

Contents

Part I. Principles and Techniques

Introduction	3
1. Automata	5
1.1 Introductory Examples	5
1.2 A Few Definitions	9
1.3 A Printer Manager	11
1.4 A Few More Variables	12
1.5 Synchronized Product	14
1.6 Synchronization by Message Passing	21
1.7 Synchronization by Shared Variables	24
2. Temporal Logic	27
2.1 The Language of Temporal Logic	28
2.2 The Formal Syntax of Temporal Logic	32
2.3 The Semantics of Temporal Logic	33
2.4 PLTL and CTL: Two Temporal Logics	35
2.5 The Expressivity of CTL*	37
3. Model Checking	39
3.1 Model Checking CTL	39
3.2 Model Checking PLTL	42
3.3 The State Explosion Problem	45
4. Symbolic Model Checking	47
4.1 Symbolic Computation of State Sets	47
4.2 Binary Decision Diagrams (BDD)	51
4.3 Representing Automata by BDDs	54
4.4 BDD-based Model Checking	56
5. Timed Automata	59
5.1 Description of a Timed Automaton	60
5.2 Networks of Timed Automata and Synchronization	62

5.3	Variants and Extensions of the Basic Model.....	64
5.4	Timed Temporal Logic	67
5.5	Timed Model Checking.....	68
	Conclusion	73

Part II. Specifying with Temporal Logic

	Introduction	77
6.	Reachability Properties	79
6.1	Reachability in Temporal Logic	79
6.2	Model Checkers and Reachability	80
6.3	Computation of the Reachability Graph	80
7.	Safety Properties	83
7.1	Safety Properties in Temporal Logic	83
7.2	A Formal Definition	84
7.3	Safety Properties in Practice	86
7.4	The History Variables Method	87
8.	Liveness Properties	91
8.1	Simple Liveness in Temporal Logic	92
8.2	Are Liveness Properties Useful?	92
8.3	Liveness in the Model, Liveness in the Properties	94
8.4	Verification under Liveness Hypotheses	96
8.5	Bounded Liveness	97
9.	Deadlock-freeness	99
9.1	Safety? Liveness?	99
9.2	Deadlock-freeness for a Given Automaton	99
9.3	Beware of Abstractions!	101
10.	Fairness Properties	103
10.1	Fairness in Temporal Logic	103
10.2	Fairness and Nondeterminism	104
10.3	Fairness Properties and Fairness Hypotheses	104
10.4	Strong Fairness and Weak Fairness	106
10.5	Fairness in the Model or in the Property?	107
11.	Abstraction Methods	109
11.1	When Is Model Abstraction Required?	110
11.2	Abstraction by State Merging	110
11.3	What Can Be Proved in the Abstract Automaton?	110
11.4	Abstraction on the Variables	114

11.5 Abstraction by Restriction	118
11.6 Observer Automata	120
Conclusion	125

Part III. Some Tools

Introduction	129
12. SMV – Symbolic Model Checking	131
12.1 What Can We Do with SMV?	131
12.2 SMV’s Essentials	131
12.3 Describing Automata	132
12.4 Verification	135
12.5 Synchronizing Automata	136
12.6 Documentation and Case Studies	137
SMV Bibliography	138
13. SPIN – Communicating Automata	139
13.1 What Can We Do with SPIN?	139
13.2 SPIN’s Essentials	139
13.3 Describing Processes	140
13.4 Simulating the System	141
13.5 Verification	142
13.6 Documentation and Case Studies	144
SPIN Bibliography	144
14. DESIGN/CPN – Coloured Petri Nets	145
14.1 What Can We Do with DESIGN/CPN?	145
14.2 DESIGN/CPN’s Essentials	145
14.3 Editing with DESIGN/CPN	146
14.4 Simulating the Net	147
14.5 Analyzing the Net	149
14.6 Documentation and Case Studies	149
DESIGN/CPN Bibliography	150
15. UPPAAL – Timed Systems	153
15.1 What Can We Do with UPPAAL?	153
15.2 UPPAAL’s Essentials	153
15.3 Modeling Timed Systems with UPPAAL	154
15.4 Simulating a System	157
15.5 Verification	157
15.6 Documentation and Case Studies	158
UPPAAL Bibliography	158

16. KRONOS – Model Checking of Real-time Systems	161
16.1 What Can We Do with KRONOS?	161
16.2 KRONOS' Essentials	161
16.3 Describing Automata	162
16.4 Synchronized Product	164
16.5 Model Checking	165
16.6 Documentation and Case Studies	167
KRONOS Bibliography	167
17. HYTECH – Linear Hybrid Systems	169
17.1 What Can We Do With HYTECH?	169
17.2 HYTECH's Essentials	169
17.3 Describing Automata	170
17.4 System Analysis	172
17.5 Parametric Analysis	174
17.6 Documentation and Case Studies	176
HYTECH Bibliography	176
Main Bibliography	179
Index	183