

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

Designing CBL Systems for Complex Domains using Problem Transformation and Fuzzy Logic

**A thesis presented in partial fulfillment of
the requirements for the degree of**

**Doctor of Philosophy
in
Computer Science**

**at Massey University, Palmerston North,
New Zealand.**

**Selvarajah Mohanarajah
2007**

Acknowledgement

This research would not have been possible without the support, inspiration, and enthusiasm of my supervisors, Associate Professors R. Kemp and E. Kemp. They gave me the appropriate guidance as well as unbounded freedom, whenever it was necessary. A special thanks to Ray for keeping me as an assistant lecturer for six long years at the computer science division of IIST. I am also grateful to Janina for her support during my final months at Massey. I would also like to thank my former colleagues at IIST including the office staff and technical support staff who has extended their goodwill in numerous ways.

It was Loji, my late wife, whose constant understanding, endless patience, encouragement and love made this mission achievable. While I was studying- during our entire married life- she was awaiting to see my success. It is to her, I dedicate this work. I am also thankful to my daughters Archu and Mathu for their sacrifice and understanding when it was most required. Finally, I am forever indebted to my mother, brothers, Loji's family and our friends, without their unflinching encouragement, this thesis might not have been appeared in its present form.

Publications

The following publications are made out of this research.

1. Mohanarajah, S., R. H. Kemp, and E. Kemp. 'Towards a Computer Based Learning System for Object-Z. Proceedings of the International Conference on Computers in Education 2003 (ICCE'03), Hong Kong.
2. Mohanarajah, S., R. H. Kemp, and E. Kemp. 'Towards an Interactive Learning Environment for Object-Z'. Journal of Issues in Information Sciences and Information Technology Education Vol 1, 2004, pgs 0993-1003 ISSN: 1539-3585, Informing Science, Institute Santa Rosa, California USA. Paper presented at the International Conference on Information Science and IT Education 2004 (InSITE'04), Rockhampton, Australia (awarded Best Paper medal)
3. Mohanarajah, S., Kemp, R.H., Kemp, E., and E.Heinritch. 'Unfold the scaffold & Externalizing Learner Models', World Conference on Educational Multimedia, Hypermedia, and Telecommunications 2005 (ED-MEDIA'05), P. Kommers, G. Richards (Eds.), Association for the Advancement of Computing in Education, Norfolk, USA, pp 4004 - 4009.
4. Mohanarajah, S., R. H. Kemp, and E. Kemp. 'Intelligent Pedagogical Action Selection under Uncertainty'. International Conference on Artificial Intelligence in Education 2005 (AIED'05), Chee-Kit, L, McCalla, G., Bredewreg, B. and J. Breuker (Eds), IOS Press, Amsterdam, Netherlands, pp 881-883
5. Mohanarajah, S., Kemp, R.H., and Kemp, E.. 'Opening a Fuzzy Learner Model', Proceedings of the Workshop (LEMORE) at the International Conference on Artificial Intelligence in Education 2005 (AIED'05), Amsterdam, Netherlands.
6. Mohanarajah, S., R. Kemp, H., et al. (2006). Adaptable Scaffolding - A Fuzzy Approach. LNCS 4053 - Intelligent Tutoring Systems (ITS' 6). M. Ikeda, K. Ashley and T. W. Chan. Berlin, Springer-Verlag: 604-614.

**dedicated to
LOJI**

Abstract

Some disciplines are inherently complex and challenging to learn. This research attempts to design an instructional strategy for CBL systems to simplify learning certain complex domains. Firstly, problem transformation, a constructionist instructional technique, is used to promote active learning by encouraging students to construct more complex artefacts based on less complex ones. Scaffolding is used at the initial learning stages to alleviate the difficulty associated with complex transformation processes. The proposed instructional strategy brings various techniques together to enhance the learning experience. A functional prototype is implemented with Object-Z as the exemplar subject. Both objective and subjective evaluations using the prototype indicate that the proposed CBL system has a statistically significant impact on learning a complex domain.

CBL systems include Learner models to provide adaptable support tailored to individual learners. Bayesian theory is used in general to manage uncertainty in Learner models. In this research, a fuzzy logic based locally intelligent Learner model is utilized. The fuzzy model is simple to design and implement, and easy to understand and explain, as well as efficient. Bayesian theory is used to complement the fuzzy model. Evaluation shows that the accuracy of the proposed Learner model is statistically significant. Further, opening Learner model reduces uncertainty, and the fuzzy rules are simple and resemble human reasoning processes. Therefore, it is argued that opening a fuzzy Learner model is both easy and effective.

Scaffolding requires formative assessments. In this research, a confidence based multiple test marking scheme is proposed as traditional schemes are not suitable for measuring partial knowledge. Subjective evaluation confirms that the proposed schema is effective. Finally, a step-by-step methodology to transform simple UML class diagrams to Object-Z schemas is designed in order to implement problem transformation. This methodology could be extended to implement a semi-automated translation system for UML to Object Models.

Contents

Acknowledgement.....	i
Publications	iii
Abstract	vii
Contents	ix
List of Figures	xiii
List of Tables.....	xv
Chapter 1 <u>I</u> ntroduction	1
1.1 Background.....	1
1.2 Research Questions.....	2
1.3 Objectives	5
1.4 Scope.....	6
1.5 Methodology	6
1.5 Thesis Outline	7
Chapter 2 <u>C</u> omputer Based Learning Systems	9
2.1 Introduction.....	9
2.2 Learning Theories – An Overview.....	10
2.2.1 Behaviourism	10
2.2.2 Cognitivism	10
2.2.3 Constructivism.....	12
2.2.4 Beyond Situated Cognition- Academic Knowledge	13
2.3 CBL Systems – An Overview	14
2.3.1 From CAI to ICAI.....	14
2.3.2 From ICAI to ITS.....	14
2.3.3 From ITS to ILE.....	16
2.3.4 From ILE to Bi-modus Learning Environment	17
2.3.5 Collaborative Learning Systems	17
2.3.6 E- Learning.....	18
2.4 Learning Theories and CBL Systems.....	18
2.4.1 Constructionism and Discovery Worlds.....	19
2.4.2 Bi-modus Learning Environment.....	20
2.5 Motivation and Learning.....	21
2.6 Cognitive Apprenticeship Model.....	23
2.6.1 Scaffolding.....	23
2.7 Authoring Systems	24
2.8 Pedagogical Action Selection	25
2.8.1 Curriculum Sequencing.....	26
2.8.2 Feedback.....	26
2.9 Learner models.....	28
2.9.1 Uncertainty in Learner modelling	30
2.9.2 Opening Learner models	33
2.10 Assessment in CBL systems - The Role of MC test.....	35
2.10.1 Rewarding Partial Knowledge	37
2.11 Summary	38
Chapter 3 <u>C</u> BL Systems for Programming and Formal Methods	41

Contents

3.1 Introduction	41
3.2 Teaching Systems for Programming	42
3.2.1 Overview	42
3.2.2 Challenges and Drawbacks	42
3.2.3 CBL Systems for Programming	43
3.3 Formal Methods in Software Engineering	48
3.3.1 Overview	48
3.3.2 Formal Method Tools	50
3.3.3 Teaching and Learning Formal Methods	52
3.4 Teaching Systems for Formal Methods	53
3.4.1 Formal Method Tools for Teaching	53
3.4.2 CBL Systems for Formal Methods	54
3.5 Summary	56
Chapter 4 <u>Learning by Transformation and Transforming UML models to Object-Z</u>	57
4.1 Introduction	57
4.2 Learning by Transformation	58
4.2.1 Problem Transformation	59
4.3 Transforming UML models to Object-Z Specifications	61
4.3.1 Requirement Specification - Overview	62
4.3.2 UML Overview	62
4.3.3 Object-Z Overview	65
4.3.4 Transforming Semi-formal models to Formal models – An Overview	68
4.3.5 Formalizing UML models	73
4.3.6 Transforming UML models to Object-Z	75
4.4 Summary	83
Chapter 5 <u>Designing a Problem Transformation Based CBL System</u>	85
5.1 Introduction	85
5.2 Survey on a Formal Method (Z) and a Tool (Formalizer)	86
5.3 Designing an Instruction Strategy	88
5.3.1 Problem Domain	88
5.3.1 Refinement Unit	89
5.3.2 Scaffolding Abstraction	90
5.3.3 Case Study Approach	92
5.4 A Four Phase Instructional Model	93
5.4.1 The Four-Phases of FoPSI model	94
5.5 Overview of LOZ	96
5.6 Domain Model	97
5.6.1 Mental States	99
5.6.2 Multiple Choice tests for Formative Assessment	102
5.6.3 Misconceptions and Feedback	103
5.7 MC Tests as Scaffolding Blocks	105
5.7.1 Drawback of Traditional MC test	105
5.7.2 Designing a Confidence-Based MC Test Schema	106
5.8 Mentor Model	110
5.8.1 Pedagogical Action Selection	110
5.8.2 PAS Tables for Traditional MC tests	111
5.8.3 Fully Learner-Controlled Environment	115
5.9 Refinement Unit	117
5.10 Summary	118

Chapter 6	Designing a Fuzzy Learner Model	121
6.1	Introduction	121
6.2	Learner Model Complexity	122
6.2.1	Locally-intelligent Learner Model	122
6.3	A Simple Learner Model	123
6.4	Determining Pedagogical Actions under Uncertainty	128
6.4.1	Uncertainty Handling	129
6.4.2	Adaptable Scaffolding	130
6.4.3	Fuzzification, Rule Propagation and Defuzzification	135
6.5	Predicting the Strength of Mental States	139
6.5.1	Dynamic Relations	141
6.6	Updating the Strength of Mental States	144
6.7	Initialising Strength of Mental States	151
6.8	Determining General Learning Ability	152
6.9	Opening Learner Models	155
6.9.1	Opening Mentor Model	158
6.10	Summary and Conclusion	159
Chapter 7	Prototype – Design and Implementation	161
7.1	Introduction	161
7.2	Scope of the Prototype	162
7.3	Implementation Issues	163
7.4	System Architecture	163
7.5	Domain Model Design	167
7.6	Learner Model Design	170
7.7	Fuzzy Model for Uncertainty Handling	171
7.8	Mentor Model Design	174
7.9	Database Design	176
7.10	Interface Design	182
7.11	Opening the Learner Model	186
7.12	Summary	187
Chapter 8	Evaluation and Discussion	189
8.1	Introduction	189
8.2	Research Claims	190
8.3	Evaluation Strategies	191
8.3.1	Evaluating the System: Empirical Study	192
8.3.2	Evaluating the Learner Model: Log Files	195
8.3.3	Questionnaire	198
8.3.4	Evaluating CBM schema	199
8.4	Pilot Study	201
8.5	Evaluation Process	202
8.5.1	Empirical Study	202
8.5.2	Log Files for Learner Model Accuracy	203
8.6	Results and Discussion	204
8.6.1	Hypothesis– 1: Effectiveness of LOZ	204
8.6.2	Hypothesis – 2: Accuracy of the Learner Model	208
8.6.3	Evaluating CBM Schema	211
8.6.4	Subjective Analysis	212
8.6.5	Summary of Results & Discussion	215

Contents

8.7 Summary	217
Chapter 9_CBL system for Object-Z – Phase II	221
9.1 Introduction	221
9.2 A CBL System for Object-Z - Phase-II.....	222
9.2.1 Mentor Model	222
9.2.2 Domain Model.....	229
9.2.3 Learner model	229
9.2.4 Refinement Unit	234
9.3 Transforming UML models to Object-Z.....	235
9.3.1 Overview	235
9.3.2 Transforming Structure of a Class Diagram to a Class Schema	236
9.3.3 Transforming Methods to Operation schemas.....	239
9.4 Summary	247
Chapter 10_Conclusions and Future Research.....	249
10.1 Conclusions.....	249
10.2 Contributions.....	251
10.3 Future Research	252
References	257
Appendix A: Survey on Z and Formalizer.....	268
Appendix B: Prototype Screen Shots.....	277
Appendix C: Prototype Code	295
Appendix D: Final Evaluation	297
Appendix E: Fuzzification Mechanism for Prediction.....	321

List of Figures

Figure 2.1 Stages of an Information Processing Model (after (Reed 2003, p.71)).....	11
Figure 2.2 Intelligent Tutoring Systems (after Shute <i>et al</i> (1995)).....	15
Figure 2.3 Types of Motivation (Ryan <i>et al.</i> , 2000, p. 72).....	22
Figure 2.4 A Decision-Making Framework for Feedback (Mason <i>et al.</i> 1999).....	28
Figure 2.5 (a) Learner Model Usage (Mayo 2001) (b) Uncertainty Escalation	31
Figure 3.1 A Z Specification for Credit Card Application.....	49
Figure 3.2 Z-EVES environment, a formal method tool.....	51
Figure 3.3 Formalizer Feedback	51
Figure 4.1 Use case Model: Key- Room Case Study.....	63
Figure 4.2 Class Diagram: Key- Room Case Study.....	64
Figure 4.3 Operation Specifications: Key- Room Case Study.....	65
Figure 4.4 Sequence Diagram: Key- Room Case Study.....	65
Figure 4.5 Structure of a Class Schema.....	66
Figure 4.6 Class Schema for Room	68
Figure 4.7 Class Schema for Key	69
Figure 4.8 Value Semantics and Identity of an Object.....	76
Figure 4.9 Formalizing Association: Local	78
Figure 4.10 Formalizing Association: Recursive	79
Figure 4.11 Formalizing Association: Central	80
Figure 4.12 Formalizing Association: Variations	80
Figure 4.13 The Notion of Environment in Object-Z	82
Figure 5.1 Architecture of LOZ.....	96
Figure 5.2 Lesson Organization – An Example	99
Figure 5.3 A Sub-concept and related Mental States	99
Figure 5.4 Mental States- Causal Relations.....	101
Figure 5.5 Overview of the Domain Model in LOZ	104
Figure 5.6 Interface for Marking Confidence.....	107
Figure 5.7 The notion of distance used in CBM test.....	108
Figure 5.8 Fuzzy Membership Functions for Performance (P)	114
Figure 6.1 Lesson Organization (including Mental States).....	124
Figure 6.2 Fuzzy Membership Functions for SMS	132
Figure 6.3 Fuzzy Membership Functions for PAS.....	133
Figure 6.4 Causal Relations for PAS	134
Figure 6.5 Defuzzification, using Larsen’s Product Rule	138
Figure 6.6 Causal Relations for SMS and L.....	140
Figure 6.7 Dynamic Relations	143
Figure 6.8 Modified Causal Relations	146
Figure 6.9 Dynamic Relations for GLA	153
Figure 7.1 JDatastore Explorer Environment.....	164
Figure 7.2 Interface for Editing Table in JDataStore	164
Figure 7.3 Layered Architecture	165
Figure 7.4 Log-in Process: A Happy Day Scenario	166
Figure 7.5 Class Diagram for Domain Model – Concepts.....	167
Figure 7.6 A Sub-concept: Visibility List.....	168
Figure 7.7 Class Diagram for Domain Model – Mental States and Concepts	169
Figure 7.8 A MC Test & Answering Interface.....	169
Figure 7.9 Class Diagram for Learner Model – Learner and GLA.....	170
Figure 7.10 Class Diagram for Learner Model – Learner, SMS and PAS	171
Figure 7.11 Uncertainty Handling – Fuzzy Model.....	172
Figure 7.12 Code for a Fuzzification Process.....	173
Figure 7.13 Code for Fuzzy rule application process.....	173

Figures

Figure 7.14a Sequence Diagram for finding PAS (continues in Figure 7.14b).....	175
Figure 7.14b Sequence Diagram for finding PAS (continued from Fig 7.14a).....	176
Figure 7.15 Designing Associations.....	177
Figure 7.16 An Overall ER Diagram.....	177
Figure 7.17a Relational Tables – Concepts and Learner.....	177
Figure 7.17b Relational Tables – Learners.....	178
Figure 7.18 Offline Table Definitions.....	179
Figure 7.19a Code for the Abstract Parent class of Database Brokers.....	180
Figure 7.19b Code for a Database Broker.....	181
Figure 7.19c Code for the Cache Access.....	180
Figure 7.20a A PERSONA (Sam).....	181
Figure 7.20b A Happy day Scenario – New Returning Learner.....	182
Figure 7.20c A Happy day Scenario –Learner.....	183
Figure 7.21 Screen Navigation.....	184
Figure 7.22 New Learner Screen.....	185
Figure 7.23 Learner Model Estimates.....	186
Figure 7.24 Explaining Learner Model Behaviour.....	186
Figure 8.1 Fuzzy Membership Functions for Performance.....	196
Figure 8.2 Causal Relations for PAS.....	197
Figure 8.3 MC Test Answering Interface – CBM Strategy.....	200
Figure 8.4 Pre Vs Post-test scores- Bar Chart.....	204
Figure 8.5 Pre Vs Post-test scores- Basic Statistics.....	205
Figure 8.6 Power of the Experiment.....	207
Figure 8.7 Statistics for Tests - Ver. 1 Vs Ver.2.....	208
Figure 8.8 Pearson Correlation Co-efficient.....	209
Figure 8.9 Performance Prediction Mechanism (two levels).....	210
Figure 8.10 Performance Prediction Mechanism (three levels).....	211
Figure 9.1 Case Study.....	221
Figure 9.2 Scaffolding Level-I: Support for Syntax.....	224
Figure 9.3 Challenge: Specifying Functional Abstraction.....	226
Figure 9.4 Interactive Help on Semantics.....	226
Figure 9.5 Challenge: Functional Abstraction.....	227
Figure 9.6 Challenge: Data Specification and Functional Abstraction.....	227
Figure 9.7 Challenge: Data & Functional Abstraction.....	228
Figure 9.8 A portion of Domain Model for Phase-II.....	231
Figure 9.9 Dynamic Causal Relations for Phase-II.....	230
Figure 9.10 Causal Relations for Task Sequencing in Phase-II.....	231
Figure 9.11 Predicting Performance on a Testable Concept.....	232
Figure 9.12 Interactive Help on Semantics.....	234
Figure 9.13 Sequence Diagram for the Main Use case.....	242
Figure 9.14 Class Schema Generated for Room.....	243
Figure 9.15 Class Diagram.....	243
Figure 9.16 State Schemas for Room and Key.....	244
Figure 9.17 Operation Partially Completed.....	244
Figure 9.18 Class Schema for Key.....	245
Figure 9.19 Alternate Use case Realization: Sequence Diagram.....	246
Figure 9.20 Operations of an Alternate Schema.....	246

List of Tables

Table 2.1 Feedback Types (based on (Mason <i>et al.</i> 1999).....	27
Table 5.1 Pedagogical Actions for Correct Answers.....	111
Table 5.2 Pedagogical Actions for Incorrect Answers.....	112
Table 5.3 Pedagogical Actions for Medium Answers.....	115
Table 6.1 SMS and Pedagogical Actions.....	127
Table 6.2 Fuzzy Rules for PAS.....	134
Table 6.3 Fuzzy Rules for L (Degree of Learning).....	140
Table 6.4 Fuzzy Rules for the Dynamic Variable SMS.....	143
Table 6.5 Conditional Probability Distribution (SMS&D).....	147
Table 6.6 Conditional Probabilities and Likelihood Ratios (SMS&D).....	150
Table 6.7 Conditional Probabilities (for Dynamic Relation).....	153
Table 6.8 Conditional Probabilities and Likelihoods Ratios (GLA&SMS).....	154
Table 8.1 Fuzzy Rules for Performance.....	197
Table 8.2 Subjective Evaluations – Questionnaire.....	199
Table 8.3 Difference (post-test – Pre-test) Statistics.....	205
Table 8.4 Learner Model Accuracy – Statistics.....	210
Table 8.5 Subjective Evaluation – Response Statistics.....	214
Table 9.1 Fuzzy Rules for Predicting Performance.....	231
Table 9.2 Fuzzy Rules for Task Sequencing in Phase-II.....	231
Tables 9.3 Fuzzy Rules for Interactive Problem Solving Support.....	233
Tables 9.4 Fuzzy Rules for Feedback during Problem Solving.....	233