OCAL S More on threed logics Contents to evication within M

	reface	ge viii x
1	A Classic Theory of Reactive Systems	262
1	Introduction Aims of this book	1
	1.1 What are reactive systems?	2
14	1.2 Process algebras	
2	The language CCS	7
	2.2 000, 101111111	7 16
3	Rehavioural equivalences	31
	3.1 Criteria for good behavioural equivalence	31
	3.2 Trace equivalence: a first attempt	34
	3.3 Strong bisimilarity notteviold 1.1	36
	3.4 Weak bisimilarity	53
	3.5 Game characterization of bisimilarity	65
	3.6 Further results on equivalence checking	
4	Theory of fixed points and bisimulation equivalence	75
	4.1 Posets and complete lattices	75
	4.2 Tarski's fixed point theorem	78
	4.3 Bisimulation as a fixed point with a limited beautiful base beautiful.	85
5	Hennessy–Milner logic 5.1 Introduction to Hennessy–Milner logic 5.2 Hennessy–Milner theorem	89 89

vi			Contents

6	HMI	with recursion	102
	Intro	duction	102
	6.1	Examples of recursive properties	107
	6.2	Syntax and semantics of HML with recursion	109
	6.3	Largest fixed points and invariant properties	113
	6.4	A game characterization for HML with recursion	115
	6.5	Mutually recursive equational systems	120
	6.6	Characteristic properties	125
	6.7	Mixing largest and least fixed points	134
	6.8	Further results on model checking	139
7	Mod	elling mutual exclusion algorithms	142
	Intro	duction	142
	7.1	Specifying mutual exclusion in HML	147
	7.2	Specifying mutual exclusion using CCS itself	149
	7.3	Testing mutual exclusion	152
1		Classic Theory of Reactive Systems	AI
11	AIh	eory of Real-time Systems	159
8	Intro	duction and an area of the second and the second an	159
	8.1	Real-time reactive systems	159
9	CCS	with time delays	161
	9.1	Intuition Association Associat	161
	9.2	Timed labelled transition systems	163
	9.3	Syntax and SOS rules of timed CCS	165
	9.4	Parallel composition	169
	9.5	Other timed process algebras and discussion	173
10	Time	Trace equivalence with a deempt a retrouted at the control of the second and a second at the second	175
	10.1	Motivation ISBN 978-9-521-97848-2 Hardward Withalianiaid gnoul?	175
	10.2	Syntax of timed automata	176
	10.3	Semantics of timed automata	180
	10.4	Networks of timed automata	185
	10.5	More on timed-automata formalisms	190
11	Time	ed behavioural equivalences	193
	11.1	Timed and untimed trace equivalence	193
		Timed and untimed bisimilarity	195
	11.3	Weak timed bisimilarity	200
	11.4	Region graphs	203
		Zones and reachability graphs	214
	11.0	To the state of th	

Introduction 220 12.1 Basic logic 221 12.2 Hennessy–Milner logic with time and regions 229 12.3 Timed bisimilarity versus HML with time 232 12.4 Recursion in HML with time 237 12.5 More on timed logics 246 Introduction 248 Introduction 248 Introduction 248 13.1 Mutual exclusion using timing 250 13.2 Modelling Fischer's algorithm 251 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects 261 A.1 Alternating-bit protocol 261 A.2 Gossiping girls 262 A.3 Implementation of regions 267 References 267 Index 281
12.2 Hennessy–Milner logic with time and regions 12.3 Timed bisimilarity versus HML with time 12.4 Recursion in HML with time 12.5 More on timed logics 13 Modelling and analysis of Fischer's algorithm 13.1 Mutual exclusion using timing 13.2 Modelling Fischer's algorithm 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions References References 267 Index 281
12.3 Timed bisimilarity versus HML with time 12.4 Recursion in HML with time 12.5 More on timed logics 13 Modelling and analysis of Fischer's algorithm 13.1 Mutual exclusion using timing 13.2 Modelling Fischer's algorithm 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions References 267 Index 281
12.4 Recursion in HML with time 12.5 More on timed logics 246 13 Modelling and analysis of Fischer's algorithm 248 Introduction 248 13.1 Mutual exclusion using timing 250 13.2 Modelling Fischer's algorithm 251 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol 261 A.2 Gossiping girls 262 A.3 Implementation of regions 263 References 267 Index 281
12.5 More on timed logics 13 Modelling and analysis of Fischer's algorithm Introduction 248 13.1 Mutual exclusion using timing 13.2 Modelling Fischer's algorithm 251 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions 267 Index 281
Introduction 248 13.1 Mutual exclusion using timing 250 13.2 Modelling Fischer's algorithm 251 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects 261 A.1 Alternating-bit protocol 261 A.2 Gossiping girls 262 A.3 Implementation of regions 263 References 267 Index 248 Introduction 248 248 259 250 261 262 263 References 267 Index 281
Introduction 248 13.1 Mutual exclusion using timing 250 13.2 Modelling Fischer's algorithm 251 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects 261 A.1 Alternating-bit protocol 261 A.2 Gossiping girls 262 A.3 Implementation of regions 263 References 267 Index 281
13.1 Mutual exclusion using timing 13.2 Modelling Fischer's algorithm 251 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions References 267 Index 281
13.2 Modelling Fischer's algorithm 13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions References 267 Index 281
13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions 261 262 A.4 Implementation of regions 263 References 267 Index
13.3 Further exercises on timing-based mutual exclusion algorithms 258 Appendix A Suggestions for student projects A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions 261 262 A.4 Implementation of regions 263 References 267 Index
A.1 Alternating-bit protocol A.2 Gossiping girls A.3 Implementation of regions 261 262 A.3 Implementation of regions 263 References 267 Index
A.2 Gossiping girls A.3 Implementation of regions References Index 262 263
A.3 Implementation of regions 263 References 267 Index 281
References Index 267
References Index 267
Index 281 281 281 281
3.2 A bisimulation showing that $B_0^2 \sim B_0^4 \mid B_0^4$.
6.2 A process. 109
11.2 Partitioning of the valuations for the automaton in Figure 11.1. 205