
Contents

Part I Introduction

1	Background on Concurrency Theory	3
1.1	Concurrency Is Everywhere	3
1.2	Characteristics of Concurrent Systems	4
1.3	Classes of Concurrent Systems	6
1.3.1	Basic Event Ordering	6
1.3.2	Timing Axis	7
1.3.3	Probabilistic Choice Axis	8
1.3.4	Mobility Axis	9
1.4	Mathematical Theories	9
1.5	Overview of Book	13

Part II Concurrency Theory – Untimed Models

2	Process Calculi: LOTOS	19
2.1	Introduction	19
2.2	Example Specifications	20
2.2.1	A Communication Protocol	20
2.2.2	The Dining Philosophers	22
2.3	Primitive Basic LOTOS	22
2.3.1	Abstract Actions	26
2.3.2	Action Prefix	28
2.3.3	Choice	29
2.3.4	Nondeterminism	30
2.3.5	Process Definition	34
2.3.6	Concurrency	41
2.3.7	Sequential Composition and Exit	47
2.3.8	Syntax of pbLOTOS	50
2.4	Example	52

3	Basic Interleaved Semantic Models	55
3.1	A General Perspective on Semantics	55
3.1.1	Why Semantics?	55
3.1.2	Formal Definition	57
3.1.3	Modelling Recursion	61
3.1.4	What Makes a Good Semantics?	63
3.2	Trace Semantics	63
3.2.1	The Basic Approach	63
3.2.2	Formal Semantics	66
3.2.3	Development Relations	73
3.2.4	Discussion	75
3.3	Labelled Transition Systems	76
3.3.1	The Basic Approach	76
3.3.2	Formal Semantics	78
3.3.3	Development Relations	85
3.4	Verification Tools	101
3.4.1	Overview of CADP	102
3.4.2	Bisimulation Checking in CADP	103
4	True Concurrency Models: Event Structures	105
4.1	Introduction	105
4.2	The Basic Approach – Event Structures	107
4.3	Event Structures and pbLOTOS	112
4.4	An Event Structures Semantics for pbLOTOS	115
4.5	Relating Event Structures to Labelled Transition Systems	123
4.6	Development Relations	126
4.7	Alternative Event Structure Models	134
4.8	Summary and Discussion	138
5	Testing Theory and the Linear Time – Branching Time Spectrum	141
5.1	Trace-refusals Semantics	141
5.1.1	Introduction	141
5.1.2	The Basic Approach	143
5.1.3	Deriving Trace-refusal Pairs	145
5.1.4	Internal Behaviour	146
5.1.5	Development Relations: Equivalences	152
5.1.6	Nonequivalence Development Relations	154
5.1.7	Explorations of Congruence	158
5.1.8	Summary and Discussion	159
5.2	Testing Justification for Trace-refusals Semantics	160
5.3	Testing Theory in General and the Linear Time – Branching Time Spectrum	161
5.3.1	Sequence-based Testing	162
5.3.2	Tree-based Testing	163

5.4	Applications of Trace-refusals Relations in Distributed Systems	166
5.4.1	Relating OO Concepts to LOTOS	166
5.4.2	Behavioural Subtyping	167
5.4.3	Viewpoints and Consistency	177

Part III Concurrency Theory – Further Untimed Notations

6	Beyond pbLOTOS	185
6.1	Basic LOTOS	185
6.1.1	Disabling	185
6.1.2	Generalised Choice	188
6.1.3	Generalised Parallelism	189
6.1.4	Verbose Specification Syntax	190
6.1.5	Verbose Process Syntax	190
6.1.6	Syntax of bLOTOS	191
6.2	Full LOTOS	192
6.2.1	Guarded Choice	193
6.2.2	Specification Notation	193
6.2.3	Process Definition and Invocation	194
6.2.4	Value Passing Actions	194
6.2.5	Local Definitions	202
6.2.6	Selection Predicates	202
6.2.7	Generalised Choice	203
6.2.8	Parameterised Enabling	204
6.2.9	Syntax of fLOTOS	206
6.2.10	Comments	206
6.3	Examples	207
6.3.1	Communication Protocol	207
6.3.2	Dining Philosophers	210
6.4	Extended LOTOS	213
7	Comparison of LOTOS with CCS and CSP	215
7.1	CCS and LOTOS	217
7.1.1	Parallel Composition and Complementation of Actions	217
7.1.2	Restriction and Hiding	220
7.1.3	Internal Behaviour	221
7.1.4	Minor Differences	221
7.2	CSP and LOTOS	222
7.2.1	Alphabets	222
7.2.2	Internal Actions	224
7.2.3	Choice	225
7.2.4	Parallelism	227
7.2.5	Hiding	227

7.2.6	Comparison of LOTOS Trace-refusals with CSP Failures-divergences.....	228
8	Communicating Automata	233
8.1	Introduction	233
8.2	Networks of Communicating Automata.....	234
8.2.1	Component Automata	234
8.2.2	Parallel Composition.....	236
8.2.3	Example Specifications.....	239
8.2.4	Semantics and Development Relations	240
8.2.5	Verification of Networks of Communicating Automata..	241
8.2.6	Relationship to Process Calculi.....	246
8.3	Infinite State Communicating Automata	250
8.3.1	Networks of Infinite State Communicating Automata ..	251
8.3.2	Semantics of ISCA's as Labelled Transition Systems...	254
Part IV Concurrency Theory – Timed Models		
9	Timed Process Calculi, a LOTOS Perspective	261
9.1	Introduction	261
9.2	Timed LOTOS – The Issues	262
9.2.1	Timed Action Enabling	262
9.2.2	Urgency	267
9.2.3	Persistency	270
9.2.4	Nondeterminism	271
9.2.5	Synchronisation	272
9.2.6	Timing Domains	273
9.2.7	Time Measurement	273
9.2.8	Timing of Nonadjacent Actions	274
9.2.9	Timed Interaction Policies	275
9.2.10	Forms of Internal Urgency	276
9.2.11	Discussion	278
9.3	Timed LOTOS Notation	278
9.3.1	The Language	278
9.3.2	Example Specifications.....	281
9.4	Timing Anomalies in tLOTOS	283
9.5	E-LOTOS, the Timing Extensions	285
10	Semantic Models for tLOTOS	287
10.1	Branching Time Semantics	287
10.1.1	Timed Transition Systems	287
10.1.2	Operational Semantics	289
10.1.3	Branching Time Development Relations	299
10.2	True Concurrency Semantics	304

10.2.1	Introduction	304
10.2.2	Timed Bundle Event Structures	305
10.2.3	Causal Semantics for tLOTOS	308
10.2.4	Anomalous Behaviour	318
10.2.5	Discussion	320
11	Timed Communicating Automata	321
11.1	Introduction	321
11.2	Timed Automata – Formal Definitions	323
11.2.1	Syntax	324
11.2.2	Semantics	325
11.3	Real-time Model-checking	332
11.3.1	Forward Reachability	333
11.3.2	Example: Reachability Analysis on the Multimedia Stream	341
11.3.3	Issues in Real-time Model-checking	342
12	Timelocks in Timed Automata	347
12.1	Introduction	347
12.2	A Classification of Deadlocks in Timed Automata	349
12.2.1	Discussion: Justifying the Classification of Deadlocks	350
12.2.2	Discussion: Timelocks in Process Calculi	351
12.3	Time-actionlocks	352
12.3.1	Timed Automata with Deadlines	353
12.3.2	Example: A TAD Specification for the Multimedia Stream	358
12.4	Zeno-timelocks	359
12.4.1	Example: Zeno-timelocks in the Multimedia Stream	359
12.4.2	Nonzenoness: Syntactic Conditions	361
12.4.3	Nonzenoness: A Sufficient-and-Necessary Condition	368
12.5	Timelock Detection in Real-time Model-checkers	374
12.5.1	Uppaal	374
12.5.2	Kronos	376
13	Discrete Timed Automata	377
13.1	Infinite vs. Finite States	377
13.2	Preliminaries	380
13.2.1	Fair Transition Systems and Invariance Proofs	381
13.2.2	The Weak Monadic Second-order Theory of 1 Successor (WS1S) and MONA	383
13.3	Discrete Timed Automata – Formal definitions	384
13.3.1	Syntax	384
13.3.2	Example: A DTA Specification for the Multimedia Stream	386
13.3.3	Semantics	387

XXIV Contents

13.4 Verifying Safety Properties over DTAs	389
13.5 Discussion: Comparing DTAs and TIOAs with Urgency	394
References	397
Appendix	409
14.1 Enabling as a Derived Operator	409
14.2 Strong Bisimulation Is a Congruence	409
14.3 Weak Bisimulation Congruence	414
14.4 Timed Enabling as a Derived Operator	419
14.5 Hiding is Not Substitutive for Timed Bisimulations	420
14.6 Substitutivity of Timed Strong Bisimulation	420
14.7 Substitutivity of Timed Rooted Weak Bisimulation	422
Index	429