool sheet to organize the problem and then go back to the equations. Others were asked to first organize the problem using the Pal Tool sheet and then try to write down the equations.
Seven students used the low-fidelity prototype to solve algebra problems. The users were video recorded using the tool and an informal interview was conducted to get a better understanding of their impressions about the experience. An account of this experiment is given below.

5.7.1 Goals of the study
The pilot study aimed to evaluate how effective the PAL Tool was in helping students understand and identify the elements of a problem. The questions put forward were:

1. Does the tool help students go further in problem translation than they would without it?
2. Is the tool conceptual model understandable/appropriate to the students?
3. What type of difficulties do students report regarding the use of the tool?
5.7.2 Methodology

Participants: The sample consisted of seven volunteers aged 21 and above. Six of them had already graduated in social sciences courses and five had not taken a maths course for more than two years. All of them reported difficulties in maths problem solving in general.

Procedure: The participants filled in a demographics form and took an 8-item maths test that evaluated algebra translation skills. Participants then read a tutorial on the tool and saw an example. Two different procedures followed:

1. five students used a low-fidelity prototype of the PAL Tool to translate a fairly difficult algebra problem;

2. two students that reported being very weak in problem solving (and did poorly in the maths test: 1 out of 8 questions solved correctly), tried first a simple problem using a blank sheet of paper and then organized the same problem into the prototype. They were then given the same problem that was given to the other participants to be organized into the tool prototype.

Participants were asked to think aloud during the session. At the end, they were interviewed and asked about the interaction with the tool.

Results: Although it is not possible to gather statistically significant data with only seven participants, some quantitative information was obtained:

(i) Average time spent translating problems: 22 minutes;

(ii) Completion of the task: 2 students translated a complete problem without difficulty; the remaining 5 either did not complete the problem or needed help from the experimenter in order to move forward;

(iii) Difficulty: the goals were the easiest parts to identify (5 students did it correctly), followed by the givens (3 students did it correctly; 3 did it partially, including wrong givens; and 1 did not identify the givens at all). Deriving relations was the most difficult activity (only 2 students completed the task). See Figure 5.4.

![Figure 5.4](image.png)

**Figure 5.4:** Results of pilot study: ability to identify parts of the problem: students found that relations were the most difficult components to be identified.
5.7.3 Discussion

The study results were in line with research showing that construction of relations is the most difficult part of the translation process (Heffernan, 2001). The two weakest students, even after receiving hints and items to complete their task, were unable to understand the meaning of relations and solve the problem. One student commented:

“I am still trying to figure out what this means, it is all abstract to me.” (participant 3)

This suggests that weaker students found the tool less helpful because it did not provide instruction in the underlying principles of algebra, such as the value of translating a problem into an algebraic expression, the meaning of variables, etc.

In the interview, six participants reported that they benefited to some extent from the Pal Tool as an organizational tool. One participant made the following comment:

“It helped me to organize the problem, but then it turned out I wasn’t organizing it the right way.” (participant 3)

The same six students reported that the hints were useful to keep them on the right track. One participant made the following statement during a session when she “pressed” the “Show Hint” button:

“I want to be given one just to make sure I am at the right track here.” (participant 1)

Nevertheless, bringing parts together into relations was reported by all as confusing and hard. The time spent on the task was not well balanced; some participants spent most of their time identifying the elements of the problem and lost view of the necessity of integrating them in the equations.

From this study it was clear that in addition to the tool, the MIRA environment should include facilities to support the conceptual understanding of algebra. A glossary with examples of expressions, involving the arithmetical operations and typical expressions, might help students to translate words into equations.

5.8 The Implementation of the PAL Tool

With the insights produced by these results we revised the design of the PAL Tool, which was implemented in JAVA.

All features of the low-fidelity prototype were implemented, except the “fade out” feature. This feature was a dangerous one: if the student mistakenly crossed out an important part of the text, she was unlikely to review the decision.

As one of the requirements of the PAL tool was to provide input information to the metacognitive engine about students’ understanding of the problem, we used a technique to check the information students’ entered in the tables.

We adapted an existing matching algorithm, called The Porter Stemming Algorithm (Porter, 1990), that scans students’ answers and tries to determine if they are correct.

The algorithm identifies the stem of each word in the student’s answer and checks it against a set of word stems representing the correct answer. To make it more robust we included for each
word its most common synonyms. MIRA uses the information provided by this algorithm as part of the calculations of the student problem solving performance.

This method of analysing students’ responses was chosen because of its simplicity. We are aware that other more accurate and flexible methods exist from which we could get more accurate interpretations of students’ responses (e.g. using a parser driven by a context free grammar that takes a string as input, performs a series of lexical, syntactical and semantic analyses, produces an intermediate representation of the students’ responses in the form of a tree, for example). However, they are also more costly to implement. We judged that the matching method was a good compromise for our purposes, taking into account that: (i) the PAL tool was an additional feature in MIRA and not the main outcome of the research; (ii) the use of the PAL tool in MIRA would be optional and therefore, we wouldn’t have precise information for those students that decided to not use it anyway; and (iii) we only needed to differentiate between low, average and high understanding of the problem, and not a fine grained information about students’ performance.

5.8.1 Interface design

The PAL Tool uses tables as representations of the components of the problem. It uses four tables: givens, goals, unknowns and relations. Each line in these tables describes one element of the problem. Figure 5.5 shows the tables in the PAL Tool.

**The Givens Table**

In the givens table the student includes a textual description of each given she identified in the problem, along with its value and unit. For example, if a problem states that “John is 40 years old” then the given description would be “John’s age”, the value would be 40, and the unit would be “years”. By having to state the value of the given the student can check if the element she is including in the givens table really is a given.

**The Goals Table**

In the goals table each goal is listed in a separate line along with a description and a variable name. Students have to use the correct variable that was previously created to represent that goal. The PAL Tool allows students to insert, edit and delete lines in tables. It also allows students to cut sections of the problem description and paste them into the tables.

**The Unknowns Table**

Similarly, in the unknowns table the student includes a description of each unknown identified in the problem and creates a variable name for it.

**The Relations Table**

The givens, unknowns and goals are normally related to each other. The translation task involves recognizing and isolating the relationships described in sentences in the text and working out which elements are related in each sentence. In the “relation description” column the student can either enter a sentence that represents a relationship or copy and paste sentences directly from the problem description. The middle column has a button with a pencil icon on it. When the student clicks on this button a list of all variable names and values (only for givens) of the elements previously created (givens, goals, unknowns) pops up.
The student can select from this list all the elements that are involved in the relation she described (the list appears in a small window - see the window on the right-hand side in Figure 5.5). This feature was added to help students overcome the difficulty of putting parts together into relations, observed in the pilot experiment (see Section 5.7.3 for a discussion of this issue). After selecting the variables or values, this information appears in the “elements related” column. Thus the student identifies the elements that will compose each equation.

One design restriction is that the word matching algorithm described above does not check if the relationships created by the students are correct; it only considers givens, goals and unknowns. This feature should be implemented in a future version of the tool.

5.8.2 The help features in the PAL Tool
The implemented version of the PAL Tool has two instructional features that help the student to figure out the components and fill in the tables:

• **Hints**: These are questions with multiple choice answers that help students identify the givens, goals, unknowns and relations of the problem. There are four possible answers, only one of which is correct. The wrong answers were created with the aid of a maths teacher and represent common misunderstandings or misconceptions of algebra problems. However, we decided not to give feedback as to whether the answer chosen is correct or not, leaving it to the student to work it out.

The Hints feature presents only one question (i.e. hint) for each component, which is seen as a limitation of its help capability. Figure 5.6 shows an example of the Hints facility.
Another interesting approach would involve multiple levels of help (Koedinger et al., 1997), so that more detailed information about a component could be obtained by making repeated help requests. This would require a series of hierarchically structured questions for each component. Given the main goals of MIRA and the time constraints, this approach was not adopted.

- **Give One**: The PAL Tool would not fulfill its purpose if students continued to feel “stuck” in the problem and could not proceed in building its representation because they did not yet master the strategy or had difficulty with any aspect of the representation. The “Give One” feature supports students who do not fully understand, for example, what they should put in the “variable name” field of the unknown, or if they are unsure about their choices and need some confirmation. To prevent abuse (e.g., students asking for all components) the “Give One” feature only provides information about one component in any given problem. If the student attempts to obtain help on a different component she will receive a message indicating that she has already invoked help once.

The information provided by the “Give One” facility includes a description of the components and suggestions of variable names. See Figure 5.7. It supports all elements in a category. For instance, if the student asks for “unknowns” she will see all unknowns in that problem; she can then complete the table of unknowns with the information provided.

![Figure 5.6: The Hints facility in the PAL Tool. If the student is uncertain about any component of the problem, she can request a hint. This consists of a question and four possible answers designed to make the student reflect on that concept. No feedback on the student’s answer is provided.](image)

### 5.8.3 Limitations of the PAL Tool

Our observations indicate that the PAL Tool is more likely to be effective if used at the beginning of the problem solving as a starting point for helping the translation stage. As students have to read the problem several times in order to identify the main components, they start to think about
Chapter 5. The Domain in MIRA: Algebra Word Problems

98

Figure 5.7: The “Give One” facility in the PAL Tool. It displays all existing information about a chosen component.

the problem in a more structured way before jumping into wild goose chase attempts to write equations.

For students who normally have difficulty in solving this type of problem, the PAL Tool gives them an opportunity to identify some of these difficulties. However, for such students the tool is of limited use in achieving the goal of translating the problem into equations; they may need additional help, either by building other representations of the problem (such as diagrams or charts), by studying similar examples, or by taking basic algebra lessons.

Another limitation of the PAL Tool is that it does not suggest strategies to disentangle complicated problem descriptions (e.g. problems with hidden relations). It also lacks a glossary of terms, which could help students articulate and apply their linguistic or factual knowledge.

The current version of the PAL Tool does not force students to fill in all fields of the tables, neither does it provide feedback about the correctness of the information entered. For experimental purposes students were allowed to use the PAL Tool as an optional aid in MIRA, and they had to decide for themselves whether the representation they created for the problem was correct or not. Our intention was to observe students’ self-monitoring behaviour, allowing them to struggle with the problem or give it up if they so wished.

In future versions it would be desirable to extend the tool so that it provides more sophisticated feedback on several levels, such as incomplete set of components, wrong relations, missing variable names, etc.
5.9 Conclusion

This chapter has introduced algebra word problems and discussed the knowledge and processes required to solve these problems. Prior to solving the equations, the student needs to identify the algebraic elements embedded in the problem description and establish the relationships between them; this process is referred to as translation. It was shown that students experience greater difficulty in the translation stage, where they have to depict equations, than in solving them to yield a numerical result.

The literature proposes strategies to foster this process. Based on them we devised a heuristic and implemented it in MIRA in the form of a help system - the PAL Tool. The heuristic requires an understanding of the different components of an algebra word problem and their mutual relationships. It proposes a structured way to build an intermediate representation of the problem using these components to help the student build equations.

Thus, the PAL Tool assists students to identify and organize the components of a problem and translate them into linear equations. This chapter described a preliminary study aimed to assess the usefulness of the PAL Tool and to inform its final design.

We believe that metacognitive development and self-reflection take place in contexts where the student is challenged but also appropriately supported. On that basis we devised and implemented the PAL Tool as well as the other problem solving facilities in MIRA. Together they provide a supportive environment, where the student actively directs her own learning. In the next two chapters we describe the MIRA system and how it brings together the domain of algebra problem solving and the Reflective Assistant model.
Chapter 6

A typical learning Session with MIRA: the ILE that implements the RA Model

6.1 Introduction

This chapter and the next are about the MIRA interactive learning environment. At the domain level, MIRA deals with algebra word problems; more specifically with the process of translating a problem from English into equations. The student is not meant to solve these equations to obtain numerical values for the unknowns. In MIRA sessions students develop skills and practise using algebraic notation to represent concisely the underlying structure of the situation presented in problems. MIRA is not a tutoring environment for first time learners of algebra. It assumes that the users have some operational knowledge of arithmetics (including operations with integers and fractions), and some previous experience in creating and working on with very simple algebraic expressions.

This chapter gives an overview of MIRA from the user’s perspective, showing the typical flow of student activities within the system. An introductory explanation of the sequence of activities in MIRA is given, followed by a step by step description of a typical learning session. Every step is illustrated with screenshots of the system showing the corresponding appearance of the graphical user interface.

It appeared to us that the issues at stake here were better approached by giving first a flavour of the overall behaviour of the system without getting into detailed explanations of the whys and hows of every feature. This is the purpose of this chapter.

Once the reader has seen the whole picture it is easier to examine the components in detail and explain their relation to the whole and the motivation behind their creation. This will be done in the next chapter.

6.2 Overview of activities in MIRA

The interaction with MIRA follows a pre-determined sequence of activities whose emphasis is either on metacognitive training or on problem solving skills. The sequence is shown on Figure 6.1. Each of the activities represented by a rectangular box is clearly signalled to the student. In
Chapter 6. A typical learning Session with MIRA 101

fact, every transition from one activity to the next is accompanied by a noticeable change of the user interface in order to offer only those functionalities relevant to the activity at hand. The rest of this chapter will be organised around those activities.

Figure 6.1: Sequence of activities in MIRA. Students perform one activity at a time. All activities, except for the Quiz, are performed in one iteration. The Quiz is included depending on the student’s performance on the problem.

6.3 A typical Learning Session

When the student sits at the computer, the application is already running and she has a number of options available in a navigation sidebar on the left. Some of these offer background information about the application and its goals. One allows the student to register and another allows her to login if she is already registered. We will describe what happens in the latter case as this is more representative of the typical case and only slightly different from what happens when she uses the system for the first time.
Clicking on the Login link will bring a popup window prompting her for her username (Figure 6.2). She types it and the system loads her history of previous interactions.

The first screen after being logged in is labelled “Session Activities - General View” (Figure 6.3). It is a map of the activities that make up one iteration. In order to make things simpler for the student, this general view groups the activities represented in Figure 6.1 into 5 stages and it will only appear before those activities which are the first in a new stage (labelled with ⋆ on Figure 6.1). In each of its appearances, this screen only differs in the fact that the stage which is about to start is highlighted. In the present case, the student sees that the stage “Reflection on your skills” is about to start. We will not mention this screen anymore in the rest of the text.

Clicking on the button labelled “Start Activity” will take her to the first activity.

**Reflection on judgement of understanding of concepts from previous problems**

For the first of the two pre-reflection activities, the user interface appears in the form shown in Figure 6.4. The text on the top explains that the student has to use drop-down menus to complete the sentences on the right into meaningful interpretations of the graphs on the left. Each bar graph compares the student’s self-assessed understanding of the givens and relations with the actual understanding demonstrated for these concepts. She looks at the top graph in Figure 6.4 and sees that she had assessed her understanding of the givens of problem 2 as high and that she demonstrated a low understanding of those givens when she tried to solve the problem. She thus fills the text next to the graph so that it reads: *In the last problem I said that I had a high understanding of the givens and I demonstrated a low performance for givens. So, my prediction of my understanding of givens was inaccurate. Looking at all past problems I believe that my judgement of my understanding of the givens is decreasing.* She does the same for the other graph and clicks on “Interpretation Done”.

---

**Figure 6.2:** MIRA: Login screen with navigation sidebar in the background. The student brings up the login screen by clicking on the last of the links in the navigation sidebar on the left.
Chapter 6. A typical learning Session with MIRA

Figure 6.3: MIRA: General View Window, indicating that the stage “Reflect on Your Skills” is about to begin. Variants of this screen with a different stage in bold appear along the sequence of activities to indicate beginning of a new stage.

Figure 6.4: MIRA: Reflection on predicted and demonstrated knowledge skills. She has to fill in the blanks with her interpretation of the graphs related to her understanding of the givens and the relations of the last problem.
Reflection on knowledge monitoring ability

She is now presented with the second pre-reflective activity (Figure 6.5). She can see two reflectometers that show her the scores MIRA has calculated for her knowledge monitoring skills. The first one indicates the accuracy of her estimate of her own knowledge (KMA). The second shows any bias that exist in this estimation (KMB).

![Image of reflectometers](image.png)

**Figure 6.5:** MIRA: Analysis of knowledge monitoring state (KMA / KMB). The student observes the system’s interpretation of her metacognitive skills and reads a feedback message.

If she does not understand the meaning of those graphs, she can click on the Help button and some text pops up with an explanation as shown in Figure 6.6.

She looks at the top reflectometer in Figure 6.5 and sees that its needle is low and the text next to it confirms that her assessment of her own knowledge is low. The other reflectometer has its needle in the optimistic range and the text next to it confirms that she is demonstrating an optimistic assessment behaviour, suggesting that she thinks she knows more than she actually does. The feedback message at the bottom gives her a tip based on her profile. Having finished these reflective activities, she is allowed to proceed to the problem solving stage by clicking on the button at the bottom right (this is consistently the way to proceed to the next activity).

Reading the problem and assessing understanding and difficulty

The first activity encountered in this stage is shown in Figure 6.7. As on previous screens, the text at the top tells her what to do: she must read the problem and answer questions about her understanding of the problem and her confidence that she will succeed in solving it. The problem on this occasion looks quite easy and she is confident she can solve it.

Selecting strategies

Now she has to select strategies that she expects will be helpful for solving that problem (Figure 6.8). The instructions tell her that she needs to choose at least one strategy from each of the three
Chapter 6. A typical learning Session with MIRA 105

Figure 6.6: MIRA: Help Window. The help always shows the goals of the activity and what the student should do to complete it.

Figure 6.7: MIRA: pre-assessment on problem understanding and difficulty. The student reads a new problem and has to answer the questions about her understanding of some concepts in the problem and difficulty perceived to solve it.
groups:

- Strategies for Monitoring Understanding;
- Strategies for Controlling Errors;
- Strategies for Revising.

She examines the strategies suggested and ticks some. She is also free to include new ones of her own making.

![Figure 6.8: MIRA: Planning strategies. The student plans the metacognitive strategies most useful for achieving the goal, i.e. translating the problem into equations.](image)

**Solving the problem**

Now she is going to start work on the problem and the system’s interface takes the form shown on Figure 6.9. It includes the problem description on the top left, an editable area on the top right where she can write her answer, a group of buttons offering resources on the middle left and a reminder of the metacognitive strategies she has selected in an earlier activity.

She decides to take advantage of the PAL Tool which she has learnt to use in the tutorial and found quite useful in solving a previous problem. Clicking on the appropriate button brings a popup window as shown on Figure 6.10. The tool encourages her to think about the problem in terms of givens, goals, unknowns and relationships between them.

When she can identify a given, she makes an entry in the table labelled “Givens of the Problem”. She gives a description of this given as well as its value and the unit in which this value is expressed. She identifies an unknown and the goal of the problem. She also identifies a relationship between these entities and the table for relationships allows her to input it. She knows that the equations of the problem are obtained by translating each of the relations into mathematical notation.
Chapter 6. A typical learning Session with MIRA

Figure 6.9: MIRA: Problem solving window. This is the main screen with the menu of problem solving resources, list of strategies selected and working area for including the answer.

Figure 6.10: MIRA: PAL Tool. The student uses the PAL Tool as an aid to break the problem into smaller parts and to organize her thinking.
Now she feels confident to write down her equations using the information organized in the PAL Tool: she uses the name of variables defined in the PAL Tool and makes one equation for each relation she identified in the relations table.

![Image of a learning session with MIRA](image)

**Figure 6.11:** Problem solving resources: Calculator and Answer to Problem. The student checks out her equations using these two resources available.

She decides to make use of the resources and clicks on “Answer to this problem”. She gets the numerical answer to the problem (Figure 6.11). She knows that it is useful for checking out her own answer, which she starts to work on. Her answer seems fine and she clicks on “finished”.

**Answering the problem**

This takes her to the point where she has to answer the problem (Figure 6.12). MIRA’s interface suggests how she has to proceed to provide a final answer to the problem. The student has to follow three steps:

1. using the values presented on the box on the left-hand side of the screen she is proposed to mentally check her own solution by plugging back those values into her equations (her solution is presented on the top-right corner of the window);
2. then, she has to select from a multiple choice the answer that she believes is the correct one;
3. and finally, she has to answer a questions about how similar her produced answer is to the selected answer.

She chooses an answer that she strongly believes is correct and she realises her equations are a bit different in format from that answer, but she also believes both are correct. A message box appear saying her answer is correct (Figure 6.13).

**Checking the teacher’s solution**

She skims through the teacher solution quickly, because she is confident the problem was quite easy for her to solve (Figure 6.14).
Chapter 6. A typical learning Session with MIRA

Figure 6.12: MIRA: Check Answer. The student chooses from a multiple choice the answer that is most similar to her own answer and judges how close they are.

Figure 6.13: MIRA: Feedback on Answer Given: it says that the answer chosen in the multiple choice was correct. The system does not have means of checking the correct answer from the student’s written answer.
Chapter 6. A typical learning Session with MIRA

Figure 6.14: MIRA: Teacher’s Solution. The student reads the solution provided by the teacher that emphasizes the process of creating the equations, rather than the calculation of the numerical solution.

Reflecting on problem solving process

Now she is going to reflect on the process of solving “The Flowers” problem. The system provides a window composed of two main parts: reflection on activities with the aid of a timeline graph and reflection on strategies (Figure 6.15).

First the student reads the text next to the graph. It gives positive feedback saying that she pre-assessed the problem as “quite easy” and indeed she worked out the correct solution in less than 11 minutes (“not bad!”, she thinks). She looks at the graph and sees that she spent more time pre-assessing her understanding of the problem and planning strategies than giving the answer or checking the teacher’s solution. She reckons the reflection prior to the problem solving might have helped her to succeed. Then, she moves to the second part of the reflection. She includes some comments on her use of strategies: “I worked out this problem by spending time breaking down the problem into small parts. After realising all the relations the problem was very simple...”. She also notices the list of strategies she planned to use on the left side and she realises that she indeed used some of them during the problem solving.

She finishes the reflection and clicks on “done”. The cycle of activities for that problem is over and she can start all over again if she wants to proceed to the next problem.
Chapter 6. A typical learning Session with MIRA

Figure 6.15: MIRA: Post Reflection. After finishing the problem solving it is time to reflect on the process, looking back to the time spent on each stage of the problem solving enterprise. Also, written reflections about strategies are encouraged.
Chapter 7
The design of MIRA

7.1 Introduction

The MIRA learning environment was built with the main purpose of testing the Reflection Assistant model and allowing empirical studies. This chapter presents the design of MIRA and the functionalities implemented at the metacognitive and domain levels.

7.2 Main Features

The Metacognitive Instruction using a Reflective Approach system (MIRA) is a problem solving learning environment built in Java. The main focus in the design of MIRA was to create a comprehensive problem solving environment that could be used in real classroom settings. Therefore, besides implementing the RA model, MIRA includes problem solving tools and resources to support students at the domain level.

Another version of MIRA, called MIRA-Ctrl, was also built to allow experimental studies of the effects of the RA model. The MIRA-Ctrl version has all the problem solving resources and functionalities of MIRA, except the reflective activities.

MIRA cannot be considered an Intelligent Tutoring System, because it does not implement a knowledge base of algebra nor does it have a diagnostic model of typical misconceptions of algebra problems nor does it give “intelligent” (customized and fine grained) feedback on the domain knowledge level.

The two major design features of MIRA are:

*Independence from the domain*

The design of MIRA keeps a certain level of independence of the domain. Thus, its content can be quite easily replaced for another domain (for example, we could add Computer Sciences problems related to constructing algorithms).

MIRA’s main structure was designed separately from its content. MIRA access external files that hold the database of problems and all other parts related to the domain.
Chapter 7. The design of MIRA

**Generation of research oriented log files**

MIRA’s main objective is to serve as a research tool for examining the effects of metacognitive activities implemented following the RA model. Thus, MIRA incorporates mechanisms to record problem solving behaviour, to track users’ actions, and to observe the manifestations of metacognitive skills.

In order to implement this feature, students are identified individually using mechanisms of registration and login. The information logged is used both to calculate the students’ metacognitive state, generate appropriate feedback, and to collect detailed information about students. This information is kept in research oriented log files that can be analysed *posteriori*. Details about the structure of these log files are presented in Appendix B.

The technical description of the design features of MIRA, including the log file format can be found in Appendix B.

### 7.3 Principles

The main principles observed in the design of MIRA were:

- To allow students to perform their problem solving activities with minimal interference, following a self-regulated approach to learning, where students have the study resources available and decide whether to use them or not.

- To give freedom and space for students’ autonomous thinking. MIRA presents activities where the opinion of the student about her learning process is important but it is not criticised. The student’s judgement of her progress is valued and, even if it is not accurate, the student will eventually realise it on her own with MIRA’s help with reflection on that matter.

- To provide the information students need to be able to reflect on their learning but do not always direct the content or the format of that reflection.

Initially, we have envisioned the Reflection Assistant model as adaptive to students’ learning styles, preferences, and evolution. Unfortunately, this principle was not fully implemented in MIRA’s interface. The analysis of students’ KMA and KMB is updated after each problem solving interaction and the system considers the history of the student. But the individualization in MIRA’s interface occurs only for the feedback students receive. To make up for the lack of adaptivity, we have adopted a self-regulated learning approach in MIRA, which makes the environment more under the control of the student.

### 7.4 Algebra Word Problems in MIRA

MIRA presents a series of complex algebra word problems and the students’ main learning activity is to translate the problems into linear equations with two or more variables. However, the students are not asked to solve the equations and find out the numerical answer to the problem. Actually, the numerical answer is given to the student at the beginning of the problem solving, and can be used as a resource. Thus, the difficulties students may experience in solving the problems are in the process of identifying the algebraic components of the problem and translating them into equations, as opposed to difficulties related to calculations to find out the answer.
This learning task is also quite new to students, who are rarely just asked to translate problems into equations or to interpret equations in a significant way as a self-contained activity in conventional school settings. Normally they build the equations and are asked to manipulate them using techniques from algebra or calculus.

One common feature to all problems presented in MIRA is that they require students to:

- discard unnecessary information;
- identify hidden unknowns or goals;
- untangle relations involving the givens, unknowns, and goals; and
- discover implicit relations.

A complete list of the problems used in MIRA, categorized by type (e.g. age problems, money problems, etc.) and level of difficult is given in Appendix A.

### 7.5 Stages within MIRA

MIRA offers a structured approach to the problem solving process by helping students proceed through stages and providing feedback on their actions whenever it is necessary.

The learning activities in MIRA were grouped into 5 stages as shown in Figure 7.1, which follow the conceptual stages proposed in Chapter 4 in Section 4.3.2 on page 62.

To help students understand and remember the organization of the activities in MIRA, the stages are labelled in such a way that students are always informed of which stage they are about to enter (see example in Figure 6.3 on page 102). The stages contain the following activities:

**Figure 7.1:** Stages in MIRA, which correspond to the problem solving stages proposed.

**Stage 1: Reflecting on skills**

This stage is divided into the two pre-task reflective activities implemented, based on the RA model.

1. Comparison of knowledge monitoring and performance for specific concepts of the problems.
2. Analysis of knowledge monitoring state (KMA / KMB).
Chapter 7. The design of MIRA

Stage 2: Assessing understanding and planning strategies
This stage is also divided into two reflective activities implemented based on the RA model.

1. Self-assessment of the understanding of components of the problem and problem difficulty.
2. Selection of metacognitive strategies.

Stage 3: Solving problem, presenting answer & doing quiz
The goal of this stage is making students concentrate on translating the problem into algebraic equations and monitor the application of their planned strategies.

We have designed a set of problem solving resources and an interface that enables students to see the strategies selected in the previous stage. The activities that comprise this stage are:

1. Problem solving.
2. Checking answer.
3. Quiz.

Stage 4: Checking the teacher’s solution
This stage corresponds to the conceptual problem solving stage named as Evaluation. It contains only one activity; the student sees the solution provided by a teacher, and uses it to compare with and study her own solution.

Stage 5: Reflection on problem solving experience
This stage was designed to promote reflection on the problem solving processes and strategies related to the previous problem attempted. It was implemented based on the guidelines of the RA Model.

The main aspects and functionality of these activities are detailed in the sections below.

7.6 Design of the Reflective Activities

7.6.1 Comparison of knowledge monitoring and performance
This pre-task reflective activity presents bar-type graphs and a task to students related to the graphs (see example in Figure 6.4 on page 102). The main components of this activity are:

The reflective graphs
Two graphs are presented and each one focuses on one specific algebra component. Instead of asking students to reflect on their overall understanding of the problem, or reflect on all concepts of the problem, we focused on two concepts: the givens and relations. The decision was made based on the fact that comprehending relational sentences is extremely important in order to produce correct equations and on the results of the pilot study (shown in Chapter 5, section 5.7). The students that participated of that study ranked the givens as the most new (or previously unknown) concept and relations as the most difficult concept to fully grasp in a problem.

Two vertical bars are produced for each problem attempted so far and they represent, respectively, the learner self-assessed understanding rated prior to solve the problem and the demonstrated performance. Whenever a problem is finished two new bars are included in each graph.
The bars may appear in three different heights, representing low, average or high understanding of the givens or relations.

**Explanations to graphs**
The student has to fill in the blanks of the text positioned next to the graphs. Each text is divided into two parts. The first part refers to the previous problem and the second part refers to all past problems shown in each graph. This serves as an aid to reflection and also to gather some evidence that the student understood the graphs’ contents.

For example, next to the graph related to the understanding of the givens the text referring to the previous problem attempted says:

| In the last problem I said that I had a ___ [low/ average/ high] understanding of the givens and I demonstrated a ___ [low/ average/ high] performance for givens. So, my prediction of my understanding of givens was ___ [accurate/ inaccurate]. |

The student fills in the blanks with the option corresponding to the correct height of the equivalent bar. If she selects the wrong height MIRA will present an error message.

The second part of the text refers to an overall comparison between the student’s self-assessment of her knowledge and her demonstrated understanding of the concept throughout problems. The goal is to make students reflect on changes on their knowledge monitoring and their understanding of those concepts. The text presented is:

| Looking at all past problems I believe that my judgement of my understanding of givens is ___ [decreasing/ random/ constant/ increasing/ I don’t know]. |

Students can choose among the possible options offered. Given that it is a personal interpretation and consequently there is no “correct” answer, the student’s selection is not checked by MIRA. The answers are recorded and may be used for further investigation of students’ perception of their own judgement of their understanding across problems.

7.6.2 **Analysis of knowledge monitoring state (KMA/KMB)**
This activity shows MIRA’s evaluation of their knowledge monitoring ability. The reflectometers proposed in the RA model are implemented (Figure 6.5 on page 103). The activity design is twofold:

- **Evaluation graphs with explanatory text:**
  The graphs in the form of reflectometers give a quick estimate of the student’s KMA and KMB levels and a text positioned next to each reflectometer helps with the interpretation of the information conveyed in the graphs. The content of the text depends on the values shown in the reflectometers.

- **Feedback message:**
  Textual feedback suggests an action to be taken in the future in order to improve the student’s level of problem understanding. This message varies according to the values shown.
During this activity, students are not explicitly asked to provide their opinion about MIRA’s evaluation of their KMA and KMB, or if they accept MIRA’s feedback. We expect, however, that students look at the reflectometers’ values, read the texts, and reflect on their meaning.

7.6.3 Self-assessment of understanding and difficulty

The student reads a new problem description and, before trying to solve it, she has to reflect on her understanding of the givens and relations of the problem and her confidence to solve it (see Figure 6.7 on page 104).

Four metacognitive multiple-choice questions are presented for this purpose. Table 7.1 shows the self-assessment questions presented and all possible answers. Students have to read the problem and answer all four questions by clicking on one answer (which are multi-exclusive) before proceeding to the next reflective activity.

<table>
<thead>
<tr>
<th>Understanding of the givens</th>
<th>Understanding of the relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q: Can you identify the givens of this problem?</td>
<td></td>
</tr>
<tr>
<td>○ No, I cannot identify the givens.</td>
<td></td>
</tr>
<tr>
<td>○ I can identify at least some of the givens for sure.</td>
<td></td>
</tr>
<tr>
<td>○ I can identify all givens of this problem.</td>
<td></td>
</tr>
<tr>
<td>Q: Can you identify the relations of this problem?</td>
<td></td>
</tr>
<tr>
<td>○ I cannot identify the relations at all.</td>
<td></td>
</tr>
<tr>
<td>○ I identify only some relations.</td>
<td></td>
</tr>
<tr>
<td>○ I can identify all relations well.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Difficulty</th>
<th>Solving prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q: How difficult do you think it will be to solve this problem?</td>
<td></td>
</tr>
<tr>
<td>○ Very easy.</td>
<td></td>
</tr>
<tr>
<td>○ Not difficult.</td>
<td></td>
</tr>
<tr>
<td>○ A bit difficult.</td>
<td></td>
</tr>
<tr>
<td>○ Very difficult.</td>
<td></td>
</tr>
<tr>
<td>○ Challenging.</td>
<td></td>
</tr>
<tr>
<td>Q: Do you think you can solve this problem?</td>
<td></td>
</tr>
<tr>
<td>○ No.</td>
<td></td>
</tr>
<tr>
<td>○ Maybe.</td>
<td></td>
</tr>
<tr>
<td>○ Yes.</td>
<td></td>
</tr>
</tbody>
</table>

Table 7.1: Pre-task reflection in MIRA: self-assessment questions & answers

7.6.4 Selection of metacognitive strategies

Three lists of pre-compiled strategies are offered, based on the RA Model initial list (see page 75). Students are required to select at least one strategy from each list. Alternatively, she can compose a new strategy and select it. The strategies found in the lists at the beginning of the training are only a starting point. Students can include new strategies, delete the ones perceived as not useful, or adjust existing ones. In this way, a personal library of strategies is built along with the interaction with MIRA. The initial set of strategies is given below:

- List of Strategies for Monitoring Understanding
  1. Read the problem more than once.
  2. Read the problem to separate the important parts.
  3. Think of a related problem I have already done and use it as a model.
Chapter 7. The design of MIRA

4. Before starting to solve the problem, think what I am supposed to learn from it.
5. Read the problem and determine which parts I don’t understand well.
6. Review the basic concepts that are not clear, before attacking the problem.
7. Set a goal to myself and think about the steps to reach this goal.

- List of Strategies for Controlling Errors
  1. Stop and review each step to see if I made a mistake.
  2. Reread the problem to check for forgotten important parts.
  3. Change strategies if I get lost and confused and don’t seem to move anywhere.

- List of Strategies for Revising
  1. Think about a way of checking to see if my solution is correct.
  2. Review all I did to make sure I am not forgetting anything.
  3. Reread the task description and ask myself if my solution really meets the task goal.

The lists of strategies are organized into tables. Figure 6.8 on page 105 shows the table with the strategies from the list “Strategies for Monitoring Understanding”. To include new strategies the user clicks on an empty line and just types the new strategy. The content is saved automatically. Similarly, to editing an existing strategy, the user clicks on the line she wants to edit and types the new text. Deleting strategies is achieved by clicking on the line to be deleted and then clicking on the delete button\(^1\) (it is located on the right side of the tables).

### 7.6.5 Evaluation of problem solving experience

This activity is based on the guidelines of the RA model and takes place after a problem is finished and the teacher’s solution is shown. It works in a threefold manner:

- the student examines the Activities Timeline Graph;
- the student reads the feedback message;
- the student writes a reflective journal about the strategies chosen and used.

Part of this reflective activity is designed to provide process-related feedback, including observations about the degree of use of the resources and issues related to effort in trying to solve a problem. The goal of this feedback is to make students reflect on the notion that when one is not confident in performing a task, one can look for help via the resources available in the environment itself. Alternatively, if one is already using the resources as an aid one can reflect on how one is using them. In this way, the feedback can raise links between the process and the outcomes obtained.

\(^1\)All buttons in MIRA have a “tool tip text” explaining their function. It helps users to understand the purpose of the button without actually clicking on it.
Activities Timeline Graph

This graph highlights important activities performed during the previous problem attempted and the time spent on each activity. The graph has horizontal lines in different colours representing each activity. If the student did not perform some activity no line will appear for that activity.

The student is instructed to observe the graph and reflect on the amount of time spent on each activity (represented by the length of the horizontal line), the absence of an activity, etc.

The activities presented in the graph are:

- Reflecting before the problem
- Self-assessing kills & planning strategies
- Solving the problem
- Using the PAL Tool
- Giving an answer
- Checking the solution

Feedback on the problem solving process

Next to the graph MIRA provides a textual summary of the problem solving process, including information about how the student assessed the problem initially and if she managed to solve it or not.

The content of the feedback message is dependent on a combination of factors, based on the values of certain variables. Five variables related to the interaction of the learner with the problem are used: answer, difficulty, effort, check-time, and PALToolUse. Table 7.2 shows the possible values of these variables and their meaning.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Possible Values</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Answer</td>
<td>Correct</td>
<td>Answer checked was similar to worked out solution.</td>
</tr>
<tr>
<td></td>
<td>Partially-correct</td>
<td>Learner thinks worked-out answer is correct, but could not find equivalent equations from multiple choice - i.e. worked out solution had mistakes.</td>
</tr>
<tr>
<td></td>
<td>Gave up</td>
<td>No answer was provided because student gave up on problem.</td>
</tr>
<tr>
<td></td>
<td>Wrong</td>
<td>Student checked the wrong option in the multiple-choice.</td>
</tr>
<tr>
<td>Difficulty</td>
<td>Low</td>
<td>Student rated the problem as “very easy” or “not difficult” prior to start the problem.</td>
</tr>
<tr>
<td></td>
<td>Average-High</td>
<td>Student rated the problem as “bit difficult”, “very difficult” or “challenging”.</td>
</tr>
<tr>
<td>Effort</td>
<td>Low</td>
<td>Time spent in problem was less than 50% the total time available (which is 15 minutes by default).</td>
</tr>
<tr>
<td></td>
<td>Average-High</td>
<td>Time spent in problem was greater or equal to 50% the total time.</td>
</tr>
<tr>
<td>Check-time</td>
<td>Low</td>
<td>Time spent checking the teacher’s solution was less than 1.5 minutes.</td>
</tr>
<tr>
<td></td>
<td>Not low</td>
<td>Time checking the teacher’s solution was greater or equal to 1.5 minutes.</td>
</tr>
<tr>
<td>PALToolUse</td>
<td>None</td>
<td>Student did not use the PAL Tool for the problem.</td>
</tr>
<tr>
<td></td>
<td>Low</td>
<td>Student used the PAL Tool for less than 25% of total time available (which means less than 3.75 min. of the default total time).</td>
</tr>
<tr>
<td></td>
<td>Average-High</td>
<td>PAL Tool was used for more than 25% of total time.</td>
</tr>
</tbody>
</table>

Table 7.2: Post-task reflection: variables evaluated in the feedback message
A decision algorithm was implemented to check the values of these variables in order to construct a complete message. Figure 7.2 presents the algorithm and the content of the messages. For simplification purposes the algorithm is shown here in four parts, each part corresponding to one possible value of the variable “answer” (cases 1 to 4). Part of the content of the messages are displayed after the algorithm.

Following the algorithm we can observe that, depending on the student’s performance, the message may include the total time spent on the problem with a comment stating whether the student should have spent longer trying to solve the problem. It also includes feedback on how the student could use that time in a more sensible way. Specific messages concerning the time spent on checking the teacher’s solution can also be included.

**Reflection on strategies chosen and used**
The list of metacognitive strategies selected at the beginning of the problem is presented again to the student and some reflective questions are posed concerning the use of those strategies during problem solving. The questions serve as a trigger to reflection. The box below presents the introductory text and questions.

```plaintext
Here is a space to make your personal reflections about your use of strategies in the last problem.
Do you remember when you used them?
Did they work well? What would you do differently next time?
Feel free to write whatever you want.
```

The students write down their reflections in a free-text format and they can see their previous reflections as well.

### 7.7 Design of the Problem Solving Activities

#### 7.7.1 Problem solving resources

When the student is trying to solve a problem, MIRA makes available a set of resources. They were designed to scaffold students in developing their cognitive skills for translating algebra word problems. The resources created are:

**PAL Tool:** As presented in Chapter 5 the PAL Tool aims at helping students to identify and organize the relevant parts of the problem.

**Library of Problems:** This presents all past problems solved. For each problem in the library it shows the problem description, the student’s answer, and the teacher’s detailed solution. Figure 7.3 shows the interface for this resource.

**Algebra Basics:** This is a textual explanation of the basic concepts of algebra with some examples. It is divided into 6 sections: (a) *what is algebra*, including definitions for expressions, variables and equations; (b) *what are variables*, including explanations and examples of when to use variables and how to represent variables; (c) *components of word problems*, which contains explanations about the givens, goals, unknowns and relations with an example; (d) *heuristic* about how to translate word statements into algebraic expressions (explained in section 5.5); (e) *examples of expressions*, showing how to apply the heuristic to translate word statements into algebraic expressions; and (f) *examples of equations*, showing how to apply the heuristic to translate word problems into equations.
Chapter 7. The design of MIRA

You spent only <number minutes> trying to solve the problem out of <total minutes solving time> minutes you had. You could have used the remaining time to move further in the problem or use the problem solving resources to help you to get a better understanding of algebra problems.

You have spent only <number minutes> checking the teacher’s solution. You could take this opportunity to study the problem more carefully to understand where you made mistakes.

You used the PAL Tool for only <number minutes> minutes. You could have spent more time using this resource to get a better understanding of the Givens and Relations.

You haven’t used the PAL Tool at all. This is a good resource to help you to separate the parts of the problem. Why don’t you give it a try next time?

Figure 7.2: Instructional messages presented in post-task reflective activity: an algorithm was created for deciding upon the content of the messages.
Chapter 7. The design of MIRA 122

Glossary of Relations: This is a series of examples of verbs and English expressions one can find in algebra word problems and their equivalent maths expressions. The examples are organized into tables, according to the arithmetic operations they represent (multiplication, subtraction, etc.). It is a pedagogical technique taken from algebra courses.

Calculator: This is a line calculator, which allows the student to enter arithmetic expressions with parentheses. The calculator is helpful if the student wants to check if her equations are correct, for example. Figure 6.11 on page 107 shows the interface of the calculator.

Answer to the Problem: As mentioned before, the numerical answer to the problem is given to the student. This reinforces the idea that the student is not being required to calculate the values. Instead those values can be used to check for correctness of the equations produced.

Time Left: This displays or hides a count-down timer. For each problem students have a maximum time to try the problem out (we have used 15 minutes in the main experiment). Also, it is important to note that students are not obliged to provide an answer to the problem or use all the time available. There are 3 distinct ways to reach the end of a problem attempt:

1. Give up: the student chooses this option if she wants to give up at any point or if she thinks she cannot solve the problem in the time left. If she chooses this option, she will skip the “Checking Answer” screen and go straight to the “Quiz” screen.
2. Finish: if the student finishes the problem before the time is up, she chooses this option and moves to the Check Answer.
3. Time is up: this is an automatic option that appears if the student reaches the maximum time (e.g.: 15 minutes). In this case, a message will come up asking if she has finished the problem. Depending on the answer provided MIRA shows the “Check Answer” screen or goes straight to the “Quiz”.

Monitoring Strategies Resource: Besides the resources for scaffolding problem solving, students have available the list of strategies selected for the problem while they are attempting
to solve it. They can browse the list at any point and we expect it to be helpful for recalling the strategies and triggering monitoring behaviour.

### 7.7.2 Checking answer

Whenever a student provides an answer to a problem (i.e., does not give it up) she is asked to check if her answer is correct, and relate it to one answer provided in a multiple choice question. The activity is thus completed in three steps:

1. The values of the goals and unknowns of the problem are provided. The student has to, mentally, substitute these values back in her equations (that is still available). She is instructed to check if both sides of each equation even up their values, using the information provided.

2. The student has to select one equation (or one group of equations) from a multiple-choice. The option selected should match or be very similar to the equations she produced. If none is similar, the student is requested to choose the option that seems correct.

3. The student has to answer a question stating how the selected answer relates to the produced equations. From a multiple-choice the student has to choose the answer that best describes the reasons for selecting the option in step 2. The question reads like this:

   ![Question](image)

   - I have worked out identical or equivalent equations.
   - I still think my equations are correct, but I cannot find an equivalent option.
   - Now I realized my equations were wrong and I selected the one I think is correct.

### 7.7.3 Answering quiz

A quiz is presented to the student whenever the answer chosen in the multiple choice (in the Check Answer activity) is either an incorrect one or it is correct, but the student acknowledges that it is different from her worked out answer.

The quiz aims at verifying the understanding of the givens and relations of the problem. The answers to the quiz inform the system about the student’s level of understanding of these concepts and it is used to calculate the student’s KMA and KMB and final performance score for that problem. The quiz is composed by two multiple choice questions. Figure 7.4 shows a typical quiz. The questions presented here are the same ones posed by the PAL Tool “Hints”. So, if the student has already answered any of the questions during the problem solving, the answer previously selected will appear marked, but the student can change her answer in the quiz.

### 7.7.4 Checking the teacher’s solution

In this activity students receive feedback on the problem’s solution. MIRA doesn’t interpret the student’s solution and explain why it is wrong or right. Instead, it shows the teacher’s solution and the student’s attempted solution side-by-side. The student is instructed to read both answers and compare them. The teacher’s solution is shown in detail and structured in a step-by-step way that emphasizes the identification of the elements of the problems before the construction of the equations. It does not include the stage of calculation of the numerical results of the problems, terminating when the equations are formed. The organization of the solution seeks to reinforce the
application of the heuristic adopted to translate problems and to indicate that it is not necessary to calculate the final values in the translation process.

We are aware of the limitation of this type of feedback for novice students as it can produce a tough situation where the student, even seeing the differences between her answer and the correct answer, may still be unable to understand “why” she made the mistakes. For more experienced students this form of feedback may provide an exciting opportunity to explore the reasons of her mistakes.

7.8 Conclusion

This chapter has presented the principles, design, and functionalities of MIRA, a fully functioning implemented learning environment for the domain of Algebra that incorporates the Reflection Assistant Model, with pre and post-task reflective activities.

As presented in Section 7.2 MIRA maintains a certain independency of the domain of algebra. The description and step-by-step solutions of the algebra problems are kept in separate files and can be easily replaced by files with problems from another domain (e.g. physics or computer science problems). The quizzes are also kept separated in text files. The teacher can create new quizzes which will appear in the appropriate moments in MIRA. In the problem solving screen there are some resources to help the student to solve the algebra problems. Changes in the interface have to be undertaken in order to adjust these resources to another domain. For example, instead of having a button labeled “Algebra Basics” that opens an HTML file about algebra relational problems, the designer may create a button “Algorithms Basics” that will open an HTML file with information about algorithm structures. The only tool specifically designed for algebra is the PAL Tool. The programmer has to disable all JAVA classes related to the PAL Tool if using MIRA in a domain which is not algebra. Similarly some interface and programming changes will need to
be performed in the reflective activities. For example, instead of checking students’ understanding of the givens and relations of the problem in the pre-task reflection, a different version of MIRA could ask for understanding of another specific concept of the new domain or understanding of the problem in general.

A detailed description of the implementation of MIRA, including the structure of the log files, the directories and files organization and a UML diagram of the main JAVA classes can be found in Appendix B.

MIRA has been used in a laboratory experiment with undergraduate students, where students interacted with it individually for three sessions, answered questionnaires and performed pre and post tests. The next chapter details the design of the experiment and the analysis of the results.
Chapter 8
Experimental Study of MIRA

8.1 Introduction

This chapter presents the main study conducted with the MIRA learning environment and the analysis of the results of that study. The first part of the chapter describes the experimental design, the experimental hypotheses put forward, and the organization of the experiment with the materials used. The second part of the chapter presents a description of the general profile of the groups at the beginning of the treatment, including a profile of proficiency in algebra problem solving. The third part is dedicated to the statistical analysis of the experiment. We start by presenting the normality test done to select appropriate inferential tests selected to the experiment. We proceed then to testing the hypothesis. The chapter ends with the presentation of the limitations of the experiment and general conclusions.

8.2 Description of the Experiment

8.2.1 Objective

Previous chapters have discussed the difficulties of designing a metacognitive model for ILEs, not least because it is hard to test changes in metacognitive state, and therefore establish whether the model is effective. We have developed MIRA following a rationale that reflective activities that encourage students to think about their knowledge monitoring ability, to select metacognitive strategies and to evaluate their learning experience, are a good way to improve students’ awareness of the problem solving processes and have a positive impact on their learning gains. Given this, the aims of the experiment described here were to investigate:

1. Changes in students’ metacognition as a result of interacting with the reflective activities proposed in MIRA.

2. The effects of the pre-task reflective activities as a tool for improving students’ knowledge monitoring accuracy (KMA).

3. The effect of interacting with the reflective activities on students’ problem solving performance.
8.2.2 Experimental design

The experiment had a pretest-posttest design randomized between subjects. MIRA was used by two groups of subjects each associated with a different condition. The conditions allowed us to investigate the effects of the reflective activities on students’ problem solving performance and metacognitive gains. The experimental conditions are summarised in Table 8.1. The *experimental group* trained in the first condition, interacted with MIRA to solve problems *and* performed reflective activities. The *control group*, trained in the second condition, interacted with a version of MIRA where all reflective activities had been removed. These students had at their disposal all problem solving resources (PAL Tool, library of problems, timer, calculator, etc.) and passed through the problem solving stages in MIRA (see Figure 6.1 on page 100 with the sequence of domain-level activities).

<table>
<thead>
<tr>
<th>Condition 1</th>
<th>Condition 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Experimental group</em></td>
<td><em>Control group</em></td>
</tr>
<tr>
<td>Students interact with MIRA</td>
<td>Students interact with</td>
</tr>
<tr>
<td>to translate problems into</td>
<td>MIRA only to translate</td>
</tr>
<tr>
<td>equations and perform</td>
<td>problems into equations</td>
</tr>
<tr>
<td>reflective activities.</td>
<td>- no reflective</td>
</tr>
<tr>
<td></td>
<td>activity is included.</td>
</tr>
</tbody>
</table>

*Table 8.1: Experimental Conditions for MIRA’s evaluation. Pre-test/post-test design randomized between subjects.*

8.2.3 Hypotheses

For each of the conditions in Table 8.1 students were expected to demonstrate some gains in solving algebra word problems. However, each condition was expected to influence students’ learning gains and metacognitive changes to different degrees.

Students in the experimental group were expected to evolve more favourably than those in the control group with respect to performance and metacognitive behaviour. The hypotheses were as follows:

- Greater improvement in accuracy of knowledge monitoring for the experimental group as measured by KMA scores compared to the control group;
- Students in the experimental group exhibit better time management than those in the control group.
- The overall improvement in performance in problem solving in the experimental group would be higher then for the control group;
- Students in the experimental group give up fewer problems than students in the control group;

8.2.4 Participants

Between the 29th of October and the 22nd of November of 2002 twenty seven (27) subjects took part in the experiment on a voluntary basis and were paid £21 at the end of it. All subjects were first-year undergraduates taking either Psychology/Social-psychology or Geography/History Courses at the University of Sussex (UK).
Chapter 8. Experimental Study of MIRA

128

<table>
<thead>
<tr>
<th>Experimental Group</th>
<th>14 subjects</th>
<th>Training with MIRA with the reflective activities.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Group</td>
<td>13 subjects</td>
<td>Training with MIRA-Control: version of MIRA without the reflective activities.</td>
</tr>
</tbody>
</table>

Table 8.2: Description of the subject groups.

Unfortunately, two of the participants did not finish the experiment; one was from the experimental group and the other from the control group. Thus, at the end of the experiment we had collected data from 25 students (13 subjects in the experimental group and 12 in the control group).

The subjects were assigned randomly to one of the two conditions. Table 8.2 summarises the configuration of the groups at the beginning of the experiment. The subjects were not informed that there were two conditions in the experiment.

Initially, we intended to perform the experiment with adolescents from 13 to 15, mostly because MIRA involves algebra concepts which are taught at secondary level. However, organizing an experiment like this in a school proved to be very complicated (e.g., we needed at least 3 hours of each student on different days, etc.). So we decided to use first-year undergraduate students (freshers) from non maths-related courses. This group is also representative as they have learnt algebra at school, but have not necessarily mastered it and may still have difficulties in solving algebra word problems.

Metacognition theory affirms that the ability to reflect on one’s knowledge and strategies develops with age (Flavell, 1981). In fact, most theorists believe that metacognitive knowledge development begins at a young age, and continues at least throughout adolescence (Schraw and Moshman, 1995). Also, there is much evidence that suggests that metacognitive awareness and metacognitive skills continue to develop in adulthood (Baker, 1989).

Our experiment is more suitable for adolescents and adults than for children because it involves reading a number of explanations about metacognitive strategies and interpreting different graphs about metacognitive monitoring and control.

8.2.5 Tasks

Each participant taking part in the experiment was required to attend three sessions on different days no more than three days apart. Sessions lasted for a bit more than one hour each.

The first session was a familiarization session in which the subject was introduced to MIRA through a step-by-step demonstration of all stages and resources available within MIRA. The second session was a training session focused on the interaction with MIRA. The third and final session was a training and assessment session.

All subjects, regardless of their group, undertook the same tasks in the same order and with the same time-length. Table 8.3 summarises the organization of the sessions. A detailed description of the sessions follows.
### Session 1: Familiarization (66 min.)

<table>
<thead>
<tr>
<th>Duration</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 min.</td>
<td>Read instructions and fill in consent form</td>
</tr>
<tr>
<td>3 min.</td>
<td>Answer questionnaire about strategies</td>
</tr>
<tr>
<td>13 min.</td>
<td>Perform pretest: part I and II</td>
</tr>
<tr>
<td>3 min.</td>
<td>Read introductory text about algebra components (givens, goals, unknowns, relations, and equations) used in MIRA</td>
</tr>
<tr>
<td>40 min.</td>
<td>Use MIRA - the experimenter guides the student through MIRA for Problem 1</td>
</tr>
<tr>
<td>6 min.</td>
<td>Participate in a semi-structured interview</td>
</tr>
</tbody>
</table>

### Session 2: Training (60 min.)

<table>
<thead>
<tr>
<th>Duration</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 min.</td>
<td>Sign in and read instructions for session 2</td>
</tr>
<tr>
<td>51 min.</td>
<td>Use MIRA</td>
</tr>
<tr>
<td>7 min.</td>
<td>Participate in a semi-structured interview</td>
</tr>
</tbody>
</table>

### Session 3: Training & Assessment (69 min.)

<table>
<thead>
<tr>
<th>Duration</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 min.</td>
<td>Read instructions for session 3 (same as for session 2)</td>
</tr>
<tr>
<td>40 min.</td>
<td>Use MIRA</td>
</tr>
<tr>
<td>18 min.</td>
<td>Perform post-test: part I and II</td>
</tr>
<tr>
<td>4 min.</td>
<td>Review questionnaire about strategies</td>
</tr>
<tr>
<td></td>
<td>Fill in MIRA assessment form</td>
</tr>
<tr>
<td>6 min.</td>
<td>Participate in a semi-structured interview</td>
</tr>
</tbody>
</table>

**Table 8.3:** Organization of the experiment sessions.
**The Familiarization Session**

First, the subject read instructions about the experimental procedures and signed a consent form. If the subject agreed to continue the experiment, she filled in a form with demographic questions (see Appendix D). The subject was then given a questionnaire about strategies and, immediately thereafter, a pre-test (see Section 8.2.6 for details about these materials). The pre-test was divided into two parts, where part one could take a maximum of 3 minutes and part two, 10 minutes.

After that, the subject worked at the computer and read an introductory text about the algebra concepts trained in MIRA. The text was available via a link from the main menu of MIRA (a copy of this text can be found in Appendix D). The subject was instructed to take her time to read the text and ask questions for clarification if she needed to. Meanwhile the experimenter checked the subject’s answers for the pre-test and computed the KMA and KMB scores and annotated them on an observation sheet (see an example of this instrument in Appendix D).

The subject then received a tutorial on paper with step-by-step instructions and drawings of the screens in MIRA. The tutorial contained simple instructions on how to register, and some tips on how to edit and delete content (see a copy of the tutorial in Appendix D). The instructions in the tutorial were different for the control group just for the parts that did not exist in the MIRA-Control version. The experimenter instructed the subject to follow the tutorial to register. To complete the registration, the subject needed her login number and the KMA and KMB scores in the pre-test. The experimenter provided this information.

Immediately after registering, the subject started interacting with MIRA under the guidance of the experimenter. All subjects received the same instructions on how the environment worked, the options it contained and what was expected from them for answering the problems. The experimenter also taught the students how to use the PAL Tool. When the first problem was done, the experimenter left the subject alone to continue the training.

The time allocated for the training in the first session was 40 minutes. At the end of this period a message automatically popped-up in MIRA and the subject was logged out. Before leaving the subject was asked some questions about the session.

**The Training Session**

When subjects arrived for the second session they first read the instructions for that session (a copy of the instructions is found in Appendix D) before logging into MIRA. The interaction with MIRA in this session was the longest, lasting 51 minutes. When the time was up, the subject answered some questions in a semi-structured interview, similar to the one in the first session.

**The Training & Assessment Session**

In the third and final session, the subjects started by interacting with MIRA for 40 minutes. Immediately after, the subject was given a post-test. Similarly to the pre-test, it was divided into two parts. The subject had about 3 minutes to complete part one and a maximum of 15 minutes for part two. The subject was then given the same questionnaire about strategies that she had completed at the beginning of the experiment for a second attempt at it. After finishing, the subject was given an evaluation form about the resources in MIRA. Finally, the subject answered a semi-structured interview.
8.2.6 Materials
Appendix D contains a complete list and samples of all materials used in the experiment.

Experimental Setup
The room setup allowed the experimenter to observe the subjects’ actions in MIRA without interfering with it. The student used a monitor and keyboard connected to the experimenter’s laptop computer but a standard office division panel separated the experimenter and the subject during the session, as shown in Figure 8.1.

![Room setup for the experiment. The experimenter was able to observe the interaction with MIRA without disturbing the subject.](image)

**Figure 8.1:** Room setup for the experiment. The experimenter was able to observe the interaction with MIRA without disturbing the subject.

Pre-test and post-test
These tests were designed to measure knowledge of algebra problems translation and knowledge monitoring skill (KMA and KMB) similar to Tobias & Everson’s assessment instrument. The tests
were divided into two parts:

1. First the participants read the algebra problems and predicted if they could solve them correctly or not. For each problem they had to answer the question: “Do you think you can give the answer that is asked for this problem?” by yes or no. Participants were given 3 minutes for completing this phase of the test. The time was enough for all participants.

2. Immediately after, the participants received part two, where they had to translate the same problems, writing down an expression or equation as answers (in a similar way as in MIRA). This part was restricted to 10 minutes in the pre-test, and 15 minutes in the post-test. These times were based on pilot observations and on the fact that the post-test was more difficult.

The pre-test and post-test were devised with the aid of a maths teacher, who was asked to include a mixed set of easy, medium and difficult algebra problems. Both pre- and post-test had five problems in total.

The scoring system was as follows: 2.0 points for each correct answer; 1.0 point for partially correct answers; 0.5 for incomplete answers or answers with serious mistakes; and 0.0 for blank or wholly incorrect answers. A scheme with several possible answers and common mistakes was made to help marking the tests.

We decided that the post-test should be a bit harder than the pre-test, given that the subjects would have gained more practice in solving those kinds of problems during the treatment. Also we were interested in comparisons between the experimental and control groups and not in absolute measures of performance improvement in either group separately.

The problems in the pre-test required students to write one expression or equation (thus, manipulating a maximum of two variables), whereas in the post-test they were required to write more complicated systems of linear equations (with multiple variables). Unfortunately, as we will show later on in this chapter, the increase in difficulty was too high. That was not our intention.

**Questionnaire about strategies**

This questionnaire was applied at the beginning and again at the end of the experiment. The aim of the questionnaire was to gather information about the strategies students could remember and those they normally used when solving mathematical problems. It was inspired by an instrument devised by Hartman (2001a) in which students had to define strategies and reflect on their use in reading a given text. Similarly, our questionnaire does not contain statements like *when I solve a problem I organize my time in order to best accomplish my goals*, because such statements can influence students’ answers. Instead, subjects had to recall the strategies they normally used and write them down.

It worked as follows: in the first session, the subject received the questionnaire and was asked to make a list of all the “things” she considered important to think about, to do, etc. when one is about to solve a maths problems. The word “strategies” was purposefully not used. We intended to leave the subject free to write down whatever she considered important without guiding her to specific issues. After making a list, the subject was asked to tick those she reckoned she often used. In the final session the subject was given the same questionnaire, this time with her list attached to it. She was asked to: (i) include any new strategies she could remember and, (ii) tick in the opposite column the strategies she remembered that she used in MIRA in any of the sessions. These instructions were given verbally.
So, four kinds of information were gathered with this instrument:

1. Strategies students considered relevant for problem solving.
2. Strategies students normally used in problem solving.
3. Additional strategies students considered relevant after the interaction with MIRA.
4. Strategies students remembered having used during the interaction with MIRA.

**MIRA assessment questionnaire**

This questionnaire was designed to measure the subjects’ perception of helpfulness of MIRA’s resources (e.g. the PAL Tool, the library of problems, the Quiz, etc.) and the frequency of use of these resources on a 5-point scale. There were two types of questionnaires, depending on the experimental condition - one included the reflective activities and the other covered only the problem solving resources.

Answers to this questionnaire provided additional information about students’ feeling regarding the interaction with MIRA, complementing the information collected in the electronic logs.

### 8.3 Preparation for Data Analysis

We used SPSS (version 10) to perform the exploratory analysis, descriptive analysis, and inferential analysis of the data. We have adopted a significance level of 5% \((\alpha=0.05)\) in our analysis.

Note that not all graphs and tables created were included in this chapter to maintain the flow of the argument and improve readability. Instead, we have included them in Appendix C for further consultation.

#### 8.3.1 Data Normality

Shapiro-Wilk tests of normality were performed on the main variables (e.g. pre-test scores, post-test scores, GCSE scores, age, KMA in pre-test, KMA in post-test, etc.). These revealed that, possibly owing to the small sample size \((N=25)\), most variables were not normally distributed (see results in Table C.1 in Appendix C). The non-parametric Mann-Whitney U and Wilcoxon tests were therefore used to investigate between group and repeated measures differences, respectively. In order to avoid the possibility of Type II errors occurring due to the use of these less sensitive non-parametric tests, t-tests were also used on the normally distributed variables (Cramer, 1998).

#### 8.3.2 Categorizing data for analysis purposes

Some of the ordinal data collected was regrouped into a smaller number of categories. This categorization was necessary for group comparisons, analysis of frequencies, and other types of analysis. The categories are described below.

**Age**

We divided the students into standard-age students, younger mature students and older mature students. A “mature” student is defined as someone over 21 years old on entry at university. Because the mature students are a wide and heterogeneous group, we divided it into younger and older mature students. The table below summarises the age categories:
Chapter 8. Experimental Study of MIRA

<table>
<thead>
<tr>
<th>Age ranges</th>
<th>Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>under 22</td>
<td>standard-age students</td>
</tr>
<tr>
<td>22-29</td>
<td>younger mature students</td>
</tr>
<tr>
<td>over 30</td>
<td>older mature students</td>
</tr>
</tbody>
</table>

**GCSE Maths Grades**

This classification was created to map the information provided in the demographics form about GCSE scores in general maths into numerical values. We converted the alphabetical grades into numerical ones for analysis purposes.

<table>
<thead>
<tr>
<th>GCSE grades</th>
<th>Equivalent Scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>A*</td>
<td>8</td>
</tr>
<tr>
<td>A</td>
<td>7</td>
</tr>
<tr>
<td>B</td>
<td>6</td>
</tr>
<tr>
<td>C</td>
<td>5</td>
</tr>
<tr>
<td>D</td>
<td>4</td>
</tr>
<tr>
<td>E</td>
<td>3</td>
</tr>
</tbody>
</table>

**Problem Solving Performance**

This classification was used for the analysis of performance in algebra problem solving. It divided the data sample into three groups: low, average and high problem solvers. This categorization was used in the following measures: pre-test, post-test and problems solved in MIRA.

<table>
<thead>
<tr>
<th>Score ranges</th>
<th>Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0 - 8.0</td>
<td>high algebra problem solver (stronger)</td>
</tr>
<tr>
<td>7.9 - 5.0</td>
<td>average algebra problem solver</td>
</tr>
<tr>
<td>4.9 - 0.0</td>
<td>low algebra problem solver (weaker)</td>
</tr>
</tbody>
</table>

**Metacognition: KMA and KMB**

For the analysis of the experiment we used the same categories defined for the KMA and KMB values described in Chapter 4 in Tables 4.2 and 4.4.

### 8.3.3 Definition of measures of performance

To minimize any problems in the analysis resulting from initial differences found in the groups, we decided to measure performance changes using the absolute and relative differences between post-test and pre-test scores. Similarly, we used the absolute difference for analysing KMA changes.

- the **absolute difference** between the post-test and pre-test is calculated by:
  \[ \Delta \text{score} = (\text{PosttestScore}) - (\text{PretestScore}) \]  
  \[ (8.1) \]

- the **relative difference**, which emphasizes gains for initially weak participants is calculated by:
  \[ \frac{\Delta \text{score}}{\text{score}} = \frac{(\text{PosttestScore}) - (\text{PretestScore})}{(\text{PretestScore})} \]  
  \[ (8.2) \]
• the **absolute difference** between the KMA measured in the post-test and the KMA measured in the pre-test is calculated by:

\[
\Delta KMA = (\text{posttest}KMA) - (\text{pretest}KMA) \tag{8.3}
\]

### 8.4 Exploratory Analysis of the Experiment

We performed an exploratory analysis of the data prior to the hypotheses testing. This exploratory analysis helped to create a profile of the groups and to make initial observations of the groups’ algebra problem solving performance before the treatment.

#### 8.4.1 General profile of the experimental group

The subjects consisted of 13 first-year undergraduate students, where 76.9% were females and 23.1% were males; their ages ranged between 18 and 38 years old, with a mean of 23.85 years (stdev=6.998). Using our age categorization, 38.5% were in standard-age, 38.5% were younger mature students and 23.1% were older mature students.

The selected responses for the “Motivation for maths problem solving” item in the demographic form were distributed as follows: the majority of 76.9% selected fairly motivated, 15.4% very motivated and 7.7% chose extremely motivated. Thus, no one in this group selected the not at all motivated or a bit motivated options in the 5-point scale.

In terms of familiarity with computer learning applications, 38.5% classified themselves as unfamiliar, whereas another 38.5% stated an average familiarity. The rest were newcomers (7.7%) or beginners (15.4%).

With respect to GCSE grades, 7.7% got an A*, 38.5% got an A, 15.4% got a B and 38.5% got a C. In this group five subjects had A Levels in Maths (38.5% of the sample). Of these A Levels students, 80% had a B or a C, and the remainder 20% had an A. Please, refer to Table C.2 for details of the frequencies for this group.

#### 8.4.2 General profile of the control group

The control group, which consisted of 12 undergraduate students, had a very similar gender distribution compared to the experimental group: 75% were females and 25% were males. Ages ranged from 18 to 27 years old, with a mean age of 20.83 years (stdev=3.326); 66.7% fell into the category of standard-age students and 33.3% in the younger mature students category. There were no older mature student in this group.

In terms of “Motivation for maths problem solving” it was found that 41.7% considered themselves fairly motivated, 33.3% a bit motivated and the remainder, not at all motivated (8.8%) or very motivated (8.8%). One participant failed to inform us about her motivation.

With respect to familiarity with computer learning applications, 47.1% classified themselves as beginners, 25% as average, another 25% as unfamiliar, and 8.3% as newcomers.

With respect to the maths GCSE grade, 16.7% got an A, 25% got a B and 58.3% got a C. Only 1 subject had A Levels in Maths (representing 8.33%); she had a C grade. Table C.3 presents all frequencies for the control group.
8.4.3 Comparison of groups’ general profile

Figure 8.2 presents graphs comparing the groups’ general profiles. Looking at graph (a) we notice that the groups are quite different in age: the control group is much younger with 66.7% of its participants as standard-age students, whereas 77% of the experimental group is evenly spread between the standard-age and younger mature students (18 to 29 yrs).

The motivation for maths problem solving (b) is also quite different, as the participants in the experimental group present themselves as fairly motivated (n=10) or higher, whereas the majority in the control group are distributed between fairly (n=5) or a bit (n=4) motivated with one participant not at all (n=1) motivated. The familiarity with learning applications (c) varied only a bit, with the majority of participants in both groups (n=8) ranging from unfamiliar to beginner.

Regarding their GCSE grades the groups are also quite different from each other (d). The control group is weaker with almost 60% of the subjects with C grades. The majority of the experimental group has either A or B grades. Also, in the experimental group we had 5 subjects with A Levels in maths against only one subject in the control group.

8.4.4 Groups initial profile for algebra problem solving

Initially, we expected to have similar groups in terms of algebra problem solving abilities, considering that they were taken randomly from the same population. To analyse the profile of the groups for algebra problem solving abilities at the beginning of the treatment we looked at their pre-test scores.

The tables on Figure C.1 (a) show the descriptive statistics values for both groups in the pre-test. Looking at the median and mean scores we can observe that the control group had a slightly better performance than the experimental group (Control Mean=5.5 and Experimental Mean=4.96). The standard deviation for the control group was greater than the one of the experimental group (Control stdev=2.68 and Experimental stdev=1.61), which shows a much smaller variability in the scores for the experimental group. Therefore, the experimental group was more homogeneous than the control group in their scores for the pre-test.

The bar graph on Figure C.1(b) shows the graphs of frequencies organized by categories. A bit more than 60% of the subjects in the experimental groups fell into the average problem solver group against 33.3% in the control group. Moreover, there was no one in the experimental group who was categorized as a high algebra problem solver, whereas 25% (3 subjects) of the control group fell into this category. The low algebra problem solvers set was almost identical for both groups.

Although with some evident differences in the pre-test scores distribution, we cannot reject the hypothesis that the groups come from the same population. In order to test this hypothesis we performed the inferential Mann-Whitney U test (Table C.4) and found that there is no statistical significant difference between the groups’ pre-test means ($p=0.57 > \alpha$).

So, in spite of some observed differences in the pre-test scores, from a statistics point of view the differences between the experimental and control pre-test scores were not significant.

Considerations about the groups’ GCSE maths and algebra proficiency

If we take the groups’ GCSE scores we will observe that there is a statistically significant difference between the groups (Table C.5). And we believe that this difference is due to the subjects
with A Levels in maths - they were very strong compared to the rest of the group. It was not expected as we have assigned the subjects randomly to the groups. Performing another test without those subjects, we found that the statistical differences between the experimental and control group disappear (see test in Table C.5).

However, we need to draw attention to the fact that even with more students with A Levels in maths, the experimental group did worse in the pre-test. We could expect that it would be the opposite, but the control group was actually stronger in the pre-test.

Another curious observation is that no correlation was found between the GCSE scores in maths and the pre-test scores (see tests in Table C.6). This information suggests that the GCSE scores were not a good predictor of the pre-test scores, and possibly of the performance on the problems proposed in MIRA; one possible explanation is that the translation task proposed in the pre-test (and also in MIRA) were new to the students. This conclusion is supported by several
comments from students in the interview related to the novelty of the translation task which the experiment asked them to do.

Therefore, in the analysis of learning gains we will concentrate our statistical tests on the pre-tests and post-tests, rather than on the GCSE grades.

8.5 Testing Hypotheses related to Problem Solving

8.5.1 Overall performance

The experimental hypothesis was that the increase in performance of the group who received the reflective activities would be greater than that of the other group. To test this hypothesis we had at our disposal two sources of information:

1. the groups’ performance in pre- and post-tests, which can be analysed looking at:
   - descriptive values of pre- and post-tests scores.
   - the absolute and relative differences of the post- and pre-tests scores ($\Delta score$ and $\frac{\Delta score}{score}$).

2. the groups’ performance in MIRA.

Descriptive analysis of pre- and post-tests

Looking at the values in the Table C.7 for pre- and post-tests scores we observe that:

- For both groups, there was a drop in the mean and median scores between pre- and post-test. This was unexpected, because we predicted that the students would score better after having been exposed to problems in MIRA. However the pre-test and the post-test were not designed to be of the same difficulty: we knew the post-test was a bit more difficult than the pre-test, but we thought that it would be compensated by the fact that students would have done many similar problems with MIRA. However the difference in difficulty was greater than we thought and, as a result, scores dropped on average.

- For the control group, even with a decrease in the median score (with the pre-test Median=5.75 and post-test Median=4.5) we can observe an increase in the upper quartile (pre-test 75th percentile=7.63 and post-test 75th percentile=7.88) and a decrease in the scores variability (pre-test variance=7.18 and post-test variance=6.43), which signifies a performance improvement of some subjects and a greater uniformity of scores in the post-test. The same cannot be said of the experimental group. In fact, a large increase of variability of scores was observed for the experimental group (pre-test variance=2.60 and post-test variance=5.80).

We have plotted the pre-test and post-test data points into comprehensive graphs that show individual scores as well as the means of groups, the numbers falling in each category (low, average, and high) as well as boxplots for data. Figure 8.3 shows the graph for the experimental group. Looking at it we can see that in the pre-test the scores were more concentrated (between 3.5 and 6.0) than in the post-test. The “cloud” shows a group of four subjects positioned below the median and above the lower quartile. This group as a whole had only a slight movement downwards. The two subjects with the lowest scores in the pre-test also had the lowest score in the post-test, keeping their ranking position. Apart from three subjects, all the others presented a decrease in their scores in the post-test.
Looking at the same graph for the control group (Figure 8.4), we observe that more subjects demonstrated an increase of their scores than in the experimental group. Looking at the post-test boxplot we can see that the area between the median and the upper quartile got larger. Differently from the experimental group, this group had subjects classified as high problem solvers in both pre- and post-tests.

One subject in the experimental group had a dramatic decrease in her score, which was an atypical change. In the same group another subject presented an atypical increase of score. In the control group, one subject presented a relative increase of score much higher than the rest of the group - it was the subject with the lowest score in the pre-test (with only 0.5 points) that moved to the lower quartile in the post-test (reaching 3.5 points).

**Pre- post-tests absolute and relative differences**

Figure 8.5 shows the mean scores for both groups in pre- and post-test, together with the absolute and relative differences of scores ($\Delta score$ and $\frac{\Delta score}{score}$). The graph excludes the three outliers which were clearly not representative of their groups.

Looking at the mean of the absolute and relative differences, we see that the control group had a slightly better performance than the experimental group.

However, the Mann-Whitney tests performed detected no significant difference between the groups’ means of differences of scores (results in Table C.9). Similarly, the Wilcoxon test did not detected significant difference between the pre-test and post-test scores in each group, as shown in Table C.8.

In summary, if the metacognitive activities have caused changes in the performance difference between pre-test and post-test, they are not reflected in these results. A closer look at the students’ performance during the interaction with MIRA becomes essential to draw a better conclusion.
Chapter 8. Experimental Study of MIRA

Figure 8.4: Graphical view of pre-test and post-test scores for control group: evolution of individual performance for all students.

Figure 8.5: Graphical view of post-test pre-test differences: mean of group scores.
Chapter 8. Experimental Study of MIRA

Performance in MIRA

Looking at the groups’ problem solving performance during the interaction with MIRA we found a different scenario.

Each student answered a certain number of problems in MIRA. To analyse their problem solving performance we have classified each problem solving episode according to two sources of information from MIRA:

(i) Whether the answer chosen in the multiple-choice was correct or not (variable: correct-answer)
(ii) The qualification the student gave of her answer, through the explanation chosen in the multiple-choice (variable: explanation). The possible multiple-choice values for the explanation can be found in Section 7.7.2.

We created a marking scheme to categorize students’ responses to the problems in MIRA based on these two pieces of information. The possible values in our scheme were: correct answer, answer almost correct (with minor errors), own answer is wrong but was able to recall or recognize the correct one in the multiple-choice, and wrong answer chosen. The algorithm devised to generate these categories is presented below.

\[
\text{If answer-chosen} = \text{not(correct-answer)} \text{ then}
\]

\[
\text{solution} \leftarrow \text{“wrong”}
\]

\[
\text{else}
\]

\[
\text{If answer-chosen} = \text{correct-answer} \text{ then}
\]

\[
\text{If explanation} = \text{“own answer and answer chosen are equivalent” then}
\]

\[
\text{solution} \leftarrow \text{“correct”}
\]

\[
\text{else}
\]

\[
\text{If explanation} = \text{“own answer and answer chosen have some differences” then}
\]

\[
\text{solution} \leftarrow \text{“correct with minor errors”}
\]

\[
\text{else}
\]

\[
\text{If explanation} = \text{“own answer is wrong, but recognized correct one” then}
\]

\[
\text{solution} \leftarrow \text{“recall correct”}
\]

We predicted that the experimental group would solve fewer problems than the control group, because both groups had the same amount of time to interact with MIRA, and the former had to solve problems and perform reflective activities, while the latter just had to solve problems.

Indeed, we found that the experimental group attempted 112 problems compared to 136 for the control group. As the Mann-Whitney test performed shows (Table C.10), the number of problems attempted by the experimental group was highly significantly smaller than that of the control group (z=2.56, \(p=0.005 < \alpha = 0.01\)).

At the same time, the experimental group had a better performance in MIRA than the control group. Figure 8.6 shows pie charts with the groups’ performance. We can observe a big difference between the groups in terms of the percentage of correct answers provided; the experimental group with 41.96% of the answers provided being correct, whereas the control group with only 26.47%.

The Mann-Whitney test performed (summarised in Table C.11) shows that the number of correct answers per total of problems attempted of the experimental group was significantly greater.
than that of the control group \((z=1.66, p\ (1\text{-tailed})=0.048)\). It is also the case for the number of answers almost correct (with minor errors) per total of problems attempted \((z=1.82, p\ (1\text{-tailed})=0.04)\). As predicted, the experimental group had a significantly better performance than that of the control group.

For the other types of responses (wrong answers and answers recognized as correct) no statistical significant difference between the groups’ means was detected (Table C.11).

### 8.5.2 Perseverance in problem solving

The hypothesis related to perseverance was tested looking at the number of problems given up for each group. We predicted that the subjects in the experimental group would persevere more in problem solving, giving up on fewer problems than the control group.

Looking again at Figure 8.6 we can observe that the control group gave up on a slightly higher percentage of the problems than did the experimental group (control=19.12% and experimental=14.29%). However, this difference is not statistically significant (Table C.12).

### 8.6 Testing Hypotheses related to Metacognition

#### 8.6.1 Time management

All students had the same amount of time to attempt to solve each problem (15 min.). At any point during the problem solving period they had the option to see a clock showing the remaining time left. At the end of the 15 minutes, a pop up window appeared and notified them that the time was up. The student had then to state whether or not she had an answer to the problem.

We hypothesized that the experimental group would be able to better monitor their problem solving time than the control group, finishing the problem within the available time and avoiding the time-up message.

In Figure 8.7, we have divided students according to the number of problems in which they had to be warned by a time-up message. We see for instance that more than 60% of the students in
the experimental group solved all their problems within the specified time (0 problems timed out); it happened to only 8.3% of students in the control group. This fact together with the overall distribution of the bars clearly shows that the experimental group exhibited better time management.

**Figure 8.7:** Number of problems in which students have seen the time-up message.

The inferential Mann-Whitney test performed (Table C.13) shows that the proportion of problems where students saw the time-up message in the experimental group is marginally significantly smaller than that of the control group ($z=1.67$, $p$-value (1-tailed)=$0.06<\alpha=0.10$).

### 8.6.2 Changes in knowledge monitoring accuracy (KMA)

We have hypothesized that the increase in knowledge monitoring accuracy (KMA) of the group with the reflective activities would be greater than that of the other group. To test this hypothesis we used information from measures of KMA from pre- and post-tests. We believe that the KMA values measured in the post-test were not distorted by the fact that it was harder than the pre-test. Because the KMA measures the ability to assess difficulty, it shouldn’t be affected by the degree of difficulty of the problems.

**Descriptive analysis of KMA**

The descriptive values are presented in Figure C.14 in Appendix C. Below we have a table that summarises the mean KMA scores for the groups.

<table>
<thead>
<tr>
<th></th>
<th>Experimental</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>KMA in pre-test</strong></td>
<td>-0.05 (SD=0.42)</td>
<td>0.20 (SD=0.38)</td>
</tr>
<tr>
<td><strong>KMA in post-test</strong></td>
<td>0.14 (SD=0.56)</td>
<td>0.13 (SD=0.41)</td>
</tr>
</tbody>
</table>

**Table 8.4:** Mean KMA Scores by test and group - Data interval=[-1.0, 1.0]

In the beginning of the treatment the mean score of the KMA of the experimental group was much smaller than the mean KMA of the control group. Looking at the values in the post-test we can observe that the experimental group had a higher increase in the KMA mean scores than the control group as predicted. One outlier value was found in the experimental group for the KMA
in the pre-test (with value of KMA in the pre-test equals to -1) and this outlier value is included in the table above.

**Changes in KMA categories**

Looking at the distribution of KMA organized by category (low, average and high) the groups presented the following distribution (see details in Table C.15):

- At the beginning of the experiment 75% of the students in the control group were classified as having an average KMA, and the remaining 25% had a high KMA. We can observe a redistribution of these percentiles at the end of the experiment, where 8.3% presented a low KMA, 58.3% had an average KMA, and 33.3% a high KMA.

- For the experimental group, the distribution at the beginning of the experiment was: 76.9% had an average KMA, 15.4% a low KMA and 7.7% presented a high KMA at the beginning of the experiment. This distribution changed at the end of the experiment, showing an increase of the high KMA group with 30.8% of the students falling into this category. There was also an increase of low KMA students (23.1%) and a decrease of average KMA students, representing at the end of the experiment 46.2% of the group.

It is interesting to notice the rise of the high KMA students in the experimental group as seen in Figure 8.8.

Unfortunately, according to the Wilcoxon tests performed for differences in the KMA as measured in the pre-test and post-test scores of each group, there were no significant improvements in the experimental group KMA scores (results for both groups are reported in Table C.16).

Nevertheless, it is important to draw attention to a relevant fact: in the experimental group at the beginning of the experiment only one (1) participant presented a “high KMA” (KMA value=0.6); at the end of the experiment this participant preserved her KMA level (i.e. high KMA), however increasing her score (KMA value=1.0). Furthermore, another three (3) participants were upgraded to the high KMA level.

The control group, had already three (3) subjects with high KMA (KMA values=0.6) at the beginning of the experiment. Two of them preserved their level, although their KMA scores did not increase. Two other students upgraded to high KMA level. Table C.17 shows the values of this crosstabulation.
Differences of KMA scores between groups

Figure 8.9 presents the means of KMA in both pre- and post-test, as well as the $\Delta KMA$ for both groups. This graph excludes the outlier identified in the experimental group, in order to better illustrate the groups’ mean situation - the mean values are very sensitive to outliers, which can distort the view of the group’s tendency (Field and Hole, 2003).

This outlier was a subject who scored -1 for the pre-test KMA (suggesting an extremely low KMA) and then scored 1 in the post-test KMA (meaning an extremely high KMA). Here it is important to explain the nature of this change: this participant scored a high KMA at the end because she predicted that she could not do any of the problems in the post-test correctly and indeed she either did the problems wrong or she left them blank.

The graph shows that the control group presented a negative $\Delta KMA$, suggesting a decrease of KMA (very small though). On the other hand, the experimental group presented a positive $\Delta KMA$, suggesting an increase of KMA.

![Figure 8.9: Mean values of KMA for the groups: mean values excluding outliers.](image)

However, the groups’ KMA scores are not statistically significantly different, as demonstrated by the Mann-Whitney test performed (see results in Table C.18).

8.6.3 Changes in knowledge monitoring bias (KMB)

We had no a priori expectation for the evolution of the KMB. It must be emphasized that some amount of change in KMB takes place as a result of changes in KMA. This is because the two measures are not independent: a greater accuracy in knowledge monitoring (increase of KMA) must be accompanied by a diminution of the bias (reduction of the KMB magnitude).

One aspect of KMB change which can, however, be examined independently from the KMA is the stability of the sign of the KMB for any given student. This is equivalent to asking whether people maintain their characteristics as pessimists or optimists even as they improve in accuracy.

Table 8.5 shows the crosstabulation of KMB changes for each of the groups separately. If
Chapter 8. Experimental Study of MIRA

we look first at the experimental group, we see that of the 9 optimists which the group originally contained, the great majority (7) remained optimists. Of those 9, 2 became realists which is indicative of their progression in accuracy. The number of optimists remained constant at 9 due to the fact that the 2 students which had no consistent bias in their assessment (random) became optimists. The number of realists went up another unit due to the fact that one pessimist became realist as a result of greater accuracy. The other one was and remained realist. Overall we see that a minority of people changed categories. Of the 5 who did change categories, 3 entered the realistic category and 2 went from random to optimists.

Table 8.5:

<table>
<thead>
<tr>
<th>Group: Experimental</th>
<th>KMB post-test</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Optimist</td>
<td>Realist</td>
</tr>
<tr>
<td>KMB pre-test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pessimist</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Optimist</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>Realist</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td>9</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Group: Control</th>
<th>KMB post-test</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Random</td>
<td>Optimist</td>
</tr>
<tr>
<td>KMB pre-test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Optimist</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Realist</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 8.5: Table of frequencies: pre-test KMB vs. post-test KMB.

In the control group, 5 of the 12 subjects remained in their original category (3 optimists and 2 realists). The optimist category lost 2 subjects to the random group but gained another 2 back from that same group. A similar zero-sum exchange of 1 student happened between the optimist and realist groups. In addition, one random individual became realist. Therefore, this group, like the other one, shows quite a high stability in those categories. Only 2 students have changed category in a way that betrays some instability: from optimist to random. The significant finding, though is that we never observe any transition from pessimist to optimist or vice-versa.

This pattern of change suggests that the KMB captures something quite fundamental about the student’s metacognitive profile and validates the use of the KMB as a tool for automatic student categorization. This information should be usable in powerful ways to guide the metacognitive training. In MIRA, we only reflect this information back to the student. But it should be possible to use it to differentiate better the treatment that the student receives from the ILE.

8.7 Observations about Correlations

We computed a range of correlations, in order to infer causality or find predictor factors. We used the Spearman’s non-parametric correlation test and scatter graphs for this purpose (they can be found in Appendix C). The main correlations computed were:

- Correlation between pre-test and post-test scores: we would like to know if the scores in the pre-test were a good predictor of the scores in the post-test. We have found that it was true
Correlation between pre-test KMA and pre-test scores: previous research on KMA and problem solving (Tobias and Everson, 2002) indicates that the KMA levels are a good predictor of performance. In our experiment we found a positive correlation, considering all participants (N=25), with a *p*=0.01. However, looking at the samples separately, only the experimental group shows this positive correlation (*p*=0).

Correlation between post-test KMA and post-test scores: it was intriguing to notice that there was no longer a correlation in the experimental group. A closer look showed that the correlation could be restored if we left out four specific subjects. We looked for a pattern in the evolution of these subjects but we did not find any. Two increased their KMA score without increasing their performance; one increased her performance without increasing her KMA and one decreased her performance while increasing her KMA.

We also tried to correlate motivation in maths with several variables: pre-test scores, \(\Delta score\), \(\Delta KMA\), KMA in pre-test, and \(\Delta KMA\). The Spearman test shows a positive correlation between motivation in maths and pre-test scores only for the control group. All the rest did not show a significant correlation with motivation in maths.

### 8.8 Analysis of interaction with MIRA

#### 8.8.1 Evaluation of resources in MIRA

At the end of the experiment all participants filled in a 5-point scale evaluation form about the resources of MIRA (Appendix D).

The questionnaire focused on the degree of helpfulness and frequency of use of the resources. Both groups answered the items about the problem solving resources and the experimental group had some additional items about the reflective activities. Below we present the results of the evaluation together with some comments the students made during the interviews.

**Students’ evaluation of the problem solving resources**

The *Teacher’s Solution* and the *PAL Tool* were considered the most helpful resources. Their mean scores were above the “consistently helpful” level. All the other resources were rated between “consistently helpful” and “sometimes helpful”.

In the interviews the PAL Tool was frequently mentioned. Students commented that it made them organize their solutions in a novel way. Remarks like the following ones were common:

**Researcher:** Did you find any activity useful? Which one?

*p03e:* “The PAL tool for working out, because I wouldn’t normally. I would work it out but I wouldn’t normally sit and look at the givens, the unknowns and the goals, and that was really helpful, it made it much clearer”.

**Researcher:** Which activity was most helpful for you to move further when you faced difficult problems?

*p07e:* “The PAL system, which helped separate the equations. [...] Just to separate it into parts and making you give variables and name them separately”.

It is interesting to notice that the least appreciated resource was the *Library of Problems*. Contrarily to what is commonly stated in the Problem Solving Literature, students did not use previous...
problems performed as a source of aid for similar problems (MIRA problems were organized into four types with similarities between them). The Library of Problems also included teacher’s solutions of all problems done so far. This suggests that the participants used the teacher’s solution only to check their own solution in the current problem and did not regard the teacher’s solutions as a general resource that could help them in future similar problems. Table 8.6 shows the mean scores.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Experimental Group</th>
<th>Control Group</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Helpfulness</td>
<td>Freq. Use</td>
<td>Helpfulness</td>
</tr>
<tr>
<td>Teacher’s Solution</td>
<td>3.42</td>
<td>3.50</td>
<td>3.50</td>
</tr>
<tr>
<td>PAL Tool</td>
<td>3.18</td>
<td>3.08</td>
<td>2.92</td>
</tr>
<tr>
<td>Quiz</td>
<td>2.83</td>
<td>3.08</td>
<td>3.09</td>
</tr>
<tr>
<td>Calculator</td>
<td>3.00</td>
<td>2.50</td>
<td>2.80</td>
</tr>
<tr>
<td>Answer to the problem</td>
<td>2.83</td>
<td>2.92</td>
<td>2.75</td>
</tr>
<tr>
<td>Timer</td>
<td>2.60</td>
<td>2.82</td>
<td>2.73</td>
</tr>
<tr>
<td>Algebra basics</td>
<td>2.33</td>
<td>1.83</td>
<td>2.50</td>
</tr>
<tr>
<td>Glossary of relations</td>
<td>2.11</td>
<td>1.58</td>
<td>2.20</td>
</tr>
<tr>
<td>Library of problems</td>
<td>1.83</td>
<td>1.50</td>
<td>2.40</td>
</tr>
</tbody>
</table>

Table 8.6: Students’ evaluation of problem solving resources in MIRA.

Students’ evaluation of the reflective activities

The experimental group rated the reflective activities as a whole as “sometimes helpful”. In the interviews students made comments about these activities, demonstrating their understanding and appreciation of the reflective activities:

\[\textit{p07e}:\] “they were really good to review what you thought of your own ability”.

\[\textit{p09e}:\] “I think they’re helpful to make you aware of what you said, you should do to answer the questions, and what you actually did do because it showed you the time that you took on each thing”.

The most helpful activity for them was the pre-task activity of Self-assessment of the problem comprehension and difficulty; it was followed by another pre-task reflective activity, the Comparison of knowledge monitoring and performance, in which they compared their performance with their predictions of success. A common feature in both activities is that the students had to interact with the activity, stating their views.

In the interview the students often made positive comments about the bar graphs in the Comparison of knowledge monitoring and performance activity:

\[\textit{p07e}:\] “I thought they [the bar graphs] were good to make you see if you can do what you think you can do”.

\[\textit{p08e}:\] “I think it [the bar graphs] just helped you see how realistic you were, and also, on the second question where I got it wrong and I was thinking that I understood what the givens and relations were and the graphs showed clearly that I didn’t. That’s probably quite a useful and simple way of showing where your problems are”.


On the other hand, the reflective activity least appreciated was the Analysis of knowledge monitoring state (KMA/KMB) in which there was little interaction. Even though, some students made positive remarks about it:

\[ p10e: \] “I realised I was underestimating myself from the graphs [...] so I found that useful and encouraging and that sort of relaxed me a little bit cause you’re being told that look, you can do it!”

The students evaluated the Selection of metacognitive strategies as a helpful activity, but it was the least used. It indicates that students acknowledged its helpfulness but did not feel the need to use it for all problems. Table 8.7 shows the results.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Helpfulness</th>
<th>Freq. Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self-assessment of problem comprehension and difficulty</td>
<td>2.83</td>
<td>3.58</td>
</tr>
<tr>
<td>Graphs with predicted vs. demonstrated performance</td>
<td>2.36</td>
<td>3.25</td>
</tr>
<tr>
<td>Selection of metacognitive strategies</td>
<td>2.33</td>
<td>2.50</td>
</tr>
<tr>
<td>Evaluation of problem solving experience</td>
<td>2.25</td>
<td>2.92</td>
</tr>
<tr>
<td>Self-assessment of strategies usage</td>
<td>2.18</td>
<td>2.50</td>
</tr>
<tr>
<td>Analysis of knowledge monitoring state (KMA/KMB)</td>
<td>1.91</td>
<td>2.91</td>
</tr>
</tbody>
</table>

Table 8.7: Students’ evaluation of the reflective activities: experimental group.

8.8.2 Evidence of Metacognition in the Interviews

The subjects took part in semi-structured interviews at the end of each session with MIRA. The participants of the experimental group made very frequent and interesting comments about the way the reflective activities helped them in their problem solving experience and interaction with MIRA. Typical comments made about various topics will be presented here.

**Awareness of self-as-a-learner**

Looking at the evolution through the sessions we noted that the discourse of the participants incorporated an increasing focus on the learning processes and an increasing awareness of the self-as-a-learner.

\[ p04e: \] “I learnt that although I got them [the problems] right, I spend hardly any time checking my answers properly. So perhaps if I’m working on a more difficult problem, then I need to spend longer checking the answers to be sure”.

\[ p12e: \] “I learnt how to reflect on what I’d done and try to learn from that for next time you do a problem”

\[ p05e: \] “I learnt that I’m not as bad at maths as I thought I was and that I need to slow down when I’m doing the tasks.”
Chapter 8. Experimental Study of MIRA

**Awareness of difficulties**
Participants in the experimental group reported some reflection on their difficulties in translating algebra problems.

*p10e*: “I can actually do calculations quite easily, it’s actually putting them together that I’m a bit dodgy at.”.

**Time management**
Many students reported that the reflective activities helped them become aware of their problem solving behaviours. They noted for example that they should slow down and not rush to attempt to write down the solutions to the problems.

*p03e*: “For me the temptation is to go quite quickly over the maths problems and in doing that I make mistakes but this system makes me go much slower, I really read the questions more and I broke it into sections about who was doing what and who the relations were between”.

*p09e*: “[I learnt] to take a little bit more time when I’m trying to work something out, to not rush too much”.

*p01e*: “I got a couple of them [problems] wrong or I didn’t give a full answer. I tried to go too fast, and it [MIRA] points out that you should go slower with your strategy and then I got the last question right”.

**Planning**
Several participants reported planning behaviour or awareness of the need for planning. An interesting example was:

*p06e*: “[it was useful to] identify what my strategies were going to be before I actually started doing it.”

**Monitoring**
It was expected that the reflective activities, specially the selection of metacognitive strategies, would have a positive influence on students’ monitoring behaviour or at least an increasing awareness of the need to monitor their problem solving attempts. We found some interesting reports of monitoring behaviour, such as ongoing checking of the solution steps, checking the complete solution and management of the remaining time to solve the problem.
Chapter 8. Experimental Study of MIRA

p05e: “[...] I actually needed to just sit back and go through each step slowly and then I’ll get the answers right.”

p10e: “although I got them [the problems] right, I don’t spend hardly any time checking my answers properly so perhaps if I’m working on a more difficult problem then, I need to spend longer checking the answers to be sure.”

p01e: ”It’s taking the time to really dissect what you’re looking at and getting used to the terminologies [...] and thinking hang on, what does that mean again?”

p01e: ”[during the session] I thought I needed to make better use of the system cause I was still rushing in and trying to do the problems regardless of all the aids that were there to help me”.

Awareness of knowledge monitoring (KMA/KMB)

It was interesting to note that some participants incorporated in their discourse the terminology used in MIRA for expressing the knowledge monitoring ability and biases. Moreover, some demonstrated that they felt more confident about their ability to solve maths problems as a result of the feedback MIRA provided about their KMA and KMB. One student said:

p11e: “I guess I learnt to be more optimistic about what I can do and that things aren’t as hard if you break them down and look at them logically”.

Evaluation of problem solving experience

We found some evidence that the post-task reflection activity and the timeline graph were useful ways to provoke reflection on the participants’ learning experience.

p05e: “that was interesting to see how much time you’d spent on things cause you don’t often sit back and think about that, because you so want to get onto the next question. It’s good to learn what you’d done”.

p04e: “The [timeline] graphs helped me to understand how you approach problems and maybe if you’ve made mistakes you can assess where you’ve rushed things [...] I know if I had got something wrong I probably would have looked at the graph and thought oh, maybe I didn’t spend enough time changing my answer or whatever”.

8.8.3 Use of Strategies

The post-task reflective activity included a task were students reported their use of the strategies selected in the pre-task reflective activity “Selection of Metacognitive Strategies”. The content of these free-text fields were logged and indicate whether or not students monitored the use of the selected strategies during problem solving. Some students were not consistent on writing their impressions after each problem. Others tried to write down their reflections for all problems. We included below one example of a typical participant that shows how students regarded the strategies in MIRA.
Chapter 8. Experimental Study of MIRA

After Problem 1 (p01e): “Separating the important parts is essential - I could not have solved the problem without this. Substituting numbers to see if the equation worked told me my answer was correct. I had to re-read the problem to ensure I was right, and check the task description to make sure I had done what was asked. All these strategies were fundamental to me getting an answer.”

After Problem 2 (p01e): “I got confused, and in my rush didn’t check the solution correctly, thinking it was right when it was not. I should have used my selected strategies more!”

After Problem 3 (p01e): “This one went much better than the two before it, as I spent longer on strategies (hence solving problem) and got it right!”

After Problem 5 (p01e): “I got lost in the problem as I could not quite understand how to express what wanted, unfortunately these strategies did not help me with this, but helped with trying to solve it.”

After Problem 6 (p01e): “My strategies worked ok, the mistake I made was not understanding fully how to solve such an equation (I got the basic premise of house painting incorrect!). I am spending more time in the problem solving stage using the strategies.”

Similarly, during the interviews students reported that both the ‘Selection of Metacognitive Strategies’ activity and the free-text reflection activity encouraged them to think of appropriate metacognitive strategies for solving the problems and motivated them to monitor the use of these metacognitive strategies.

p06e: “I found it really useful having that section where you could write how you thought your strategies had worked, so you could see what you’d said you were going to do and whether you actually did it and whether it worked or not.”

8.9 Discussion of Results

We tested the RA model proposed by implementing MIRA - a full ILE for algebra word problems. The experimental version of MIRA included an implementation of the RA while the control version did not. The empirical study conducted with 25 first-year undergraduate students showed that students who performed the reflective activities spent significantly more time on tasks and gave up on fewer problems. Moreover, this group answered significantly more problems correctly than the control group.

As both groups had the same amount of time to interact with MIRA, we predicted that the experimental group would solve fewer problems than the control group. However, we also predicted that the experimental group would perform better. Indeed, the number of problems attempted by the experimental group (N=112) was highly significantly smaller (Mann-Whitney U test, z=2.56, p=0.005 < α = 0.01) than that of the control group (N=136). At the same time, the experimental group had a significantly better performance in MIRA than the control group: the number of correct answers per total of problems attempted was significantly greater than that of the control group (z=1.66, p (1-tailed)=0.048). This was also the case for the number of answers almost correct (with minor errors) per total of problems attempted (z=1.82, p (1-tailed)=0.04).

Curiously these results did not replicate in the pre and post-tests. One possible explanation is that the post-test was devised to be harder than the pre-test (as explained in section 8.2.6), but it ended up being more difficult than intended. As a result most of the participants performed worse
in the post-test than in the pre-test and worse than during their interaction with the system. Even
the experimental group, which presented a positive difference in performance when interacting
with the MIRA environment, had a drop in performance from pre to post-test. We suppose that this
happened because the problems in the post-test were quite different from the problems proposed
in the system. So the participants could not apply the same reasoning for those problems. This
was a limitation of the experimental method.

At the metacognitive level, the descriptive values show that there was a higher increase of
KMA in the experimental group than in the control group. Unfortunately, this difference was not
statistically significant. However, the participants reported in the interviews that they benefited
from the reflective approach and showed in their discourse awareness of their metacognitive skills.
So, even though we have some evidence of the benefits on students’ knowledge monitoring skill,
we can not make strong claims about the validity of the RA model for knowledge monitoring
development.

8.9.1 Limitations of the Experiment

The experiment was quite complex to be executed. It involved system development as well as dif-
ferent activities, materials, and information analysed from different sources. Like other controlled
lab experiments that aim to provoke learning gains, it faced design difficulties that created some
constraints on the results which prevented finding precise conclusions.

We recognize that our experiment has some limitations concerning its design. One strong
limitation was the small number of participants in the experiment. It is normally very difficult to
gather volunteers for experiments, specially if it involves specific skills (like maths) and demands
that participants return for several sessions on different days. Also, ideally we would like to ask
subjects to return for a late post-test to study retention of learning gains. It was not possible to
achieve this under the circumstances we had.

There was also another problem in the experiment concerning the sample used. Despite the
randomized allocation of subjects to the groups we ended up with differing patterns of skill, moti-
vation, and age across the two groups. We should have restricted the participation to students of a
certain age category, and with no A Levels in maths.

The randomization method for allocating participants into proved to be not a good method,
since we ended up with a stronger group in the experimental condition in terms previous knowl-
edge of maths (more students with A Levels in maths). A better approach would be to use the
maths school grades to allocate participants into groups. Another alternative approach would have
been to use the pre-test results to organize the participants into groups according to the metacog-
nitive abilities demonstrated - this alternative would require a separate session only to perform the
pre-test.

An additional factor that reduced the power of the experiment was the limited interaction with
MIRA. We believe the metacognitive development needs some time to take place. The low amount
of time using MIRA most probably affected the results.

Another methodological limitation was the fact that we did not collect any data about the KMA
for the control group during the interaction with MIRA. Thus, comparisons of evolutions of the
KMA of the experimental and control group during the interaction with MIRA were not possible.
According to Tobias and Everson (2002) just asking students to predict their knowledge during a treatment already provoke changes to their knowledge monitoring accuracy. So, we decided not to ask participants in the control group to assess their knowledge of the problems so as not to interfere with the results. As a consequence we were restricted to observe differences between groups using the pre- and post-tests only.

A trouble only detected later on the experiment was the level of difficulty of the post-test, that seemed too difficult to all students. Had we noticed it earlier, it would be revised to a more compatible test.

Another limitation, not related to the experiment, but to the design of the MIRA system is that the types of problems presented in MIRA were not tailored to each student’s skills of algebra problem solving. As a consequence some students were not very much challenged during the experiment, whereas others were. This fact may have contributed to create an unbalanced motivation for reflection among the participants.

Thus, we believe that some of the results may have been influenced by these limitations.

8.10 Conclusion

The results of the experiment show that the RA model made an overall positive difference in the way students interacted with the problem solving environment.

At the domain level, we believe that the presence of the reflective activities was a factor that influenced the quality of students’ problem solving attitude in MIRA. Support for this claim comes from the fact that the experimental group performed significantly fewer problems compared to the control group and provided significantly more correct answers in MIRA. They presented a better problem solving time management and gave up on fewer problems, which indicates a higher motivation on difficult problems.

This shift from quantity to quality is an interesting consequence of the inclusion of the RA model in MIRA. As seen in this experiment, the quantity of problems performed did not lead to better performance.

Another experiment, with a bigger number of subjects and better control to address the limitations presented above, is needed in order to draw more definite conclusions about the influence and benefits of each of the reflective activities proposed here.

Although not all of our hypotheses were supported, the experiment enabled us to learn important lessons from which we can identify some weaknesses of the RA model and envisage possible redesigns of the reflective activities which would target those weaknesses. These are detailed in the next chapter.
Chapter 9

Conclusion: Lessons Learned

The importance of metacognitive models in the AIED agenda has been pointed out by several researchers. However automatic assessment of metacognition has been almost unexplored so far and computer-based learning environments capable of addressing both cognitive and metacognitive aspects of learning are scarce. The work reported in this thesis is a step towards such learning environments for problem solving that includes a metacognitive model.

The following sections outline the main contributions of our research, the limitations, and further directions for research.

9.1 Contributions of this Research

This thesis has:

- Formalyzed, implemented and evaluated the Reflection Assistant model, which is a model of metacognition training to be incorporated into problem solving learning environments. It proposes:
  - a new instructional design for metacognition instruction in problem solving environments, including reflective activities to be performed before and after problem solving attempts;
  - novel graphical reifications of the metacognitive skills demonstrated or inferred.
  - an automatic assessment of the skill of knowledge monitoring. This assessment uses the student’s own assessment of her understanding of the problem and compares it with her performance. We have created two measures of knowledge monitoring: knowledge monitoring accuracy (KMA) and knowledge monitoring bias (KMB). These values are used by the reflective activities to promote awareness of the student’s own state of knowledge monitoring and to trigger changes and improvements.

- Designed, implemented and evaluated a fully working system - the MIRA learning environment for translation of algebra word problems. The system implements the RA model and includes problem solving resources. Among the resources we have designed and implemented is the PAL tool, which helps students to understand and represent the components of the problem.
Our main research question was: Does the Reflection Assistant model help students to improve their metacognition? And, if so, does it lead to improvements in problem solving performance?

We expected that the interaction with the reflective activities implemented in MIRA would encourage students to become more conscious about their learning processes and metacognitive skills.

Our research has made contributions in different aspects of metacognition training in ILEs as discussed in the next sections.

9.1.1 The creation of the Reflection Assistant model for metacognition training

This thesis builds on the idea that it is necessary to create separate moments where self-reflection should be triggered and scaffolded. The Reflection Assistant model defines reflective activities that should take place before and after the main domain-related activity. Two new stages in problem solving activity are proposed within the model: a pre-task reflection stage and a post-task reflection stage. During these stages the student performs solely reflective activities, which focus on her metacognitive skills and on her problem solving experience. We have defined the RA model as a separate module from the problem solving environment. This way, it can be adjusted and incorporated into existing ILEs.

The RA model specifies instructional approaches for training the following metacognitive skills: (i) the ability to accurately determine if one has understood a problem and is able to identify its important parts, (ii) the selection of suitable general strategies to be applied during problem solving to overcome difficulties, and (iii) the evaluation of one’s own use of resources, time, and strategies.

The experiment conducted showed that at the domain level the RA model had a positive effect on students’ problem solving performance. The experimental group, who used it, answered significantly more problems correctly than the control group. Moreover, it seems that the RA model motivated students to persist on difficult problems, as they gave up on fewer problems than the control group.

Some evidence of a positive effect of the RA model on students’ metacognition was observed:

1. The RA model helped students monitor their problem solving time better; it discouraged rushing to attempt to write down a solution to the problems (as reported by many students in the interviews) as well as ending their time without an answer to the problem (as shown by the statistical test as marginally significant).

2. Both the content of the free-text with reflections about the use of strategies in MIRA and the interviews showed that the students were encouraged to think of appropriate metacognitive strategies for problem solving and some of them felt motivated to monitor the use of these metacognitive strategies.

3. The subjects’ discourse in the interviews incorporated an increasing focus on their learning process and an increasing awareness of the self-as-a-learner.

*The creation of an automatic assessment of metacognition*

Another contribution of this thesis was the creation of an automatic assessment mechanism for knowledge monitoring. We have adapted an instrument from Tobias & Everson (2002), which determined the level of accuracy of students’ knowledge monitoring (KMA). We have generalized
Chapter 9. Conclusion: Lessons Learned

157

their definition of KMA to include an intermediate level of accuracy. Furthermore, we have added a new measure, called knowledge monitoring bias (KMB), which provides additional information about any tendency the student might have in her knowledge monitoring skill. With it, we are able to categorize students as pessimistic, optimistic, realistic, or random with respect to their knowledge monitoring awareness. It complements nicely the KMA in giving a picture of students’ knowledge monitoring status.

The statistical tests performed showed that our adapted KMA measured during the interaction with MIRA correlated positively with students’ performance in MIRA (see section 8.7). Thus, our assessment mechanism keeps the original correlation observed by Tobias & Everson.

The Design of new graphical components for self-reflection

We have created novel graphical reifications designed to convey to the learner information about her metacognition:

- bar-graphs were created to help students reflect on the tendencies in their judgement of performance and compare them with their actual performances (Figure 6.4 on page 102). Those graphs were supported by a complementary activity: students had to fill the blanks in a text that rephrased what was visible on the graphs. Students evaluated this activity as the second most useful among the reflective activities proposed.

- reflectometers were designed to trigger reflection on KMA and KMB (Figure 6.5 on page 103). Although they conveyed a new notion in an easy and quick way, the students evaluated this component as the least helpful. However, a few students made positive comments about their usefulness.

- a timeline graph was created to show the amount of time the student spent on each task (Figure 6.15 on page 110). Next to the timeline graph a feedback message was displayed combining information related to effort, use of the PAL Tool, and time spent on checking the teacher’s solution. Students evaluated the timeline graph as a helpful resource, although some students reported that they did not know what to do with the information provided.

9.1.2 The development of an ILE dedicated to algebra problem translation with a reflective approach

Previous research argued that the ability to take a problem situation and formulate the equations (i.e. translation) is the most important skill needed to succeed in algebra problem solving and that it is at the same time the major source of misconceptions and errors.

Our problem solving environment focused exclusively on the translation process, offering the numerical answer to the problem as an aid resource to help check if the equations were correct. This approach was seen as a fresh one by many students and showed potential to motivate them to think about algebra word problems in a novel way.

The design and development of the PAL Tool

An important by-product of our research was the PAL Tool. This tool was based on a heuristic we developed which in turn was inspired by Polya (1945), Mayer (1999), and Whimbey and Lochhead (1999).

The PAL Tool encourages students to structure their thinking about algebra word problems in terms of four categories of structural elements: the givens, goals, unknowns, and relations.
All participants of our experiment had access to the PAL Tool. In the evaluation performed the participants consistently ranked the PAL Tool as helpful. One participant commented: “[I learnt] how to break down a question so that you can form an equation which I’ve never had anyone explain that to me before. You know they basically said: ‘there’s the question, write the equation’. I was like ‘where do I start?’ (p10e)”.

However, as we said in Section 5.8.3, the PAL Tool has limited power to help in the translation process. A future version of the tool should incorporate new features to help those students with greater difficulties in problem understanding. Following a self-reflective approach, we think that the addition of some visual feedback on how the student is using the tool would make this component even more helpful.

Further research focussing exclusively on the effectiveness of the PAL Tool should be undertaken. Moreover, it would be interesting to compare it with other tools intended to help problem comprehension and translation into equations, such as TAPS (Derry and Hawkes, 1993) and HERON (Reusser, 1993).

**The definition of a classification scheme for guiding the design of metacognition in ILEs**

We have created a classification scheme for metacognition instruction that maps out the design space along three dimensions: timing, target level, and instructional approach. It proposes possible values for each dimension (as presented in Chapter 3). This classification can be used as a thinking aid for designers who wish to create a metacognition training module into ILEs.

**9.2 Limitations of the research**

The RA model is one of the first attempts to create a tool for explicit scaffolding of metacognition instruction in problem solving environments. It has several limitations, not least because of the lack of an operational and tractable model of metacognition training which could guide the instructional decisions. The main limitations are discussed below.

**Oversimplified metacognitive student model**

The student metacognitive model proposed in the RA is limited as it only contains information about the knowledge monitoring state of the learner. Other aspects of metacognition had to be overlooked, such as ability to plan the strategies, monitoring, evaluation of learning, etc. Another shortcoming is the fact that it does not take any notice of the motivational state of the learner. In reflection as in other aspects of learning, motivation is an important factor. In order for reflection to take place and be effective, there has to be a minimum motivation to do so.

**Some of the reflective activities are not engaging enough**

Some activities, such as the Reflection on KMA/KMB and the Evaluation of the Learning Experience are not as dynamic and interactive as we would have liked them to be. They present innovative graphical reification and tailored feedback messages but they lack interactivity. We believe that interactivity would contribute to the desired effect of these activities which is to trigger as much reflection as possible.

It is interesting to notice that students reported that the Teacher’s Solution was a very helpful resource, despite its lack of interactivity (it is simply displayed for the student to read). We believe
that this has to do with the relevance that students perceive in an activity. In the case of the solution, it is easier for student to see the relevance and therefore engage with the activity despite its lack of interactivity. It is also perceived as bringing closure to the problem. The reflective activities, on the other hand, require dealing with issues which are not so obviously related to the effort the student has put in solving an specific problem. Furthermore, the student doesn’t get away from them with the same feeling of closure which she gets after seeing the solution. On the contrary, the reflective activities, as their name indicates, are more designed to prolong the thinking.

**The RA model is not flexible**

The fact that the reflective activities appeared to students regardless of their performance or metacognitive state, diminished their power in some circumstances. The danger of unadjusted metacognitive scaffolds is that they can frustrate the students as they create additional cognitive load; this can result in the student ignoring the tasks entirely. This could be observed in cases where the problem was easy for the student or when the student was very confident about solving it. We believe that the RA model would be more effective if it matched the quantity and type of reflection to the student’s need.

**Limitations of the experiment performed**

Our experiment also had limited power. Although we were able to detect some important consequences of using the RA model, we were not able to make a conclusive assessment of its impact on students’ metacognition development. The small number of participants and the limited amount of interaction with the MIRA system were some of those limitations (as seen in section 8.9).

### 9.3 Future research directions

The limitations of the RA model combined with the results and insights from the experimental study performed, point in several directions that could be explored in order to take this research further.

#### 9.3.1 Improvements in the RA Model

**Refinement of the metacognitive student model**

Additional information relevant to metacognition could be taken into account, such as the strategies selected, the time spent on each activity for different problems (e.g: time spent on easy problems vs. difficult problems), etc. However some mechanism for transforming this raw data into information about metacognition also needs to be devised. Adding information about the student’s motivation would also open new possibilities.

**Including support for collaboration**

Because we intended to observe and measure individual metacognitive states, we did not include collaborative tasks in the RA model. But the reflective activity *Evaluation of the learning experience*, for example, could include comparisons with more experienced learners. Additionally, having students working in pairs with MIRA could result in deeper reflection and greater observable effects of the Reflection Assistant model.
Making the RA model more adaptive

Selective appearance of the reflective activities depending on the student’s metacognitive state could be introduced. Improving the student model would make this even more interesting as a more detailed picture of the student’s state would mean a richer set of possible conditions to play with. A good set of rules matching possible metacognitive states with possible scaffolding strategies would then be needed. For example, the activity Selection of metacognitive strategies is useful when the problem is difficult, when the student’s KMA is low, or when she is uncertain she can solve the problem. This activity would thus be offered only in those situations instead of always as is currently the case.

9.3.2 Improvements in the MIRA Learning Environment

The MIRA system can be improved in many aspects. Some possible directions are:

- A major change could be the redesign of the system, creating separated modules: the interface objects, the domain information, and the metacognitive module. With this MIRA would become a shell for different problem solving domains (physics, for example). Different experiments would be possible, comparing the usefulness of the reflective activities for those domains.

- Another improvement needed is the revision of the problem solving resources offered in MIRA.

  - The Glossary of Relations and the Algebra Text should be more dynamic and interesting. Adding some activities related to these texts can enhance its use. For example, for novice users or algebra problem solvers with difficulties, the first set of activities could be related to gaining basic information about algebra problems and how to translate problems. Tasks that involved the study of these materials should be devised. This may motivate students to use these resources later on whenever they need conceptual help.

  - The experiment conducted demonstrated that the students did not use the library of problems as a source of help, even for similar difficult problems. A mechanism that helps student to select similar problems could be devised. We have made a prototype of such mechanism, but we did not include it in our current version of MIRA because more in-depth research has to be done on the ways to describe similarities between problems. The database of problems has to include a description of the features of the problems and a matching algorithm has to be devised.

- The translation activity could be further scaffolded. Ms. Lindquist (Heffernan, 2001) is an ITS that focuses essentially on the translation stage of algebra word problems and it has investigated interesting strategies to help students translate algebra word problems. Strategies such as (i) asking the students to first solve a concrete instance before asking them to use the variable and (ii) present the student with the answer and then ask her to explain the portions of the answer. A help resource could be incorporated in a future version of MIRA, offering scaffolding using similar strategies to help the weakest students.

9.4 Final Remarks

This research has shown the feasibility and the benefits of computer-based environments that balance metacognitive and cognitive tasks. This is a vast area of enquiry and more research on how
to combine these two learning targets (problem solving and metacognitive development) has to be conducted, taking into account the many aspects still unexplored.

The inclusion of a social component, through collaborative activities, and a more flexible model of metacognition training are promising directions to make such environments more meaningful and effective to students.

We believe that this research has opened up many interesting paths trails and we hope it can motivate people to take it further.
Bibliography


Bibliography


Bibliography


Appendix A

Algebra Word Problems

This appendix contains a list of the 14 algebra word problems used in MIRA as well as details about the coding used to classify the problems. The intent of the classification was twofold:

1. to help to select appropriate problems and organize their order of appearance in MIRA;
2. to guide the interpretation of students’ attempts to solve the problems.

The problems were created or adapted from maths books or websites and they had to fit the following requirements for the intended experiment with MIRA:

- **Present meaningful situations:** Because the main experiment target primarily undergraduate students the description of the problems were adjusted to describe situations that matched students’ reality. So, when a problem originally talked of candies or toys, we have modified the situation to describe trips to and from work, startup companies, dinner preparation for friends, etc.

- **Avoid automatization of problem solving actions:** It was important to present novel situations, where the student needed to think about new strategies or use new resources to help on solving the problem. On the one hand we needed an environment where the student reflected on the problem, but on the other hand providing some similarities between the problems to motivate the student to use past experiences as a source of help. Having a limited set of different types of problems created the desired balance between not always providing a completely new problem and not offering a repeated pattern. We have used Mayer (1999)’s classification for types of maths word problems.

- **Offer a varied level of problem solving difficulty:** As we did not select participants taking into consideration their previous knowledge of algebra problem solving and we did not implement the MIRA environment with the capability of adjusting itself to the student’s knowledge state, we needed to offer problems ranging between low, average ad high difficulties. Actually we opted to have the same problems shown to all students in the same order to facilitate to comparison of the control group with the experimental group.

- **Demand an average to high attention for attaining problem comprehension:** Because we were investigating the metacognitive skill of knowledge monitoring, it was desired that in order to fully understand the problem and the concepts it involved, the participants had to read it carefully and attentively. So, we have elaborated problems which contained non straightforward components, like hidden relations, implicit goals, etc.
A.1 Classification

To fulfill the requirements presented above, 25 problems were initially classified into the dimensions presented below and, from these, 14 were selected to be included in MIRA.

A.1.1 Types of problems

We have selected 4 types from Mayer (1999)’s classification, and we have created problems from to one of those types. A description of the problem types are given below.

A - Age Relations Problems These are problems that describe relations between different people’s ages (in the present, past or future). They are single-variable algebra word problems up to three variables. These problems are usually solved by: (1) translating the relationships presented into linear equations, and (2) applying simplification methods in the equations and arithmetic operations to find the variables values;

B - Ratio-Proportion and Relations Problems These are problems that describe either ratio-proportion between different objects (concrete or abstract) or relations between objects. For those that present ratio-proportion, students may solve by: (1) translating the relations into fractions (or ratios), (2) creating proportions, i.e. equations that say two ratios are equal, and (3) applying the cross-multiply and divide method. For the problems that present relations the solution involves: (1) translating the relationships presented into linear equations, and (2) applying simplification methods in the equations and arithmetic operations to find the variables values;

C - Money Relations Problems These are problems that describe ratio-proportion between prices of products and other relations involving money. One important characteristic is that all problems of this type present real-life situations, which students might have experienced before. This creates the potential for students’ use of invented procedures and strategies to solve the problems. They are two-variable problems up to four variables. The problems can be solved following similar procedures to types A and B;

D - Velocity-Rate, Distance and Time Problems These problems require basic physics knowledge of kinematics involving distance, speed and time. All problems present situations where the speed is constant. The problems are usually solved by: (1) identifying information for distance, rate of speed and time, and (2) applying the distance equation: Distance = Rate-of-Speed * Time.

A.1.2 Difficulty level

This classification focuses on the mathematical challenges represented by the problem, including the difficulty of the mathematical concept, the type of strategies or heuristics to be applied, the number of steps included (which may cause the student conceptual difficulties), and the difficulty of mathematical operations to be done to check the equations. It is important to notice that problem difficulty is mainly defined relative to other problems from the same type presented in the list. A list of the features of the each problem was elaborated in order to facilitate the classification. It was used as basic information to be taken into account when classifying the problem into a certain level of difficulty. The rating scale consists of 4 levels of difficulty, wherein a Level 4 problem is a very difficult problem. It is agreed on problem solving literature that ultimately, the difficulty of a problem is relative to the individual- student Schoenfeld (1989), but for effect of
Appendix A. Algebra Word Problems

basic classification, it is considered here an average student that has just finished high school and has not done A levels in Maths.

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Minimal difficulty</td>
</tr>
<tr>
<td>2</td>
<td>Moderate difficulty</td>
</tr>
<tr>
<td>3</td>
<td>Difficult</td>
</tr>
<tr>
<td>4</td>
<td>Very difficult</td>
</tr>
</tbody>
</table>

A.1.3 Comprehension Level

Because the MIRA environment focuses on students’ abilities to understand and translate the problem into mathematical equations, a separate rating is necessary to evaluate each problem in terms of the level of abstraction and the synthesis required to fully understand and build an initial representation of it. The difficulty in understanding is defined relative to other problems from the same type. The levels of difficulty in comprehension are presented in the table below:

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Requires moderate translation and integration skills</td>
</tr>
<tr>
<td>2</td>
<td>Requires good translation and integration skills</td>
</tr>
<tr>
<td>3</td>
<td>Requires good translation, synthesis, and abstraction skills</td>
</tr>
<tr>
<td>4</td>
<td>Requires very good translation, synthesis and abstraction skills</td>
</tr>
</tbody>
</table>

A.2 Problems presented in MIRA

A maths teacher (and Head of Maths) of a comprehensive secondary school in Brighton volunteered to score the problems, using the classification explained above. He also reviewed the description of the problems and helped the researcher to select 14 problems from the original list
of 25 problems. The final list of problems is displayed in Table A.2.

The list shows the problems in their order of appearance in MIRA. The column “Prob. ID” has the number given to the problem (used to identify the problems in the log files). The column “Feature” presents a summary of relevant algebraic features that helped to classify the problem. The column “Diff.” presents the value for Difficulty Level and “Comp.”, the value for Comprehension Level.
Table A.1: From an original list of 25 word problems created or adapted, the table below shows 14 problems that were selected to be included in MIRA along with the scores given by a maths teacher. The problems are presented in the order they are shown in the MIRA environment.

<table>
<thead>
<tr>
<th>Prob-ID</th>
<th>Problem Description</th>
<th>Main Features</th>
<th>Type</th>
<th>Diff.</th>
<th>Comp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Paul is forty years old. He has two sons, named Mat and Kevin, and one daughter, called Emily. Mat is one quarter of Paul’s age. Kevin is two years younger than Mat. How old is Kevin?</td>
<td>• 3 variables&lt;br&gt;• 2 relations&lt;br&gt;• no hidden variable or relation&lt;br&gt;• simple subtraction and division operations&lt;br&gt;• irrelevant information included</td>
<td>A</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Dave, Fred and Bill are university friends. They have saved some money to start a business together after graduating. Graduation day arrives and it is time to see how much each one has saved to invest in their start-up company. Dave has saved six times as much money as Fred, and Bill has saved three times as much money as Fred. Together they have £10,550.00. How much does each friend have?</td>
<td>• 4 variables&lt;br&gt;• 3 relations&lt;br&gt;• one partially hidden relation (total is the sum of each person)&lt;br&gt;• simple multiplication and addition operations&lt;br&gt;• big numbers - needs calculator</td>
<td>C</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>In April, Jill planted 40 roses in her yard. Then in May, she decides she needed more flowers and she planted some tulips and some poppies. She planted twice as many tulips as she had planted roses and one forth as many poppies as tulips. How many flowers she planted all together?</td>
<td>• 4 variables&lt;br&gt;• 3 relations&lt;br&gt;• One partially hidden relation (total flowers is the sum of each type of flower)&lt;br&gt;• Simple multiplication and division operations&lt;br&gt;• word order in relations can cause reversed equations in the translation</td>
<td>B</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Prob-ID</td>
<td>Problem Description</td>
<td>Main Features</td>
<td>Type</td>
<td>Diff.</td>
<td>Comp.</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
</tbody>
</table>
| 4       | Ben and Kevin are cousins. Kevin has moved to another city and he hasn’t seen Ben for 2 years. One day they meet in a family wedding party. As they are chatting, they discover a very interesting fact. Another guest that is over hearing the conversation gets the following: five years ago Ben was 2/3 as old as Kevin was then. Ten years from now he will be 5/6 as old as Kevin will be. Then, the guest wonders: How old are they now? | - from 2 up to 6 variables  
- 2 explicit relations  
- 2 hidden relations (Ben’s and Kevin’s ages in the past and future relate to their age today)  
- operations with fractions  
- time element can cause missing parts in the translation  
- irrelevant information given | A    | 3    | 2    |
| 5       | Steve is 12 years old. He saved £65 from his pocket money to buy a skateboard. Steve goes to the Thunder Mountain Sports store to check for the prices. He sees that a GoldenSnake skateboard costs £12 more than the Silver Streak skateboard, but £5 less than the Wizard skateboard. The three skateboards together cost £164.00. Which skateboards could he not buy? | - 4 variables  
- 3 relations  
- simple addition and subtraction operations  
- irrelevant information included  
- goal comprehension needs attention | C    | 3    | 4    |
| 6       | Mr. Johnson realizes that 2 years have passed since he last painted his house. However, he is very busy at the moment and he remembers that last time it took him 15 hours to paint it on his own. So he calls his friend Bob, an experienced carpenter, to help him with the job. Bob tells him that it normally takes him 5 hours to paint a house half the size of Johnson’s. How long will it take to paint the house if both of them work together? | - up to 5 variables  
- about 5 relations (depending on the strategy)  
- hidden relations  
- addition and multiplication operations  
- knowledge of proportion needed  
- irrelevant information included  
- unit(proportion of house) may cause confusion | B    | 4    | 4    |
| 7       | Frank and Karen are siblings in a family of five children. Frank is eight years older than his sister. In three years he will be twice as old as she will be then. How old are they now?                                                                                                                                                                                                                         | - 4 variables  
- 2 explicit relations  
- 2 hidden relations (Frank’s and Karen’s ages in the future relate to their age today)  
- time element can cause confusion in the translation  
- simple addition and multiplication operations | A    | 2    | 2    |
<table>
<thead>
<tr>
<th>Prob-ID</th>
<th>Problem Description</th>
<th>Main Features</th>
<th>Type</th>
<th>Diff.</th>
<th>Comp.</th>
</tr>
</thead>
</table>
| 8       | Karin travels to work on his car without stopping from Brighton to Winchester at an average speed of 40 mph (miles per hour). On the return trip, he follows the same route without stopping at an average speed of 60 miles per hour. If Karin spends a total of 2 hours on the entire two-way trip, what is the distance, in miles, between Brighton and Winchester? | • 6 variables  
• 3 relations  
• simple addition and multiplication operations  
• knowledge of formula for distance/velocity rate/time needed                                                                                                    | D    | 4    | 2     |
| 9       | Nathalie’s maths teacher asked the students to find numbers with interesting characteristics. Nathalie started to look around. She took note of her house number, her age and her friend’s age. After some maths manipulations she found out that one of those numbers had the following characteristic: the units digit is 1 more than 3 times the tens digit. The number represented when the digits are interchanged is 8 times the sum of the digits. What is the number? | • 3 variables  
• 3 explicit relations  
• 1 implicit relation (a number with two digits is the tens digit multiplied by ten plus the units digit)  
• simple addition operations  
• abstract situation may cause confusion                                                                                           | B    | 4    | 3     |
| 10      | In January of the year 2000, Jamilla’s husband Changez was eleven times as old as their daughter Aruna. In January of 2012, he will be three times as old as their daughter. How old was Aruna in January of 2000? | • 4 variables  
• 2 explicit relations  
• 2 implicit relations (ages in the future and ages at the present)  
• simple addition and multiplication operations                                                                                    | A    | 2    | 2     |
| 11      | You are returning from a trip to Montreal where you visited some friends. It took 7 hours to fly from London to Montreal, but you notice in your ticket that it will take 8 hours to return. You are seated next to a 7 years-old boy that is travelling alone. You start a conversation with the boy who is very curious and loves airplanes. He asks you what is the average speed of the airplane (consider it constant). You missed this information at the airplane video screen, but you remember to read that the wind velocity was in average 50 miles per hour. | • 6 variables  
• 2 explicit relations  
• 2 implicit relations (wind velocity relates to final velocity and place velocity)  
• simple addition and multiplication operations  
• knowledge of formula for distance/velocity rate/time needed  
• requires to make assumptions  
• irrelevant information included  
• goal presented in a non-directive way                                                                                       | D    | 4    | 4     |
<table>
<thead>
<tr>
<th>Prob-ID</th>
<th>Problem Description</th>
<th>Main Features</th>
<th>Type</th>
<th>Diff.</th>
<th>Comp.</th>
</tr>
</thead>
</table>
| 12      | You decide to have a nice bath. The hot water tap opened alone fills the bath tub in 30 minutes. The cold water tap alone takes 10 minutes to fill half of the bath tub. If you open both taps at the same time how long will it take to fill the bath tub? | - up to 7 variables (depending on strategy used)  
- 1 explicit relation  
- 4 implicit relations  
- division and multiplication operations;  
- knowledge of proportion needed | B    | 4    | 3    |
| 13      | You’ve got a job in London, but you will stay in Brighton and commute to work. You have £3,600 of savings and think that buying a car is a good solution. You don’t want to spend more than a third of your savings with the car tough. You talk to a friend that has a garage with used cars. He has a Mini/97, a Fiesta/98 and a Nissan Micra/99. The Fiesta costs £300 more than the Mini, but £150 less than the Nissan. The Nissan costs £300.00 more than what you want to spend. Which cars can you afford? | - 5 variables  
- 4 explicit relations  
- division, multiplication, and addition operations | C    | 3    | 3    |
| 14      | Maria and Juan are preparing a dinner for four friends. They are preparing a Spanish treat with tortillas and paella. They are following a recipe book and it says that a paella should take three times as much water as rice to cook. In the rice pack it is written you should allow 50g of rice per person. They have a measure cup for the rice and water that takes 100g of rice. How much water will they need to cook the paella? | - up to 6 variables  
- 1 explicit relation  
- 3 implicit relations  
- division, multiplication, and addition operations  
- knowledge of proportion needed  
- word order in explicit relation can cause a reversed equation in the translation | B    | 3    | 3    |
Appendix B

MIRA: Technical Documentation

MIRA is a program written in Java (j2sdk1.4.0). It was built using the programming environment Borland JBuilder Personal™. MIRA has 72 JAVA classes, where a few of them (mainly the ones that implements the graphs) were reused or extended from free-distributed libraries.

A section with a class relation diagram (UML diagram) of the main classes of MIRA and a brief description of them is presented at the end of this appendix.

MIRA runs under either Linux/Unix or Windows XP/Windows 98 Platforms on standard PC computers with a minimum of 128 megabytes (MB) of memory and a 1024x768 screen resolution.

MIRA requires user registration. Details about the registration procedure is given below. It generates separate data and log files for each user. The log and data files, as well as the directory files structure are presented below.

B.1 User Registration

First time users of MIRA has to register. Figure B.1 shows the registration window. In the registration the user provides some personal information (name, age, and sex), a username and the pre-test scores for KMA and KMB.

If MIRA is used in a situation where a pre-test is not used, the user should enter 0 in all pre-test scores fields.

At the end of the registration three new log files are created. They keep information about the user for different purposes. The content and purpose of these files are detailed in the next section.

Creating a username

The username has to comply with the following rule: \texttt{p\underline{dd}e}, where \texttt{dd} is a number of two digits, between 01 and 99 (ex: p03e).

At the end of the registration a random number (between 01 and 99) is generated automatically (ex:15) and added to the username. A message is displayed to the user showing the complete username (ex: p03e15). Next time the user has to type this username to log in.

This procedure was created to avoid the problem of one student accessing the data of another student by mistake, without even noticing (e.g. mistyping the number \texttt{dd} that was allocated to another student).
B.2 Log Files

There are three files that keep information about the user’s interaction with MIRA: <username>.obj, <username>.log and <username>.xml.

<username>.obj

This file is created using the object serialization\(^1\) implemented in Java with the Java Serialization API. In brief, it is a way to “save” all JAVA objects generated in memory during one session. At the beginning of a new session, MIRA looks for this file and reloads all objects belonging to the user. So, for example, imagine that a student completes three problems in the first session and the session finishes just after she is done with the Quiz of the third problem. When she logs in the next time, MIRA will recover all previous objects left in memory, such as all problems done and will restart from the next activity just after the Quiz of the third problem, that is “Check Teacher’s Solution”.

<username>.log

This is a text file, designed to record some of the users’ actions. The information in each line is organized as:

\[<\text{time}> \quad \text{-- -- --} \quad <\text{action}>\]

where <time> is recorded in milliseconds. The <action> recorded are all the main actions the student could do in MIRA, such as “start Problem Solving”, “selected strategy”, “start Check

\(^{1}\text{Object serialization is the process of saving an object’s state to a sequence of bytes, as well as the process of rebuilding those bytes into a live object at some future time. The Java Serialization API provides a standard mechanism for developers to handle object serialization. (Java Sun Technical Report - http://developer.java.sun.com/developer/technicalArticles/Programming/serialization/retrieved on 25/02/03)}\)
Appendix B. MIRA: Technical Documentation

Teacher’s Solution”, “see help”, etc. We have used the free-distributed, open-source package LOG4J, which is a tool for testing and creating logging in JAVA applications. It offers facilities to insert logging messages in the source code and to control the logging behaviour by editing a configuration file. A sample of a log file is found at the end of this Appendix.

We have devised another program to extract the time length of each action using the information from this file.

Another log file in the format of an XML (eXtensible Markup Language) document was designed to keep a record of a different type of information: the information generated by the student, such as free-text reflections, answers to problems, KMA and KMB values after each problem, etc. So, this file maintains all information that is not time-related.

We used the facilities provided in the CASTOR system to produce this log file. CASTOR is an open source data binding framework for Java. It provides appropriate link between Java objects and XML documents.

A general mapping (or template) XML file was designed with the specification of all information that has to be registered in this log file. The mapping.xml file is described in page 184 of this Appendix, together with a sample of an XML file generated by a student’s interaction.

The XML format brings advantages, facilitating the analysis of student interaction. Another program was built to extract the information from the XML file and input them into the SPSS statistical package.

B.3 Directory Structure and Data Files

Figure B.2 shows the directories structure of MIRA. The directory images has all images and icons used. The directory mira has all Java compiled classes, and lib the libraries used by the application (castor, graphiti, jdom, log4j, and multiline).

The information related to problems and other texts used are in the directories problems and textFiles respectively. These files are either in HTML and XML format.

The HTML format makes it easier to plug the information straight into MIRA with no need of further formatting (JAVA has support to open HTML inside the interface components). The XML format was equally chosen because it was more practical to use the CASTOR system to import data than to set up a database management system to run with a Java application. Also, these formats make the task of updating the content much easier.

The problems’ directory has one subdirectory for each problem. The subdirectories are named as 0, 1, 2, etc. The problems seen in MIRA follows a pre-defined order. First, the users see problem 1 (that is defined in the directory 1), then problem 2, and so forth. Each problem has three files that describe it:

- text.html contains the textual description of the problem, formatted as it will appear in MIRA.

\footnote{More information about the LOG4J project at http://jakarta.apache.org/log4j/docs/index.html (Last accessed on 26/02/03).}

\footnote{More information about the CASTOR Project at http://castor.exolab.org (Last accessed on 26/02/03).}
Appendix B. MIRA: Technical Documentation

Figure B.2: Structure of directories of MIRA

- `solution.html` contains the teacher’s detailed solution to the problem, formatted as it will appear in MIRA. All solution texts for all problems follow the same structure.

- `answers.txt` contains the possible multiple choice answers to a problem. Each line of this file contain one option. The correct answer has an "*" at the beginning and another one at the end of the line.

- `quizElements.xml` contains the questions that appear in the PAL Tool “Hints” and in the Quiz (shown only if the student fail to solve the problem correctly). This xml file also describes the goals, givens and unknowns of the problem in terms of their attributes. The attributes are: description, value, variable name, unit, keywords and synonyms. This information is used to keep track of students’ use of the PAL Tool and to create some textual messages in MIRA.

The `textFiles` directory contains introductory texts, comprising a text about the goals of MIRA, a text with an example of an algebra word problem and a definition of the terms “givens, goals, unknowns, relations, and equations”. The remaining of the text information in MIRA is organized in the following subdirectories:

- `algebraBasics` keeps the content in HTML files of the “Algebra Basics” resource.
• **glossary** keeps the texts in HTML format containing the content of the “Glossary of Relations”.

• **tutorial** contains the texts that appear when the student uses the help facility in MIRA. There is an HTML file that explains each screen in MIRA. The content of the help files follow a standard structure, always divided into two parts, entitled: “What are the goals of this Activity?” and “What shall I do?”.

Finally, the **doc** directory contains a technical documentation of MIRA (JAVA classes, methods, etc.) in HTML files. The documentation is also available in the Internet in the URL http://www.cogs.susx.ac.uk/users/claudiag/MIRA/doc/index.html.

**B.4 Sample of Log Files**

*Sample of an .log file*

Below we have included an extract of a subject’s .log file. It refers to subject’s interaction with MIRA in session 3.

```
3357574 --- --- SESSION 2 ENDS HERE ---

70281 --- Opening Overview
70291 --- Opening Pre Reflection- Said x Demonstrated
102157 --- Closing Pre Reflection- Said x Demonstrated
102167 --- Opening Feedback Gauge
113734 --- Closing Feedback Gauge
113744 --- Opening Pre Assessment
167541 --- Closing Pre Assessment
167551 --- Opening Planning Strategies
201039 --- strategy planning: show warning activity incomplete
201740 --- view strategy planning list: Strategies for Controlling Errors
217543 --- strategy planning: show warning activity incomplete
218134 --- view strategy planning list: Strategies for Revising
226826 --- Closing Planning Strategies
226857 --- Opening Problem Solving
226887 --- Solving Problem nb: 7
239465 --- Timer: view time left
283498 --- Opening Answer Viewer
298159 --- Closing Answer Viewer
420235 --- Closing Problem Solving
420285 --- Opening Answering Problem
504336 --- Closing Answering Problem
504346 --- Opening Checking Solution
515081 --- Closing Checking Solution
515111 --- Opening Post Reflection

593924 --- ---- SESSION 3 ENDS HERE ---
```
The mapping XML file

Here is an extract of the the XML template coded to create subjects' .xml files.

```xml
- <mapping>
  - <class name="mira.Student" access="shared">
    <map-to xml="Student" />
  - <field name="firstName" type="java.lang.String" required="false" direct="false" lazy="false">
      <bind-xml name="FirstName" node="element" />
    </field>
  - <field name="lastName" type="java.lang.String" required="false" direct="false" lazy="false">
      <bind-xml name="LastName" node="element" />
    </field>
  - <field name="age" type="java.lang.Integer" required="false" direct="false" lazy="false">
      <bind-xml name="Age" node="element" />
    </field>
  - <field name="female" type="java.lang.Boolean" direct="true" required="false" lazy="false">
      <bind-xml name="Female" node="element" />
    </field>
  - <field name="sessionNb" type="java.lang.Integer" direct="true" required="false" lazy="false">
      <bind-xml name="sessionNb" node="element" />
    </field>
  - <field name="sessionTimings" type="mira.TimeInterval" collection="vector" direct="true" required="false" lazy="false">
      <bind-xml name="sessionTimings" node="element" />
    </field>
  - <field name="problemNb" type="java.lang.Integer" direct="true" required="false" lazy="false">
      <bind-xml name="problemNb" node="element" />
    </field>
  - <field name="ptdLsL" type="java.lang.Integer" direct="true" required="false" lazy="false">
      <bind-xml name="preTestDemLowSaidLow" node="element" />
    </field>
  - <field name="ptdLsH" type="java.lang.Integer" direct="true" required="false" lazy="false">
      <bind-xml name="preTestDemLowSaidHigh" node="element" />
    </field>
  - <field name="ptdHsL" type="java.lang.Integer" direct="true" required="false" lazy="false">
      <bind-xml name="preTestDemHighSaidLow" node="element" />
    </field>
  - <field name="ptdHsH" type="java.lang.Integer" direct="true" required="false" lazy="false">
      <bind-xml name="preTestDemHighSaidHigh" node="element" />
    </field>
```

Sample of an .xml file

Below we included two extracts of an .xml file. The first one presents with general information about the subject’s sessions and metacognitive values obtained.

```
<Student>
  <FirstName> XXXXX </FirstName>
  <LastName> XXXXXX </LastName>
  <Age>20</Age>
  <Female>false</Female>
  <sessionNb>8</sessionNb>
  <sessionTimings>
    <begins>2002-10-30T12:25:16.020Z</begins>
    <ends>2002-10-30T13:06:25.614Z</ends>
  </sessionTimings>
  <sessionTimings>
    <begins>2002-10-31T11:22:34.456Z</begins>
    <ends>2002-10-31T12:18:32.030Z</ends>
  </sessionTimings>
  <sessionTimings>
    <begins>2002-11-03T11:28:14.961Z</begins>
    <ends>2002-11-03T11:52:59.336Z</ends>
  </sessionTimings>
  <problemNb>14</problemNb>
  <preTestDemLowSaidLow>0</preTestDemLowSaidLow>
  <preTestDemLowSaidHigh>2</preTestDemLowSaidHigh>
</Student>
```
Below it is another extract of the same .xml file that registered information regarding one problem attempted by the subject (problem 2).

- <ProblemAttempt index="2" stage="9" gave-up="false" time-up="false">
  <start>2002-10-30T12:40:47.905Z</start>
  <assess>2002-10-30T12:44:44.889Z</assess>
  <solve>2002-10-30T12:46:55.738Z</solve>
  <answer>2002-10-30T12:49:34.355Z</answer>
  <check>2002-10-30T12:50:42.519Z</check>
  <finish>2002-10-30T12:51:09.522Z</finish>
  <trend-givens-interpretation>constant</trend-givens-interpretation>
  <trend-relations-interpretation>constant</trend-relations-interpretation>
  <understanding-givens>2</understanding-givens>
  <understanding-relations>2</understanding-relations>
  <difficulty>a bit difficult</difficulty>
  <solving-prediction>2</solving-prediction>
  - <list-strategies-monitoring selected="true">
    <description>Read the problem more than once</description>
  </list-strategies-monitoring>
  - <list-strategies-monitoring selected="true">
    <description>Read the problem to separate the important parts</description>
  </list-strategies-monitoring>
  - <list-strategies-monitoring selected="true">
    <description>Think of a related problem I have already done and use it as a model</description>
  </list-strategies-monitoring>
  - <list-strategies-monitoring selected="false">
    <description>Before starting to solve the problem, think what I am supposed to learn from it</description>
  </list-strategies-monitoring>
  - <list-strategies-monitoring selected="false">
    <description>Read the problem and determine which parts I don’t understand well</description>
  </list-strategies-monitoring>
  - <list-strategies-monitoring selected="false">
    <description>
  </list-strategies-monitoring>
Review the basic concepts that are not clear, before attacking the problem.

- Set a goal to myself and think about the steps to reach this goal.

- Stop and review each step to see if I made a mistake.

- Reread the problem to check for forgotten important parts.

- Change strategies if I get lost and confused and don’t seem to move anywhere.

- Think about a way of checking to see if my solution is correct.

- Review all I did to make sure I am not forgetting anything.

- Reread the task description and ask myself if my solution really meets the task goal.

I did read the problem through more than once. I realised that Emily was not relevant to the problem, and so eliminated her from the problem. It was not necessary to think of a related problem. I worked out how Paul related to Matt, and then how Matt related to Kevin. The second problem proved easier than originally thought. I spent most time thinking how to formulate the equations relating to Fred and then it was easy from there.

- Time-line
  - Begins 2002-10-31T11:32:05.009Z
  - Ends 2002-10-31T11:32:25.900Z
- Demonstr-givens 2
- Demonstr-performance 2
- Demonstr-relations 2


## B.5 Main Classes in MIRA

MIRA has 72 classes, where a few of them are reused or extended from free-distributed libraries. The main classes in MIRA are briefly described below, organized into two groups: Model Classes and User Interface Classes. Figure B.3 presents a class relation diagram (UML diagram) with the main classes in MIRA.

### Model Classes in MIRA

- **Class Application** Top level class, which is never instantiated, but keeps static references to many constants of the software, the main model class (Student) and the top level user interface window (MainFrame).

- **Class Student** This is the central class of the Model. From this class it is possible to reach all student’s information and the Metacognitive Model (KMA and KMB values).

- **Class ProblemAttempt** Main Model class for student’s problem attempt. It keeps information about everything related to student’s interaction during problem solving, including performance and duration of each stage of problem solving (time for answering the problem, answering quiz, checking teacher’s solution, etc.).

- **Class Problem** Model class for problem. It keeps the description of the problem, its parts and solution.

- **Class Quiz** Model class for quiz. It keeps the description of quizzes, multiple choice answers and solutions.

- **Class PALToolUsage** This class keeps information about the use of PAL Tool in a problem attempt, such as the givens and unknowns typed and the number of times and moments where the PAL Tool was used in a problem solving attempt.

### User Interface Classes in MIRA

- **Class MainFrame** Creates all GUI Frames and add them to the application Desktop.

- **PreTocFrame** Application initial Frame. Defines content and organization of the Introductory Window. This window is a split pane with links to HTML introductory files.
Class **PreClr** Controller of the pre-login windows and functions. Activate HTML links and load HTML files with introduction and opens Registration Frame or Login Frame.

Class **OverviewFrame** It presents a visual summary of the sequence of stages and activities in MIRA. This frame is showed before each new stage only for the first session. It becomes notVisible for the second session onwards.

Class **TutorialFrame** The tutorial has basic information about the objective of each stage and essential functional information about how to interact with MIRA. The tutorial is context-sensitive and is made of HTML documents that are called according to the current stage and windows open.

Class **ProblemPreFrame** Its presents the pre-task reflection activities before each problem).

Class **ProblemAssessingFrame** Frame that contains the problem description and the self-assessment questionnaire. It is shown before the ProblemSolvingFrame. It contains multiple-choice questions about metacognitive assessment of translation skills of givens and relations and assessment of the level of difficulty of the problem.

Class **ProblemSolvingFrame** Frame for the Problem Solving stage. It presents the problem text, the timer (that counts down the available time to do the problem), the menu for accessing the calculator, the PAL Tool and buttons to indicate actions of “giving up” or “finishing” the problem.

Class **ProblemAnsweringFrame** Frame that contains a multiple-choice from where the student selects the best match answer to the problem translated. It also includes a self-assessment of how the answer was worked out.

Class **ProblemCheckingFrame** Frame that shows the teacher’s solution to the problem.

Class **ProblemPostFrame** Frame for post-task reflection. It contains the ProblemTimelinePanel.

Class **ProblemTimelinePanel** Panel with Problem Timeline Graph and summary of performance. Contains graph and summary of last problem activities Vs. time spent in each activity.

Class **MiraVerticalBarChart** It is a Gant Plotting API that extends the Class VerticalBarChart of the open source JAVA packet called Graphiti4.

Class **QuestionCheckBoxAnswersPanel** Reusable class that builds quizzes or multiple-choice questions using checkbuttons.

---

4Graphiti is a set of java classes which one can use to plot numerical data in the form of bar charts (http://sourceforge.net/projects/graphiti/)
Figure B.3: Diagram of MIRA classes. UML Class relation diagram of the main classes in MIRA.
Appendix C

Experiment Analysis: Graphs and Tables

This appendix has some of the graphs and tables generated in the analysis of the experiment. We have decided to not including them in the main document to keep the readability of Chapter 8. The information of this Appendix serve mainly as an additional source of information. The order of appearance of the tables and figures follows the order they are referred to in Chapter 8.

C.1 Normality test

<table>
<thead>
<tr>
<th>Variable</th>
<th>Sig. value</th>
<th>normally distributed?</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre-test scores</td>
<td>1.00</td>
<td>yes</td>
</tr>
<tr>
<td>post-test scores</td>
<td>0.16</td>
<td>yes</td>
</tr>
<tr>
<td>absolute difference posttest - pretest</td>
<td>0.63</td>
<td>yes</td>
</tr>
<tr>
<td>relative difference posttest - pretest</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>GCSE scores</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>Age</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>KMA in pre-test</td>
<td>0.02</td>
<td>no</td>
</tr>
<tr>
<td>KMA in post-test</td>
<td>0.05</td>
<td>yes</td>
</tr>
<tr>
<td>difference KMA posttest - KMA pretest</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>final KMA in MIRA</td>
<td>0.50</td>
<td>yes</td>
</tr>
<tr>
<td>final KMB in MIRA</td>
<td>0.52</td>
<td>yes</td>
</tr>
<tr>
<td>KMB in pre-test</td>
<td>0.36</td>
<td>yes</td>
</tr>
<tr>
<td>KMB in post-test</td>
<td>0.11</td>
<td>yes</td>
</tr>
<tr>
<td>number of problems attempted</td>
<td>0.04</td>
<td>no</td>
</tr>
<tr>
<td>number of problems given-up</td>
<td>0.01</td>
<td>no</td>
</tr>
<tr>
<td>number of correct answers</td>
<td>0.08</td>
<td>yes</td>
</tr>
<tr>
<td>number of wrong answers</td>
<td>0.01</td>
<td>no</td>
</tr>
<tr>
<td>number of answers with minor errors</td>
<td>0.01</td>
<td>no</td>
</tr>
<tr>
<td>number of answers recognized as correct</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>number of problems timed-up</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>total correct/total problems attempted</td>
<td>0.10</td>
<td>yes</td>
</tr>
<tr>
<td>total given up/total problems attempted</td>
<td>0.02</td>
<td>no</td>
</tr>
<tr>
<td>total timed up/total problems attempted</td>
<td>0.00</td>
<td>no</td>
</tr>
<tr>
<td>total wrong/total problems attempted</td>
<td>0.01</td>
<td>no</td>
</tr>
</tbody>
</table>

Table C.1: Normality test: Shapiro-Wilk. The variable is normally distributed if the Sig. value is greater than the $\alpha$ ($\alpha=0.05$) (Field and Hole, 2003).
Appendix C. Experiment Analysis: Graphs and Tables

C.2 Initial profile of the groups

<table>
<thead>
<tr>
<th>Gender</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>10</td>
<td>76,9%</td>
</tr>
<tr>
<td>Male</td>
<td>3</td>
<td>23,1%</td>
</tr>
<tr>
<td>Total</td>
<td>13</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Age - categorized</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>18-21</td>
<td>5</td>
<td>38,5%</td>
</tr>
<tr>
<td>22-29</td>
<td>5</td>
<td>38,5%</td>
</tr>
<tr>
<td>30-39</td>
<td>3</td>
<td>23,1%</td>
</tr>
<tr>
<td>Total</td>
<td>13</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Motivation in maths</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tally</td>
<td>10</td>
<td>76,9%</td>
</tr>
<tr>
<td>very</td>
<td>2</td>
<td>15,4%</td>
</tr>
<tr>
<td>extremely</td>
<td>1</td>
<td>7,7%</td>
</tr>
<tr>
<td>Total</td>
<td>13</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Familiar learning appl.</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>unfamiliar</td>
<td>6</td>
<td>38,5%</td>
</tr>
<tr>
<td>newcomer</td>
<td>1</td>
<td>7,7%</td>
</tr>
<tr>
<td>beginner</td>
<td>2</td>
<td>13,6%</td>
</tr>
<tr>
<td>average</td>
<td>3</td>
<td>19,3%</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GCSE grades</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>5</td>
<td>38,5%</td>
</tr>
<tr>
<td>A*</td>
<td>1</td>
<td>7,7%</td>
</tr>
<tr>
<td>B</td>
<td>5</td>
<td>38,5%</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>15,4%</td>
</tr>
<tr>
<td>Total</td>
<td>13</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A Levels grades</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
<td>20,0%</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>40,0%</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>40,0%</td>
</tr>
<tr>
<td>Total</td>
<td>3</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

Table C.2: General profile of the experimental group: (a) gender distribution; (b) age distribution; (c) motivation in maths; (d) familiarity with ILEs; (e) and (f) school grades.

<table>
<thead>
<tr>
<th>Gender</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>8</td>
<td>75,0%</td>
</tr>
<tr>
<td>Male</td>
<td>3</td>
<td>25,0%</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Age - categorized</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>18-21</td>
<td>8</td>
<td>69,2%</td>
</tr>
<tr>
<td>22-29</td>
<td>4</td>
<td>33,3%</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Motivation in maths</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>1</td>
<td>8,3%</td>
</tr>
<tr>
<td>not at all</td>
<td>4</td>
<td>33,3%</td>
</tr>
<tr>
<td>a bit</td>
<td>5</td>
<td>41,7%</td>
</tr>
<tr>
<td>fairly</td>
<td>1</td>
<td>8,3%</td>
</tr>
<tr>
<td>very</td>
<td>1</td>
<td>8,3%</td>
</tr>
<tr>
<td>Total</td>
<td>11</td>
<td>91,7%</td>
</tr>
<tr>
<td>Missing System</td>
<td>1</td>
<td>8,3%</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Familiar learning appl.</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>unfamiliar</td>
<td>3</td>
<td>28,5%</td>
</tr>
<tr>
<td>newcomer</td>
<td>1</td>
<td>9,1%</td>
</tr>
<tr>
<td>beginner</td>
<td>5</td>
<td>41,7%</td>
</tr>
<tr>
<td>average</td>
<td>3</td>
<td>28,5%</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GCSE grades</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
<td>19,3%</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>29,9%</td>
</tr>
<tr>
<td>C</td>
<td>7</td>
<td>59,3%</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A Levels grades</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>100,0%</td>
</tr>
</tbody>
</table>

Table C.3: General profile of the control group: (a) gender distribution; (b) age distribution; (c) motivation in maths; (d) familiarity with ILEs; (e) and (f) school grades.
Groups’ differences in algebra performance and GCSE maths scores

### Table C.4: Testing differences between groups’ pre-test scores. The Mann Whitney U test shows that there is no statistically significant difference between the groups.

<table>
<thead>
<tr>
<th>Pre-test Scores</th>
<th>N</th>
<th>Mean Rank</th>
<th>Sum of Ranks</th>
<th>Pre-test scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>12</td>
<td>13,72</td>
<td>157,00</td>
<td></td>
</tr>
<tr>
<td>Experimental</td>
<td>13</td>
<td>12,15</td>
<td>150,00</td>
<td>Mann-Whitney U</td>
</tr>
<tr>
<td>Total</td>
<td>25</td>
<td></td>
<td></td>
<td>Z</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Asymp. Sig. (2-tailed)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Exact Sig. (2-tailed)</td>
</tr>
</tbody>
</table>

Figure C.1: Groups’ pre-test performance profile: (a) descriptive values for pre-test scores; (b) frequencies of pre-test scores organized by categories.
Appendix C. Experiment Analysis: Graphs and Tables

Table C.5: Testing groups’ GCSE scores differences. The Mann-Whitney U test shows a statistically significant difference between the groups. The significant difference disappears if we remove the subjects with A Levels in maths.

Analysis of correlations: GCSE maths scores and pre-test scores

Table C.6: Correlation test: GCSE scores and pre-test. A correlation between GCSE scores and pre-tests scores was not found for either group.
C.3 Differences in pre-test and post-test scores

Table C.7: Descriptive values for pre-test and post-test.

<table>
<thead>
<tr>
<th>Statistics Group Experimental</th>
<th>Pre-test scores</th>
<th>Post-test scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Valid</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Missing</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mean</td>
<td>4,90154</td>
<td>4,12581</td>
</tr>
<tr>
<td>Median</td>
<td>5,00000</td>
<td>4,00000</td>
</tr>
<tr>
<td>Mode</td>
<td>0.50670</td>
<td>2.50670</td>
</tr>
<tr>
<td>Std. Deviation</td>
<td>1.812796</td>
<td>2.465896</td>
</tr>
<tr>
<td>Variance</td>
<td>2.802568</td>
<td>5.827692</td>
</tr>
<tr>
<td>Range</td>
<td>5.50000</td>
<td>8.00000</td>
</tr>
<tr>
<td>25th Percentile</td>
<td>3.50000</td>
<td>2.50000</td>
</tr>
<tr>
<td>50th Percentile</td>
<td>5.00900</td>
<td>6.00900</td>
</tr>
<tr>
<td>75th Percentile</td>
<td>6.25000</td>
<td>8.00900</td>
</tr>
</tbody>
</table>

Table C.8: Wilcoxon test for pre- and post-test scores changes. The test for repeated measures shows that there is no statistically significant change.

<table>
<thead>
<tr>
<th>Statistics Group Control</th>
<th>Pre-test scores</th>
<th>Post-test scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Valid</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Missing</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mean</td>
<td>5,00000</td>
<td>5,25900</td>
</tr>
<tr>
<td>Median</td>
<td>5,70000</td>
<td>4,00000</td>
</tr>
<tr>
<td>Mode</td>
<td>4,50670</td>
<td>3,50670</td>
</tr>
<tr>
<td>Std. Deviation</td>
<td>2.679891</td>
<td>2.530163</td>
</tr>
<tr>
<td>Variance</td>
<td>7.101910</td>
<td>8.431190</td>
</tr>
<tr>
<td>Range</td>
<td>0.50000</td>
<td>7.50000</td>
</tr>
<tr>
<td>25th Percentile</td>
<td>3.75000</td>
<td>2.50000</td>
</tr>
<tr>
<td>50th Percentile</td>
<td>5.75000</td>
<td>4.80900</td>
</tr>
<tr>
<td>75th Percentile</td>
<td>7.50000</td>
<td>7.87500</td>
</tr>
</tbody>
</table>

Table C.7: Descriptive values for pre-test and post-test.

<table>
<thead>
<tr>
<th>Group = Experimental</th>
<th>Ranks</th>
<th>Test Statistic&lt;sup&gt;a&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>Mean Rank</td>
</tr>
<tr>
<td>Post-test &amp; Pre-test</td>
<td>9°</td>
<td>5.72</td>
</tr>
<tr>
<td>Scores: Negative</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Positive</td>
<td>12°</td>
<td>8.83</td>
</tr>
<tr>
<td>Ties</td>
<td>1°</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>13°</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Group = Control</th>
<th>Ranks</th>
<th>Test Statistic&lt;sup&gt;b&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>Mean Rank</td>
</tr>
<tr>
<td>Post-test &amp; Pre-test</td>
<td>7°</td>
<td>5.29</td>
</tr>
<tr>
<td>Scores: Negative</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Positive</td>
<td>6°</td>
<td>8.00</td>
</tr>
<tr>
<td>Ties</td>
<td>1°</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>12°</td>
<td></td>
</tr>
</tbody>
</table>

<sup>a</sup> Based on positive ranks.
<sup>b</sup> Wilcoxon Signed Ranks Test.
### Table C.9: Mann-Whitney tests for groups' score differences: there is no statistically significant difference between control and experimental group.

C.4 Group differences in problem solving performance in MIRA
Appendix C. Experiment Analysis: Graphs and Tables

### Table C.10
Table C.10: Test for differences of total problems attempted in MIRA. Mann-Whitney test shows that there is a statistically significant difference between groups.

<table>
<thead>
<tr>
<th>Group</th>
<th>N</th>
<th>Mean Rank</th>
<th>Sum of Ranks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of problems attempted</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>12</td>
<td>16.88</td>
<td>202,50</td>
</tr>
<tr>
<td>Experimental</td>
<td>13</td>
<td>9.42</td>
<td>122,50</td>
</tr>
<tr>
<td>Total</td>
<td>25</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table C.11
Table C.11: Test for groups’ differences of types of responses in MIRA. Mann-Whitney test shows that there is a statistically significant difference between groups for correct and almost correct answers. And not for wrong answers and answers recognized as correct.

<table>
<thead>
<tr>
<th>Test Statistics</th>
<th>Number of problems attempted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mann-Whitney U</td>
<td>31,000</td>
</tr>
<tr>
<td>Wilcoxon W</td>
<td>122,000</td>
</tr>
<tr>
<td>Z</td>
<td>-2.552</td>
</tr>
<tr>
<td>Asympt. Sig. (2-tailed)</td>
<td>.011</td>
</tr>
<tr>
<td>Exact Sig. [2*1-tailed] Sig.</td>
<td>.010</td>
</tr>
</tbody>
</table>

Test Statistics

<table>
<thead>
<tr>
<th>Test Statistics</th>
<th>Total mental (percent)</th>
<th>Total mental (percent) attempted</th>
<th>Total mental (percent) attempted</th>
<th>Total mental (percent) attempted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mann-Whitney U</td>
<td>57,500</td>
<td>45,500</td>
<td>45,500</td>
<td>45,500</td>
</tr>
<tr>
<td>Wilcoxon W</td>
<td>148,000</td>
<td>151,000</td>
<td>150,000</td>
<td>150,000</td>
</tr>
<tr>
<td>Z</td>
<td>-1.110</td>
<td>-1.240</td>
<td>-1.016</td>
<td>-1.060</td>
</tr>
<tr>
<td>Asympt. Sig. (2-tailed)</td>
<td>.084</td>
<td>.190</td>
<td>.066</td>
<td>.066</td>
</tr>
<tr>
<td>Exact Sig. [2*1-tailed] Sig.</td>
<td>.210</td>
<td>.196</td>
<td>.077</td>
<td>.098</td>
</tr>
</tbody>
</table>
Appendix C. Experiment Analysis: Graphs and Tables

### Table C.12: Test for groups’ differences of problems given up in MIRA. Mann-Whitney test shows that there is not a statistically significant difference between groups

<table>
<thead>
<tr>
<th>Test Statistics</th>
<th>perc</th>
<th>GiveUp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mann-Whitney U</td>
<td>58,000</td>
<td></td>
</tr>
<tr>
<td>Wilcoxon W</td>
<td>150,000</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>-1,042</td>
<td></td>
</tr>
<tr>
<td>Asymp. Sig. (2-tailed)</td>
<td>.297</td>
<td></td>
</tr>
<tr>
<td>Exact Sig. [2*(1-tailed Sig.)]</td>
<td>.320</td>
<td></td>
</tr>
</tbody>
</table>

Table C.12: Test for groups’ differences of problems given up in MIRA. Mann-Whitney test shows that there is not a statistically significant difference between groups

### Table C.13: Test for groups’ differences of time ups in MIRA. Mann-Whitney test shows that there is a marginally statistically significant difference between groups

<table>
<thead>
<tr>
<th>Ranks</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Rank</td>
<td>Sum of Ranks</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perc</td>
<td>Control</td>
<td>12</td>
<td>15.50</td>
</tr>
<tr>
<td>TimeUps</td>
<td>Experimental</td>
<td>13</td>
<td>10.09</td>
</tr>
<tr>
<td>Total</td>
<td>25</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test Statistics</th>
<th>perc</th>
<th>TimeUps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mann-Whitney U</td>
<td>48,000</td>
<td></td>
</tr>
<tr>
<td>Wilcoxon W</td>
<td>139,000</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>-1.673</td>
<td></td>
</tr>
<tr>
<td>Asymp. Sig. (2-tailed)</td>
<td>.094</td>
<td></td>
</tr>
<tr>
<td>Exact Sig. [2*(1-tailed Sig.)]</td>
<td>.110</td>
<td></td>
</tr>
</tbody>
</table>

Table C.13: Test for groups’ differences of time ups in MIRA. Mann-Whitney test shows that there is a marginally statistically significant difference between groups
C.5 Analysis of KMA and KMB

Table C.14: Descriptive values of KMA in pre- and post-test. In the experimental group was observed an important three times increase in the upper quartile (KMA Pre-test 75th percentile=0.2 and Post-test 75th percentile=0.6), suggesting an improvement of this metacognitive skill for the this group.

Table C.15: Frequencies of the categories of KMA. Groups frequencies measured in pre- and post-tests.
Table C.16: Wilcoxon test for KMA changes: shows no statistically significant difference in repeated measure.

```latex
\begin{table}
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\textbf{Group = Experimental} & \textbf{Ranks} & \textbf{Test Statistic}^b \hline
\textbf{KMA in posttest - KMA in pretest} & & & \\
Negative Ranks & 4^a & 4,00 & 12,69 \\
Positive Ranks & 5^a & 4,00 & 10,69 \\
Total & 12 & & \\
\hline
\textbf{KMA in posttest - KMA in pretest} & & & \\
Asym. Sig. (2-tailed) & & & 7.83 \hline
\end{tabular}
\end{table}

\begin{table}
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\textbf{Group = Control} & \textbf{Ranks} & \textbf{Test Statistic}^b \hline
\textbf{KMA in posttest - KMA in pretest} & & & \\
Negative Ranks & 4^a & 4,00 & 17,64 \\
Positive Ranks & 5^a & 3,50 & 10,69 \\
Total & 12 & & \\
\hline
\textbf{KMA in posttest - KMA in pretest} & & & \\
Asym. Sig. (2-tailed) & & & 7.83 \hline
\end{tabular}
\end{table}

\textbf{a.} KMA in posttest = KMA in pretest  \\
\textbf{b.} Based on posttest ranks.  \\
\textbf{c.} KMA in posttest = KMA in pretest  \\
\textbf{b.} Wilcoxon Signed Ranks Test
```
### Appendix C. Experiment Analysis: Graphs and Tables

**Table C.17:** Table of frequencies: pre-test KMA vs. post-test KMA. Each cell in the resulting table represents a unique combination of specific values of crosstabulated variables. It allows us to examine the nature of the changes in the KMA categories.

<table>
<thead>
<tr>
<th></th>
<th>KMA post-test</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Low</td>
<td>Average</td>
<td>High</td>
<td>Total</td>
</tr>
<tr>
<td>group = experimental</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>kma pre</td>
<td>high</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>low</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>total</td>
<td>1</td>
<td>6</td>
<td>4</td>
<td>13</td>
</tr>
</tbody>
</table>

**Table C.18:** Mann-Whitney test for (△KMA) differences. The test shows no statistically significant difference between the groups.

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ranks</td>
<td>mean rank</td>
<td>sum of ranks</td>
<td>diff (KMA)</td>
<td>post</td>
<td>pre</td>
<td>diff (KMA)</td>
<td>post</td>
</tr>
<tr>
<td>group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>control</td>
<td>12</td>
<td>11.67</td>
<td>140.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>experimental</td>
<td>12</td>
<td>12.22</td>
<td>168.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>total</td>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mann whitney U</td>
<td>62,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>z</td>
<td></td>
<td>-0.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>asymp. sig (2-tailed)</td>
<td>0.89</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>excat sig (2-tailed)</td>
<td>0.89</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**C.6 Analysis of correlations**
Appendix C. Experiment Analysis: Graphs and Tables

### Group = Experimental

<table>
<thead>
<tr>
<th>KMB pre-test categorized</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>1</td>
<td>2.7</td>
</tr>
<tr>
<td>Random</td>
<td>2</td>
<td>15.4</td>
</tr>
<tr>
<td>Optimist</td>
<td>8</td>
<td>69.2</td>
</tr>
<tr>
<td>Realist</td>
<td>1</td>
<td>7.7</td>
</tr>
<tr>
<td>Total</td>
<td>13</td>
<td>100.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KMB post-test categorized</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>9</td>
<td>69.2</td>
</tr>
<tr>
<td>Optimist</td>
<td>4</td>
<td>30.8</td>
</tr>
<tr>
<td>Total</td>
<td>13</td>
<td>100.0</td>
</tr>
</tbody>
</table>

### Group = Control

<table>
<thead>
<tr>
<th>KMB pre-test categorized</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>3</td>
<td>25.0</td>
</tr>
<tr>
<td>Random</td>
<td>6</td>
<td>50.0</td>
</tr>
<tr>
<td>Optimist</td>
<td>3</td>
<td>25.0</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KMB post-test categorized</th>
<th>Frequency</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>2</td>
<td>16.7</td>
</tr>
<tr>
<td>Random</td>
<td>6</td>
<td>50.0</td>
</tr>
<tr>
<td>Optimist</td>
<td>4</td>
<td>33.3</td>
</tr>
<tr>
<td>Total</td>
<td>12</td>
<td>100.0</td>
</tr>
</tbody>
</table>

**Table C.19:** Frequency tables with KMB distribution

![Graph with KMB categories’ distribution](image)

**Figure C.2:** Graph with KMB categories’ distribution
Table C.20: Correlation: pre-test vs. post-test scores. Spearman correlation test shows a positive correlation.

<table>
<thead>
<tr>
<th></th>
<th>Pre-test scores</th>
<th>Post-test scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spearman's rho</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pre-test scores</td>
<td>1.000</td>
<td>0.571*</td>
</tr>
<tr>
<td>Sig. (2-tailed)</td>
<td></td>
<td>0.050</td>
</tr>
<tr>
<td>N</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Post-test scores</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spearman's rho</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pre-test scores</td>
<td>1.000</td>
<td>0.571*</td>
</tr>
<tr>
<td>Sig. (2-tailed)</td>
<td></td>
<td>0.050</td>
</tr>
<tr>
<td>N</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>

* Correlation is significant at the .05 level (2-tailed).

a. Group = Experimental
Table C.21: Correlation: KMA in pre-test vs. pre-test scores. Spearman correlation test shows a positive correlation only for the experimental group.
Figure C.3: Scatterplots involving pre-test, post-test, and KMA: correlation pre-test vs. post-test and pre-test KMA vs. pre-test scores.
Figure C.4: Scatterplot involving KMA and post-test: correlation post-test KMA vs. post-test scores shows no correlation for the experimental group.

<table>
<thead>
<tr>
<th></th>
<th>Pre-test scores</th>
<th>Motivation in maths</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spearman’s rho</td>
<td>Correlation Coefficient</td>
<td>1,000</td>
</tr>
<tr>
<td></td>
<td>Sig (2-tailed)</td>
<td>.017</td>
</tr>
<tr>
<td></td>
<td>N</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Group = Control</td>
<td></td>
</tr>
</tbody>
</table>

Table C.22: Correlation: motivation in maths vs. pre-test scores. The Spearman Correlation test shows a correlation only for the control group.
Figure C.5: Scatterplots involving motivation in maths and other variables: shows no correlation between the stated motivation in maths and $\Delta \text{score}$, $\Delta \text{KMA}$, KMA in pre-test, and $\Delta \text{KMA}$. 
Appendix D

Experiment Materials

This appendix contains all materials used in the main experiment with MIRA that took place between the 29th of October and the 22nd of November of 2002

List of Materials

The materials are presented in the following order:

1. Model of the experiment advertisement.
2. Consent Form.
3. Participation Information.
4. Questionnaire about Strategies.
5. Pre-test: Part I.
6. Pre-test: Part II.
7. Observation Sheet.
8. Introductory text with definitions and one example of Algebra Concepts.
10. Sheet to workout the problem solution.
12. Instructions for sessions 2 and 3.
14. Post-test: Part II.
15. MIRA Assessment Questionnaire.
What is it about?
You will be coached by a computer to improve problem-solving skills.
You DON’T need to be good in maths! (In fact it helps if you are not)

How long does it take?
You must come for 3 sessions of 1 hour each on 3 different days.

How do I sign in?
Go to room 3C16 in COGS (facing reception, follow corridor to your left) and put your name and email on the timetable on the door.

What do I get?
£21 for the 3 hours (paid at the end of the 3rd session)
Insight about your problem solving skills.

Any questions: call ext. 7543 Claudia (claudiag@cogs.susx.ac.uk)
Consent Form

Thank you for agreeing to take part in this experiment. You will be asked to solve some algebra word problems using a piece of software called MIRA, and during the experiment the session will be electronically logged.

If for any reason you wish to withdraw you have the right to do so. Your formal consent is required to confirm that your participation is voluntary and that you are aware that you have the right to refuse or withdraw from the experiment at any time.

I understand the conditions of the experiment and that it is part of a scientific study, the results of which may be published but my name will not be disclosed. I hereby consent to participate on this study.

Print Name:       Date:
Signature:

Participant No.  ________

Record of Participation:

<table>
<thead>
<tr>
<th></th>
<th>Session 1</th>
<th>Session 2</th>
<th>Session 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date &amp; Signature</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Amount to be paid</td>
<td>£7</td>
<td>£7</td>
<td>£7</td>
</tr>
</tbody>
</table>

The total amount of £21 will be paid at the end of the 3rd session for participants that come for all the three sessions.
Participant Information

Participant No. ______

[ ] Male [ ] Female    Age:_____

My Programme at University is: [ ] Psychology    [ ] other: ______________________

Year: ______

Is English your first language? [Yes] [No ]

Do you have any special needs that might affect your learning/studying? [Yes] [No ]
If yes, what is it:

How familiar are you with computer learning applications? Please read the descriptions of each of the five stages and place a check mark beside the stage that best describes you.

<table>
<thead>
<tr>
<th>Unfamiliar</th>
<th>Newcomer</th>
<th>Beginner</th>
<th>Average</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>I have no experience using computer applications for learning.</td>
<td>I have attempted to use computer applications for learning, but I still need quite a lot of help.</td>
<td>I am able to use basic functions in a limited number of computer learning applications.</td>
<td>I have demonstrated a general competency in a number of computer learning applications.</td>
<td>I have acquired the ability to competently use a broad spectrum of computer technologies for learning.</td>
</tr>
</tbody>
</table>

I undertook my last maths course:
[ ] 6 months ago
[ ] 1 year ago
[ ] 2 years ago
[ ] 3 years ago
[ ] more than 3 years ago

In my last school my general performance in maths was (please also indicate the course you undertook):
[ ] A    [ ] A levels
[ ] B    [ ] GCSE
[ ] C    [ ] other _____________
[ ] D
[ ] E
[ ] lower

How motivated are you normally to solve maths problems?

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not at all</td>
<td>A bit</td>
<td>Fairly</td>
<td>Very</td>
<td>Extremely</td>
</tr>
</tbody>
</table>
Participant No.___________

**Instructions:**

Make a list of things you consider important to do, to think, or to know when one is about to solve a maths problem (ex: to read the problem, to revise the topic before, etc.). Use the middle column in the table below to include your list.

When you finish your list, then tick the on boxes on the left column for those things you normally apply when you are solving a maths problem. Leave the right column empty.

<table>
<thead>
<tr>
<th>I apply</th>
<th>I consider important to do, to know or to think when one is about to solve a maths problem to …</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Experiment: MIRA

Participant No._________

**Algebra Terminology:**
An **expression** is a combination of numbers, variables and operations. It is a "phrase" that represents a number.
An **equation** sets two expressions equal to each other.
For example, $3y + 17$ is an expression, but $3y + 17 = 50$ is an equation.

### Pre Test – PART 1

**Instructions:** Read each problem below and answer the question in the box. BUT don't solve the problems!

1. John and his wife Peg have been saving for 7 months to give the 5 charities they support some extra money for Christmas. John has saved 972 pounds and Peg has saved "p" pounds. They give each charity the same amount. Write an **expression** for how much each charity gets.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   
   [ ] YES [ ] NO

2. A doctor gave you “y” pills and told you to take 1 every 30 minutes. If you take the first one now, write an **expression** for how many hours later you will be taking the last one.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   
   [ ] YES [ ] NO

3. Peter is a journeyman bricklayer. He can lay 45 bricks per hour. His boss is a master bricklayer and he can lay 60 bricks per hour. Peter got to work three hours before his boss and has been building a foundation. Then his boss joins him and they work together. Write an **expression** for the number of bricks laid after "h" hours where "h" is the total number of hours that Peter has been working.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   
   [ ] YES [ ] NO

4. Write an **equation** that translates the following relational statement: “There are six times as many students as professors at this university”.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   
   [ ] YES [ ] NO

5. Alan’s father is $x$ years old and he is 15 years older than twice Alan’s age. Write an **expression** for Alan’s age.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   
   [ ] YES [ ] NO
Algebra Terminology:
An expression is a combination of numbers, variables and operations. It is a “phrase” that represents a number.
An equation sets two expressions equal to each other.
For example, $3y + 17$ is an expression, but $3y + 17 = 50$ is an equation.

Pre Test – PART 2

Instructions: Try to solve the problems. You have 10 minutes. The experimenter will tell you when your time is up. You can attempt the problems in any order.

1. John and his wife Peg have been saving for 7 months to give the 5 charities they support some extra money for Christmas. John has saved 972 pounds and Peg has saved “p” pounds. They give each charity the same amount. Write an expression for how much each charity gets.

2. A doctor gave you “y” pills and told you to take 1 every 30 minutes. If you take the first one now, write an expression for how many hours later you will be taking the last one.

3. Peter is a journeyman bricklayer. He can lay 45 bricks per hour. His boss is a master bricklayer and he can lay 60 bricks per hour. Peter got to work three hours before his boss and has been building a foundation. Then his boss joins him and they work together. Write an expression for the number of bricks laid after “h” hours where “h” is the total number of hours that Peter has been working.

4. Write an equation that translates the following relational statement: “There are six times as many students as professors at this university”.

5. Alan’s father is x years old and he is 15 years older than twice Alan’s age. Write an expression for Alan’s age.
Participant No.: ______

Pre-Test Results

<table>
<thead>
<tr>
<th>D High / S High:</th>
<th>D High / S Low:</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Low / S High:</td>
<td>D Low / S Low:</td>
</tr>
</tbody>
</table>

Post-Test Results

<table>
<thead>
<tr>
<th>D High / S High:</th>
<th>D High / S Low:</th>
</tr>
</thead>
<tbody>
<tr>
<td>D Low / S High:</td>
<td>D Low / S Low:</td>
</tr>
</tbody>
</table>

Observations Session 1

- Date: /
- Start: 
- Finish: 

Observations Session 2

- Date: /
- Start: 
- Finish: 

Observations Session 3

- Date: /
- Start: 
- Finish:
Example of an Algebra Word Problem

Paul has two sons and one daughter. Paul is forty years old. His son Mat is one third of Paul’s age. Kevin is the other son of Paul. He is 2 years younger than Mat. How old is Kevin?

Definitions of the Components of a Problem

The Givens

The givens of a problem are the values that are provided and will be used to solve the problem.

Example:
40 (the age of Paul) is a given.
The number of daughters of Paul = 1 is NOT a given because this information is not important to solve the problem.
Even the number of sons is not a given because we don’t need the number 2 to solve the problem.

The Goals

The goals of a problem are what we have to find. The values of the goals are unknowns (we have to find them out).

Example: Finding the age of Kevin is the goal of the problem.

The Unknowns

The unknowns are all information in the problem that we don’t know the value, but we may need to find them out to help reach the goal.

Example:
The age of Mat is unknown and we will need to find out its value to find Kevin’s age.

The Relations

The relations of the problem are the sentences that connect the givens, unknowns and goals. They are the sentences that describe some link between elements.

Example:
The sentence “His son Mat is one third of Paul’s age” makes a relation between the age of Mat and the age of Paul.
The sentence “Kevin is 2 years younger than Mat” is another relation.

Equations

They are the mathematical form of the relations. You use the arithmetic operations (+ - * /) to represent the type of relation. And you use variables to represent the unknowns.

Example:
Paul is forty years old $P = 40$
Mat is one third of Paul’s age $M = 1/3 \times 40$
Kevin is 2 years younger than Mat $K = M - 2$
This guide will help you to start your interaction with MIRA. It does not explain all features of MIRA, though. You can always use the HELP feature to know more about each screen and to understand the objectives of each activity presented.

NOTE: this version of MIRA was implemented for a laboratory experiment. That’s why it asks for a specific username, pre-test scores and it has a maximum time limit for each session.

1. Registration

1. Click Register for first time interaction; Or Click Login for returning sessions.

2. Fill in ALL personal details. The username has to follow the rule: pXXe where XX is a number between 01 and 99 (Ex: p02e)

3. Here you have two possibilities: (i) you can do the pre-test and enter the results; or (ii) enter 0 for all fields

A confirmation message of successful registration will appear. You will notice that a random number was added to your username. Take note of your complete username as you will need it for future interactions.

How to:

- Log off the session: use the combination of keys ALT + F3
- Quit MIRA: use the combination of keys ALT + F4
2. Understanding the General Screen

HELP Button – explains What to do and Goals of the current activity.

I Need a Pause Button - Click ONLY if you need to interrupt the training for some reason (it stops the internal clock that calculates the time you spend on each activity).

This image shows the sequence of activities and explains the goals of the next one.

Start Button – proceed to next activity.

3. The first activity

Here is a sample of a screen in MIRA. The top part (in white) is similar to all screens.

1. First read what you have to do in this activity.

2. Read the text of the Algebra Problem to be solved.

3. Here is the task you have to perform. Follow the instructions or click on the HELP button.

4. When you finish, click on DONE.
4. Adding / Deleting / Editing text

In some activities you can enter your own text. If you see a table like the one in the activity below, you can include text in new lines, delete text or edit previous entered text. Changes are saved automatically.

**TIP:** Click on the tab to see the text behind

**New Text:** Click on the last empty line. The line will be highlighted in blue. Type your text.

**Edit Text:** Click on the line you want to edit. The line will be highlighted in blue. Edit it.

**Delete:**
1. Click on the line you want to delete. The line will be highlighted in blue. Edit it.
2. Click on the delete button.

5. Problem Solving Window

1. Read the text labelled as "Your Task is:" It describes what you have to do in this activity.
2. Here is the description of the Algebra Problem.
3. Here are the resources you can use to help you to perform the task. Click on HELP to know more about them.
4. Your answer goes here. It is a free text area that only accepts letters, numbers and basic maths symbols (+, -, *, /, =)
5. When you've finished, click on FINISHED. If you want to give up, click on GIVE UP.

1. Click on Library of Problems

2. Select which past problem you want to see.

3. Click on the “tabs” to change from your own solution to the teacher’s solution.

7. Using the PAL Tool

In the PAL Tool tables you can type your own text or “copy & paste” from the problem text using the keys combination: 
CTRL+C (to copy); 
CTRL+V (to paste).

You can include text in new lines, delete text or edit previous entered text. Changes are saved automatically.

To Delete: 1-click on the line you want to delete. The line will be highlighted in blue.
2- Click on the delete button on the same table.

To include text: Click on the last empty cell. The cell will be highlighted in blue. Type your text.

The column “Numerical Value”: Only type numbers without formatting characters (like comma). If the cell becomes red, it means you typed an invalid character.
More about the Pal Tool ... using the “Hints” and “Give me” features:

Click here to read a hint (in the format of questions and answers) about the goals, givens, unknowns or relations of the problem.

Show hint for: choose one

To indicate which elements are involved in the relation you have described in the column "Relation Description", click on the pen symbol. A small window with the names of variables you have created and the numerical values you have typed will appear. Select only the ones that take part of the relation you want to describe. Click done. These values and variables will be the elements that will form the equations.
Participant No._________  Session: ______

Problem:_______________________
Experimenting with MIRA:
Guideline for Semi-structured Interviews

- Interviews take place at the end of each session.
- The interviews are tape recorded. At the beginning of each interview the username of the participant and the session it refers to is recorded.
- Each interview should take from 6 to 9 minutes.

Session 1 – Familiarization session:

1. How was the interaction with MIRA?
2. Did you find any of the activities presented useful? Which one?
3. Which activities you liked most? And which activities you liked least?
4. Were there any confusing parts (i.e.: you could not understand the purpose or what to do)?
5. How difficult were the problems you saw today?
6. How well did you understand the “givens” and the “relations” of those problems?
7. What was your impression of the graphs and activities that happen before and after the actual problem solving? Did they tell you anything? (only for experimental group)
8. If you can, please complete this sentence for me: “Today I learnt …”
9. Do you want to say anything else? Do you have any other comments?

Session 2 – individual training session:

1. How well did you understand the “givens” and the “relations” of the problems you solved?
2. How difficult were the problems today? Do you remember of a difficult one? Can you tell me how you dealt with this specific problem? Please recall what you did and things you thought. Let’s start in the moment you first read the problem description …
3. Can you remember if you used any strategies to solve the problems? Please tell me more about them. Can you remember if they changed from problem to problem?
4. Did you give up on any problem today? Which one? Why?
5. Do you want to say anything else? Do you have any other comments?

Session 3 – individual training session:

1. Can you tell me how was your experience using MIRA in the three sessions? Did you learn anything at all? If yes, what?
2. Can you remember if you used any strategies to solve the problems? Please tell me more about them. Did they change from problem to problem?
3. Did you give up on any problem? Why/ why not? Do you do that normally?
4. Do you think it is important to have a good judgement of your understanding of problems? Why?
5. Did you learn anything new about yourself as a maths problem solver in these sessions? Tell me more about that …
6. Do you want to say anything else? Do you have any other comments?
Instructions for Sessions 2 and 3

1. In this session you are going to use the system MIRA on your own.

2. To start, click on LOGIN in the initial menu and ask the experimenter your username. Don’t register again!

3. If you need help during the session click on the HELP button that appears on the top left hand-side of the screen. If, after reading the help content, you still need help, you can call the experimenter.

   **Remember:** press the button I NEED A PAUSE if you need to leave the room for some reason. But only for this reason.

4. You are allowed to use paper only during the problem solving stage (i.e.: the screen with the PAL Tool, timer, etc.).
   You will find sheets on the top of the computer monitor. Use them if you need it. Please pick one per problem, put the title of the problem on it, the session you are in (session 2 or 3) and your username (shown on the top left hand-side of the screen).

   But don’t forget to write your equations on the answer area in MIRA.

5. When your session with MIRA is over a message will appear on the screen. Then, you will be asked some questions (like in the previous session) by the experimenter.
Algebra Terminology:
An expression is a combination of numbers, variables and operations. It is a “phrase” that represents a number.
An equation sets two expressions equal to each other.
For example, $3y + 17$ is an expression, but $3y + 17 = 50$ is an equation.

Post Test – PART 1

Instructions: Read each problem below and answer the question in the box. BUT don’t solve the problems!

1. Rebecca makes an “h” hour car trip. For 1/4 of those hours it was raining and Rebecca drove at 40 miles per hour. The rest of the time it was sunny and she drove at 55 miles per hour. Write an expression for the total distance she drove.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   [ ] YES [ ] NO

2. Two friends, returning from a movie stopped to eat at a restaurant. After dinner, they paid their bill and noticed a bowl of mints at the front counter. Sean took 1/3 of the mints, but returned four because he had a momentary pang of guilt. Alice then took 1/4 of what was left but returned three that looked like they had been slobbered on back to the bowl. The bowl had only 21 mints left when the raid was over. Write the equations for the amount of mints were originally in the bowl.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   [ ] YES [ ] NO

3. Write an equation that translates the following relational statement: “There are six times as many students as professors at this university”.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   [ ] YES [ ] NO

4. Write the equations that would allow you to calculate the age of Carol.
Julia and Carol have the same birthday. Julia is twice as old as Carol was some years ago. At that time Julia was as old as Carol is now. Julia is 24 now. How old is Carol now?

   QUESTION: Do you think you can give the answer that is asked for this problem?
   [ ] YES [ ] NO

5. Mary has two daughters, Sarah and Catherine. Mary wins $p$ pounds on the lottery, and decides to share her winnings with her daughters. Mary keeps £300 more than she gives to Sarah. Catherine receives £100 more than her sister. Write the equations that allow you knowing how much each daughter receives.

   QUESTION: Do you think you can give the answer that is asked for this problem?
   [ ] YES [ ] NO
**Post Test – PART 2**

**Instructions:** Try to solve the problems. You have 15 minutes. The experimenter will tell you when your time is up. You can attempt the problems in any order. Please read the box below before start.

**Algebra Terminology:**
An **expression** is a combination of numbers, variables and operations. It is a “phrase” that represents a number.
An **equation** sets two expressions equal to each other.
For example, $3y + 17$ is an expression, but $3y + 17 = 50$ is an equation.

1. Rebecca makes an "h" hour car trip. For 1/4 of those hours it was raining and Rebecca drove at 40 miles per hour. The rest of the time it was sunny and she drove at 55 miles per hour. Write an **expression** for the total distance she drove.

2. Two friends, returning from a movie stopped to eat at a restaurant. After dinner, they paid their bill and noticed a bowl of mints at the front counter. Sean took 1/3 of the mints, but returned four because he had a momentary pang of guilt. Alice then took 1/4 of what was left but returned three that looked like they had been slobbered on back to the bowl. The bowl had only 21 mints left when the raid was over. Write the **equations** for the amount of mints were originally in the bowl.
3. Write an equation that translates the following relational statement: “There are six times as many students as professors at this university”.

4. Write the equations that would allow you to calculate the age of Carol. Julia and Carol have the same birthday. Julia is twice as old as Carol was some years ago. At that time Julia was as old as Carol is now. Julia is 24 now. How old is Carol now?

5. Mary has two daughters, Sarah and Catherine. Mary wins $p$ pounds on the lottery, and decides to share her winnings with her daughters. Mary keeps £300 more than she gives to Sarah. Catherine receives £100 more than her sister. Write the equations that allow you knowing how much each daughter receives.
Assessment of the MIRA Environment

Using the classifications below (helpfulness and frequency of use), give a number that best describe your interaction with each component of the MIRA Environment.

**Helpfulness:**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>don’t know</td>
<td>don’t know</td>
<td>unhelpful</td>
<td>sometimes helpful</td>
<td>consistently helpful</td>
<td>extremely helpful</td>
</tr>
</tbody>
</table>

**Frequency of use:**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>don’t know</td>
<td>don’t know</td>
<td>never</td>
<td>rarely</td>
<td>sometimes</td>
<td>very often</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>Helpfulness</th>
<th>Frequency of Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reflection vertical bar graphs and textual interpretation related to what you said and demonstrated about the givens and the relations of past problems.</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>&quot;Reflectometers&quot; (speedometers-like graphs) with system’s interpretation of your accuracy of knowledge judgment and feedback.</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Assessment of problem: understanding of the givens and relations of the problem, difficulty assessment and prediction of solving the problem.</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Planning strategies monitoring understanding, controlling errors, checking progress.</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>PAL Tool (tool for separating the givens, goals, unknowns, and relations with hints and “give me” resources)</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
</tbody>
</table>
### Helpfulness:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>don't know</td>
<td></td>
<td></td>
<td>unhelpful</td>
<td>sometimes helpful</td>
<td>consistently helpful</td>
</tr>
</tbody>
</table>

### Frequency of use:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>don't know</td>
<td></td>
<td></td>
<td>never</td>
<td>rarely</td>
<td>sometimes</td>
</tr>
</tbody>
</table>

### Component

<table>
<thead>
<tr>
<th>Component</th>
<th>Helpfulness</th>
<th>Frequency of Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Library of problems: previous problems solved with teacher’s solution and your solution.</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Algebra Basics: text about equations, variables, components of a problem, etc.</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Glossary of relations: statements in problem and equivalent arithmetic operations (addition, subtraction, multiplication and division)</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Calculator</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Answer to the problem: numerical solution to the current problem</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Timer: show/hide time left to solve the problem</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Quiz about the givens and the relations</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Teacher’s solution to the problem</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Reflection on the time spent on reflective and problem solving activities with textual summary of time spent on activities</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Written reflections about use of strategies (free text reflections)</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
</tbody>
</table>
### Assessment of the MIRA Environment

Using the classifications below (helpfulness and frequency of use), give a number that best describe your interaction with each component of the MIRA Environment.

#### Helpfulness:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>don’t know</td>
<td>unhelpful</td>
<td>sometimes helpful</td>
<td>consistently helpful</td>
<td>extremely helpful</td>
</tr>
</tbody>
</table>

#### Frequency of use:

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>don’t know</td>
<td>never</td>
<td>rarely</td>
<td>Sometimes</td>
<td>very often</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>Helpfulness</th>
<th>Frequency of Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAL Tool (tool for separating the givens, goals, unknowns, and relations</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>with hints and “give me” resources)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Library of problems: previous problems solved with teacher’s solution</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>and your solution.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Algebra Basics: text about equations, variables, components of a problem,</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>etc.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Glossary of relations: statements in problem and equivalent arithmetic</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>operations (addition, subtraction, multiplication and division)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculator</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Answer to the problem: numerical solution to the current problem</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Timer: show/hide time left to solve the problem</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Quiz about the givens and the relations</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
<tr>
<td>Teacher’s solution to the problem</td>
<td>0 1 2 3 4</td>
<td>0 1 2 3 4</td>
</tr>
</tbody>
</table>