Contents

Preface xi
Claude JARD and Olivier H. ROUX

Chapter 1. Models for Real-Time Embedded Systems 1
Didier LIME, Olivier H. ROUX and Jiří SRBA

1.1. Introduction 1
1.1.1. Model-checking and control problems 2
1.1.2. Timed models 3
1.2. Notations, languages and timed transition systems 5
1.3. Timed models 8
1.3.1. Timed Automata 8
1.3.2. Time Petri nets 10
1.3.2.1. T-time Petri nets 12
1.3.2.2. Timed-arc petri nets 15
1.3.3. Compared expressiveness of several classes of timed models . . . 19
1.3.3.1. Bisimulation and expressiveness of timed models 19
1.3.3.2. Compared expressiveness of different classes of TPN 20
1.3.3.3. Compared expressiveness of TA, TPN, and TAPN 21
1.4. Models with stopwatches 23
1.4.1. Formal models for scheduling aspects 23
1.4.1.1. Automata and scheduling 23
1.4.1.2. Time Petri nets and scheduling 24
1.4.2. Stopwatch automata 25
1.4.3. Scheduling time Petri nets 26
1.4.4. Decidability results for stopwatch models 31
1.5. Conclusion 31
1.6. Bibliography 31

Chapter 2. Timed Model-Checking 39
Béatrice BÉRARD

2.1. Introduction 39
2.2. Timed models 40
2.2.1. Timed transition system 40
2.2.2. Timed automata 41
2.2.3. Other models 44
2.3. Timed logics 46
2.3.1. Temporal logics CTL and LTL 46
2.3.2. Timed extensions 48
2.3.2.1. Timed CTL 48
2.3.2.2. Timed LTL 50
2.4. Timed model-checking 51
2.4.1. Model-checking LTL and CTL (untimed case) 51
2.4.2. Region automaton 53
2.4.3. Model-checking TCTL 56
2.4.4. Model-checking MTL 58
2.4.5. Efficient model-checking 59
2.4.6. Model-checking in practice 60
2.5. Conclusion 61
2.6. Bibliography 61

Chapter 3. Control of Timed Systems 67
Franck CASSEZ and Nicolas MARKEY

3.1. Introduction 67
3.1.1. Verification of timed systems 67
3.1.2. The controller synthesis problem 68
3.1.3. From control to game 69
3.1.4. Game objectives 70
3.1.5. Varieties of untimed games 71
3.2. Timed games 72
3.2.1. Timed game automata 72
3.2.2. Strategies and course of the game 73
3.2.2.1. The course of a timed game 73
3.2.2.2. Strategies 74
3.3. Computation of winning states and strategies 76
3.3.1. Controllable predecessors 77
3.3.2. Symbolic operators 79
3.3.3. Symbolic computation of winning states 79
3.3.4. Synthesis of winning strategies 80

3.4. Zeno strategies 82
3.5. Implementability 82
3.5.1. Hybrid automata 83
3.5.2. On the existence of non-implementable continuous controllers . . 84
3.5.3. Recent results and open problems 85
3.6. Specification of control objectives 85
3.7. Optimal control 87
3.7.1. TA with costs 87
3.7.2. Optimal cost in timed games 89
3.7.3. Computation of the optimal cost 90
3.7.4. Recent results and open problems 92
3.8. Efficient algorithms for controller synthesis 92
3.8.1. On-the-fly algorithms 93
3.8.2. Recent results and open problems 95
3.9. Partial observation 96
3.10. Changing game rules 97
3.11. Bibliography 98

Chapter 4. Fault Diagnosis of Timed Systems 107
Franck CASSEZ and Stavros TRIPAKIS

4.1. Introduction 107
4.2. Notations 109
4.2.1. Timed words and timed languages 109
4.2.2. Timed automata 110
4.2.3. Region graph of a TA 111
4.2.4. Product of TA 111
4.2.5. Timed automata with faults 112
4.3. Fault diagnosis problems 113
4.3.1. Diagnoser 113
4.3.2. The problems 114
4.3.3. Necessary and sufficient condition for diagnosability 115
4.4. Fault diagnosis for discrete event systems 115
4.4.1. Discrete event systems for fault diagnosis 115
4.4.2. Checking Δ-diagnosability and diagnosability 116
4.4.2.1. Checking Δ-diagnosability 116
4.4.2.2. Checking diagnosability 117
4.4.3. Computation of the maximum delay 120
4.4.4. Synthesis of a diagnoser 121
4.5. Fault diagnosis for timed systems 122
4.5.1. Checking Δ-diagnosability 122
4.5.2. Checking diagnosability 123

4.5.3. Computation of the maximal delay 125
4.5.4. Synthesis of a diagnoser 126
4.5.5. Fault diagnosis with deterministic timed automata 127
4.6. Other results and open problems 136
4.7. Bibliography 136

Chapter 5. Quantitative Verification of Markov Chains 139
Susanna DONATELLI and Serge HADDAD

5.1. Introduction 139
5.2. Performance evaluation of Markov models 140
5.2.1. A stochastic model for discrete events systems 140
5.2.2. Discrete time Markov chains 143
5.2.3. Continuous time Markov chain 146
5.3. Verification of discrete time Markov chain 148
5.3.1. Temporal logics for Markov chains 148
5.3.2. Verification of PCTL formulae 149
5.3.3. Aggregation of Markov chains 151
5.3.4. Verification of PLTL formulae 154
5.3.5. Verification of $PCTL^*$ 157
5.4. Verification of continuous time Markov chain 157
5.4.1. Limitations of standard performance indices 157
5.4.2. A temporal logics for continuous time Markov chains 158
5.4.3. Verification algorithm 159
5.5. State of the art in the quantitative evaluation of Markov chains 160
5.6. Bibliography 162

Chapter 6. Tools for Model-Checking Timed Systems 165
Alexandre DAVID, Gerd BEHRMANN, Peter BULYCHEV, Joakim BYG, Thomas CHATAIN, Kim G. LARSEN, Paul PETTERSSON, Jacob Illum RASMUSSEN, Jiří SRBA, Wang YI, Kenneth Y. JOERGENSEN, Didier LIME, Morgan MAGNIN, Olivier H. ROUX and Louis-Marie TRAONOUEZ

6.1. Introduction 165
6.2. UPPAAL 166
6.2.1. Timed automata and symbolic exploration 166
6.2.1.1. Example 169
6.2.2. Queries 170
6.2.3. Architecture of the tool 172
6.2.4. Reachability pipeline 173
6.2.5. Liveness pipeline 175
6.2.6. Leadsto pipeline 176
6.2.7. Active clock reduction 176

6.2.8. Space reduction techniques 177
6.2.8.1. Avoid storing all states 177
6.2.8.2. Sharing data 178
6.2.8.3. Minimal graph 178
6.2.8.4. Symmetry reduction 179
6.2.9. Approximation techniques 180
6.2.9.1. Over-approximation: convex-hull 180
6.2.9.2. Under-approximation: bit-state hashing 180
6.2.10. Extensions 181
6.2.10.1. Robust reachability 181
6.2.10.2. Merging DBMs 181
6.2.10.3. Stopwatches 181
6.2.10.4. Supremum values 181
6.2.10.5. Other extensions 181
6.3. UPPAAL-CORA 182
6.3.1. Priced timed automata 182
6.3.2. Example 184
6.4. UPPAAL-TIGA 185
6.4.1. Timed game automata 185
6.4.2. Reachability pipeline 187
6.4.3. Time optimality 188
6.4.4. Cooperative strategies 189
6.4.5. Timed games with Büchi objectives 190
6.4.6. Timed games with partial observability 192
6.4.6.1. Algorithm 194
6.4.6.2. Implementation 194
6.4.7. Simulation checking 196
6.4.7.1. Algorithm 197
6.5. TAPAAL 199
6.5.1. Introduction 199
6.5.2. Definition of timed-arc Petri nets used in TAPAAL 200
6.5.3. TAPAAL logic 203
6.5.4. Tool details 204
6.6. ROMÉO: a tool for the analysis of timed extensions of Petri nets 205
6.6.1. Models 206
6.6.1.1. Time Petri nets 206
6.6.1.2. Petri Nets with stopwatches 208
6.6.1.3. Parametric Petri nets with stopwatches 210
6.6.2. Global architecture 210
6.6.3. Systems modeling 211
6.6.4. Verification of properties 211

6.6.4.1. On-line *model checking* 211
6.6.4.2. Off-line *model checking* 213
6.6.5. Using ROMÉO in an example 214
6.7. Bibliography 217

Chapter 7. Tools for the Analysis of Hybrid Models 227
Thao DANG, Goran FREHSE, Antoine GIRARD and Colas LE GUERNIC

7.1. Introduction 227
7.2. Hybrid automata and reachability 228
7.3. Linear hybrid automata 232
7.4. Piecewise affine hybrid systems 234
7.4.1. Time discretization 234
7.4.1.1. Autonomous dynamics 234
7.4.1.2. Dynamics with inputs 236
7.4.2. Scaling up reachability computations 237
7.4.2.1. Reachability using zonotopes 237
7.4.2.2. Efficient implementation for LTI systems 239
7.4.2.3. Dealing with the discrete transitions 239
7.5. Hybridization techniques for reachability computations 241
7.5.1. Approximation with linear hybrid automata 241
7.5.2. Hybridization of nonlinear continuous system 243
7.5.2.1. Properties of the approximate reachable set 244
7.5.2.2. Approximation by hybrid systems with piecewise affine dynamics 245
7.5.3. Hybridization and refinement 246
7.6. Bibliography 249

List of Authors 253

Index 259