
Contents

Symbols	xiii
1 Introduction	1
1.1 Contribution and Background	1
1.2 Outline of the Book	4
2 An Introduction to Petri Nets	7
2.1 Automata	7
2.2 Petri Nets	9
2.2.1 Ordinary Petri Nets	9
2.2.2 Generalized Petri Nets	12
2.2.3 Additional Concepts	16
2.2.4 Special Structures	18
2.2.5 Concurrency	19
2.3 Petri Nets versus Automata	21
2.3.1 Model Conversions	21
2.3.2 Comparison	23
2.4 Bibliographical Notes	28
3 The Supervision of Petri Nets	29
3.1 Introduction	29
3.2 Supervision Based on Place Invariants	30
3.2.1 Fully Controllable and Observable Petri Nets	30
3.2.2 Petri Nets with Uncontrollable and Unobservable Transitions	31
3.3 Supervision Settings	33
3.4 Admissible and Feasible Sets of Constraints	37
3.5 Transformations to Admissible Constraints	39
3.6 Overview of Supervisory Methods	42
3.7 Concluding Remarks	50

4	Enforcing General Specifications	51
4.1	Introduction	51
4.2	Enforcing Generalized Linear Constraints	52
4.2.1	Introduction	52
4.2.2	On the Significance of the Constraints	54
4.2.3	Design in the Fully Controllable and Observable Case	57
4.2.4	Admissibility and Feasibility	59
4.2.5	Design in the Partially Controllable and Observable Case	62
4.2.6	Example	77
4.2.7	The Structural Approach	80
4.3	Language Constraints	84
4.4	Disjunctions of Constraints	86
5	Decentralized Supervision of Petri Nets	93
5.1	Introduction	93
5.2	Related Work	96
5.3	Preliminaries	100
5.4	The Model	101
5.5	Decentralized Admissibility	103
5.5.1	Definition and Application	103
5.5.2	Significance of D-Admissibility	109
5.6	Distributing a Centralized Supervisory Policy	112
5.7	Design with Constraint Transformations	116
5.7.1	Supervision without Communication	116
5.7.2	Supervision with Communication	118
5.7.3	Liveness Constraints	120
5.8	Example	120
6	Deadlock and Liveness Properties of Petri Nets	125
6.1	Introduction	125
6.2	Preliminaries	127
6.3	Results	129
6.3.1	Deadlock Prevention and Liveness Enforcement	129
6.3.2	Deadlock and (\mathcal{T}) -Liveness Characterization Based on Active Subnets	135
6.4	Implications and Discussion	142
6.4.1	Deadlock Prevention	142
6.4.2	Least Restrictive Deadlock Prevention	144
6.4.3	\mathcal{T} -liveness Enforcement	145
6.5	Algorithms	146
6.5.1	The Computation of Active Subnets	146
6.5.2	Transformation of Petri Nets to PT -ordinary Petri Nets	147
6.5.3	Transformation of Petri Nets to EAC Nets	148

7 Liveness Enforcement in Petri Nets: A Structural Approach. Part I	153
7.1 Introduction	153
7.2 Related Work	155
7.3 Problem Statement	158
7.3.1 Deadlock Prevention	158
7.3.2 \mathcal{T} -liveness Enforcement	159
7.4 Motivation	159
7.4.1 The Role of Linear Marking Inequalities	160
7.4.2 The Role of Iterations	161
7.4.3 The Need for Net Transformations	162
7.5 Procedure Definition	163
7.5.1 Definition	163
7.5.2 Siphons Not Needing Control	166
7.5.3 Generating the Sets of Inequalities (L, b) and (L_0, b_0)	167
7.5.4 Petri Net Transformations	168
7.5.5 The Effect of Net Transformation on Marking Constraints	170
7.5.6 The Computation of a \mathcal{T} -minimal Active Subnet	172
7.6 Examples	172
7.7 Properties	176
7.7.1 Preliminaries	176
7.7.2 Proof of Correctness	178
7.7.3 Permissiveness Properties	180
7.8 Extending the Permissiveness of the Procedure	183
8 Liveness Enforcement in Petri Nets: A Structural Approach. Part II	187
8.1 Introduction	187
8.2 Problem Statement	187
8.2.1 Deadlock Prevention	187
8.2.2 \mathcal{T} -liveness Enforcement	188
8.3 Motivation	189
8.3.1 Partially Controllable and Observable Petri Nets	189
8.3.2 Constraint Transformations and Deadlock	190
8.3.3 The Set \mathcal{T}'	191
8.3.4 The Use of Initial Constraints	192
8.4 Procedure Definition	193
8.4.1 Description	193
8.4.2 Definition	194
8.4.3 Transforming Constraints to Admissible Constraints	197
8.4.4 The Computation of the Active Subnet	200
8.5 Examples	201
8.6 Properties	205
8.6.1 Preliminaries	206

8.6.2 Proof of Correctness	206
8.6.3 Permissiveness	207
8.7 Extending Permissiveness	210
8.8 Convergence Issues	213
8.8.1 Termination Issues	213
8.8.2 Computational Complexity	216
8.9 Applications	217
8.9.1 Deadlock Prevention in a Manufacturing System	217
8.9.2 Minimization of the Number of Resources	220
8.9.3 Resource Preallocation	221
9 DES Control of Concurrent Hybrid Systems	225
9.1 Introduction	225
9.2 Related Work	228
9.3 Modeling	229
9.3.1 The DES Representation of Hybrid Systems	229
9.3.2 Coupling among Hybrid Systems	230
9.3.3 Transition Types	230
9.3.4 Modeling Nondeterminism	232
9.3.5 Self-Loops	232
9.3.6 Synchronization	233
9.3.7 Modeling Example	233
9.3.8 Supervision Example	236
9.4 DES-Level Supervision	236
9.4.1 Case 1: All Places Have Controllable or FNI Self-Loops	239
9.4.2 Case 2: Not All Places Have Controllable or FNI Self-Loops	239
9.4.3 Decentralized Control	242
9.4.4 Dealing with NFNI Transitions	242
10 Hybrid System Level Control	245
10.1 Introduction	245
10.2 Related Work	246
10.3 The Hybrid Automaton Model	248
10.4 Extracting the DES Abstraction	250
10.5 Computation of the Controllable Invariant Sets	256
10.5.1 The Controlled Invariance Context	256
10.5.2 The Idea of the Approach	257
10.5.3 The Computation	257
10.6 Computation of the Predecessor	261
References	265
Index	279