

---

# Contents

---

<b>PREFACE</b> . . . . .	ix
<b>ACKNOWLEDGMENTS</b> . . . . .	xi
<b>INTRODUCTION</b> . . . . .	xiii
I.1. Motivation . . . . .	xiv
I.1.1. An example of asynchronous circuit . . . . .	xiv
I.2. The good parameters problem . . . . .	xv
I.3. Content and organization of the book . . . . .	xvi
I.3.1. Content . . . . .	xvi
I.3.2. Organization of the book . . . . .	xvii
I.3.3. Acknowledgments . . . . .	xviii
<b>CHAPTER 1. PARAMETRIC TIMED AUTOMATA</b> . . . . .	1
1.1. Constraints on clocks and parameters . . . . .	1
1.1.1. Clocks . . . . .	1
1.1.2. Parameters . . . . .	2
1.1.3. Constraints . . . . .	2
1.2. Labeled transition systems . . . . .	4
1.3. Timed automata . . . . .	4
1.3.1. Syntax . . . . .	5
1.3.2. Semantics . . . . .	7
1.4. Parametric timed automata . . . . .	10
1.4.1. Syntax . . . . .	11
1.4.2. Semantics . . . . .	14
1.5. Related work . . . . .	19
1.5.1. Representation of time . . . . .	19
1.5.2. Timed automata . . . . .	20

1.5.3. Time Petri nets . . . . .	21
1.5.4. Hybrid systems . . . . .	22
<b>CHAPTER 2. THE INVERSE METHOD FOR PARAMETRIC TIMED AUTOMATA . . . . .</b>	<b>23</b>
2.1. The inverse problem . . . . .	24
2.1.1. A motivating example . . . . .	24
2.1.2. The problem . . . . .	26
2.2. The inverse method algorithm . . . . .	27
2.2.1. Principle . . . . .	27
2.2.2. A toy example . . . . .	28
2.2.3. Remarks on the algorithm . . . . .	28
2.2.4. Results . . . . .	32
2.2.5. Discussion . . . . .	40
2.3. Variants of the inverse method . . . . .	40
2.3.1. Algorithm with state inclusion in the fixpoint . . . . .	41
2.3.2. Algorithm with union of the constraints . . . . .	42
2.3.3. Algorithm with simple return . . . . .	44
2.3.4. Combination: inclusion in fixpoint and union . . . . .	45
2.3.5. Combination: inclusion in fixpoint and direct return . . . . .	46
2.3.6. Summary of the algorithms . . . . .	46
2.4. Related work . . . . .	49
2.4.1. History of the inverse method . . . . .	49
2.4.2. Time-abstract bisimulation . . . . .	50
2.4.3. Formal techniques of verification . . . . .	50
2.4.4. Problems related to the inverse problem . . . . .	51
2.4.5. Parameter synthesis for parametric timed automata . . . . .	53
<b>CHAPTER 3. THE INVERSE METHOD IN PRACTICE: APPLICATION TO CASE STUDIES . . . . .</b>	<b>55</b>
3.1. IMITATOR . . . . .	56
3.1.1. History . . . . .	56
3.1.2. Architecture and features . . . . .	56
3.2. Flip-flop . . . . .	57
3.3. SR-Latch . . . . .	58
3.3.1. Parameter synthesis . . . . .	59
3.4. AND-OR . . . . .	60
3.5. IEEE 1394 Root Contention Protocol . . . . .	62
3.5.1. Description of the model . . . . .	62
3.5.2. Synthesis of constraints . . . . .	64
3.6. Bounded Retransmission Protocol . . . . .	64
3.7. CSMA/CD protocol . . . . .	65
3.8. The SPSMALL memory . . . . .	67

---

3.8.1. Description . . . . .	67
3.8.2. A short history . . . . .	71
3.8.3. Manually abstracted model . . . . .	72
3.8.4. Automatically generated model . . . . .	75
3.9. Networked automation system . . . . .	77
3.9.1. Description of the model . . . . .	77
3.9.2. Definition of a zone of good behavior . . . . .	78
3.9.3. Comparison with other methods . . . . .	79
3.10. Tools related to IMITATOR . . . . .	79
<b>CHAPTER 4. BEHAVIORAL CARTOGRAPHY OF TIMED AUTOMATA . . .</b>	<b>81</b>
4.1. The behavioral cartography algorithm . . . . .	82
4.2. Properties . . . . .	83
4.2.1. Acyclic parametric timed automata . . . . .	83
4.2.2. General case . . . . .	84
4.3. Case studies . . . . .	84
4.3.1. Implementation . . . . .	85
4.3.2. SR latch . . . . .	86
4.3.3. Flip-flop . . . . .	91
4.3.4. The root contention protocol . . . . .	95
4.3.5. SPSMALL memory . . . . .	95
4.4. Related work . . . . .	101
<b>CHAPTER 5. PARAMETER SYNTHESIS FOR HYBRID AUTOMATA . . . . .</b>	<b>103</b>
5.1. Hybrid automata with parameters . . . . .	105
5.1.1. Basic definitions . . . . .	105
5.1.2. Symbolic semantics of linear hybrid automata . . . . .	108
5.2. Algorithms for hybrid automata . . . . .	109
5.2.1. The inverse method for hybrid automata . . . . .	109
5.2.2. Behavioral cartography of hybrid automata . . . . .	111
5.2.3. Enhancement of the method for affine dynamics . . . . .	114
5.3. Implementation . . . . .	116
5.4. Discussion . . . . .	117
5.5. Related work . . . . .	118
<b>CHAPTER 6. APPLICATION TO THE ROBUSTNESS ANALYSIS OF SCHEDULING PROBLEMS . . . . .</b>	<b>121</b>
6.1. Preliminaries . . . . .	121
6.1.1. Scheduling problems . . . . .	121
6.1.2. Timed automata augmented with stopwatches . . . . .	122
6.2. Scheduling analysis using the inverse method . . . . .	123
6.2.1. Modeling schedulability with timed automata . . . . .	123

6.2.2. Robustness analysis using the inverse method . . . . .	124
6.2.3. Schedulability zone synthesis . . . . .	124
6.3. Application to scheduling problems . . . . .	126
6.3.1. Jobs with deadlines . . . . .	126
6.3.2. Schedulability zone synthesis . . . . .	126
6.3.3. Next generation spacecraft flight control system . . . . .	127
6.4. Discussion . . . . .	130
6.5. Related work . . . . .	131
<b>CHAPTER 7. CONCLUSION AND PERSPECTIVES . . . . .</b>	<b>133</b>
7.1. Trace-based inverse method and partial orders . . . . .	134
7.2. Preservation of temporal logics . . . . .	134
7.3. Application to other formalisms . . . . .	135
<b>BIBLIOGRAPHY . . . . .</b>	<b>137</b>
<b>INDEX . . . . .</b>	<b>149</b>