

TABLES AND FIGURES

*“An undetected error [...] is like a sunken rock at sea yet undiscovered, upon which it is impossible to say what wrecks may have taken place.”
(Sir John Herschel, 1842)*

TABLES

3.1	Basic Data Types	41
3.2	Operator Precedence	54
6.1	Frequently Used LTL Formulae	137
6.2	Formalization of Properties	138
10.1	Example of Type Abstraction	237
16.1	Index of All Manual Pages	370
16.2	Typical Data Ranges	404

FIGURES

1.1	Circular Blocking	3
1.2	Deadly Embrace	4
2.1	Simple Producer Consumer Example	11
2.2	Revised Producer Consumer Example	17
2.3	Dekker’s Mutual Exclusion Algorithm (1962)	20
2.4	Faulty Mutual Exclusion Algorithm	23
2.5	Peterson’s Mutual Exclusion Algorithm (1981)	26
2.6	Data Transfer Protocol	27
3.1	Simple Model of a Telephone System	63

3.2	Word Count Program Using STDIN Feature	69
3.3	Using Channel Assertions	70
4.1	Labeling End States	77
4.2	Control Flow Structure for LTL Property $\neg \square (p \rightarrow (p \cup q))$	88
5.1	Counter Example	106
5.2	A Sink, a Source, and a Filter Process	107
5.3	Small Model for the Pathfinder Problem	109
5.4	Reachability Graph for Pathfinder Problem	111
5.5	Disk Scheduler Context	114
5.6	Minimal Device Driver Interface Model	115
5.7	Disk Scheduler Model	116
5.8	Number of Possible States for q Message Buffers	121
6.1	A Simple Finite State Automaton	129
6.2	A Possible Interpretation of the Automaton in 6.1	130
6.3	Model of a Simple Computation	133
6.4	Automaton for $\diamond \square p$	143
6.5	Automaton for $\neg \diamond \square p = \square \diamond \neg p$	144
6.6	Automaton for $\square (p \rightarrow \diamond q)$	146
6.7	Never Automaton for $\diamond (p \wedge \square !q)$	146
6.8	Automaton for $\square (p \rightarrow (r \cup q))$	147
6.9	Automaton for $\diamond (p \wedge !(r \cup q))$	147
7.1	Sample PROMELA Model	154
7.2	Transition Relation for the Model in Figure 7.1	155
7.3	PROMELA Semantics Engine	159
7.4	Specification of Procedure <code>executable()</code>	161
7.5	State Space Structure for First and Third Example	163
7.6	State Space Structure for Second Example	164
7.7	Claim Stutter	165
8.1	Basic Depth-First Search Algorithm	168
8.2	Extension of Figure 8.1 for Checking Safety Properties	170
8.3	Example for Depth-Limited Search	172
8.4	Depth-Limited Search	173
8.5	Stateless Search	176
8.6	Breadth-First Search Algorithm	177
8.7	Nested Depth-First Search for Checking Liveness Properties	180
8.8	$(k+2)$ Times Unfolded State Space for Weak Fairness	183
8.9	A Two-State Global State Space Example	184
8.10	Unfolded State Space for Example in Figure 8.9	185
9.1	The Finite State Automata T1 and T2	192
9.2	Expanded Asynchronous Product of T1 and T2	193
9.3	Effect of Partial Order Reduction	195
9.4	State Components for COLLAPSE Compression	199
9.5	Minimized Automaton Structure After Storing $\{000, 001, 101\}$	202

9.6	New Automaton Structure Storing {000, 001, 101, 100}	203
9.7	Standard Hash Table Lookup	207
9.8	Optimal Nr. of Hash Functions and Probability of Hash Collision	211
9.9	Measured Coverage of Double Bitstate Hashing (k=2)	212
10.1	The Complete Parse Tree for fahr.c	219
10.2	Part of Parse Tree Structure for the While Loop	220
10.3	Control-Flow Graph for While Construct from 10.2	221
10.4	Generated SPIN model	223
10.5	Word Count Model	233
10.6	Abstracted Word Count Model	234
10.7	Simplified Model	235
11.1	The Structure of SPIN	246
12.1	XSPIN Main Window	268
12.2	The Simulations Options Panel	273
12.3	Message Sequence Chart Display (portion)	274
12.4	Main Simulation Output Panel	275
12.5	Basic Verification Options	276
12.6	Advanced Verification Options	277
12.7	The LTL Property Manager	279
12.8	The Automata View	281
13.1	Simple Example of a Timeline Specification	284
13.2	Büchi Automaton for the Timeline in 13.1	288
13.3	Never Claim for the Timeline in 13.1	288
13.4	Variation on the Timeline from 13.1	289
13.5	Büchi Automaton for the Timeline in 13.5	290
13.6	Timeline and Automaton for a Single Required Event	291
13.7	Timeline and Automaton for a Single Fail Event	292
13.8	Timeline and Automaton for a Regular and a Required Event	293
13.9	A More Complex Timeline Specification	293
13.10	Büchi Automaton for the Timeline in 13.10	294
13.11	Timeline Specification with Three Events	294
13.12	Büchi Automaton for the Timeline in 13.12	295
13.13	Attempt to Express the LTL property $!(a \text{ U } b)$	296
13.14	The Correct Büchi Automaton for LTL property $!(a \text{ U } b)$	296
13.15	A Variant of the Timeline in Figure 13.13	296
14.1	Typical Scenario for a POTS Call	302
14.2	Initial Behavior Model of a POTS Subscriber	303
14.3	Two Alternative Subscriber Models	304
14.4	Two-State Model of Subscriber	305
14.5	Simple Switch Model for Outgoing Calls	306
14.6	SS7 Scenario for Call Setup	308
14.7	Extended Local Switch Model	310
14.8	POTS Interface Model for a Remote Switch	312

14.9	PROMELA Model of Visible Behavior of Remote Switch	313
14.10	Switch Session Management Structure	314
14.11	New Model for the Session Handler	317
14.12	New Model for the Remote Switch	319
14.13	New Version of the Session Manager	320
14.14	Revised Subscriber Process	321
14.15	Never Claim to Trigger Three-Way Calling Scenario	322
15.1	The Sieve of Eratosthenes	326
15.2	Alternative Structure for Sieve	331
15.3	Sleep-Wakeup Routines	336
15.4	Remainder of Verification Model for UTS	337
15.5	Agent and Server Processes	342
15.6	The Client Processes	343
15.7	Square Root Server Model	354
15.8	A Sample Automaton to Check C-Style Comment Conventions	359
15.9	A Simple C-Style Comment Filter	360
17.1	Example of Embedded C Code	497
17.2	Replacing <code>c_state</code> with <code>c_track</code> Primitives	499
19.1	Example Output Generated by PAN	541
A.1	Example PROMELA Model	557
A.2	Finite State Automata A_1 , A_2 , and B	558
A.3	Asynchronous Product of A_1 and A_2	559
A.4	Expanded Asynchronous Product for Initial Value $x = 4$	559
A.5	(Expanded) Synchronous Product of A.4 and Automaton B	560
C.1	Petri Net with Hang State	577