

Synthesizing Multi-View Models of Software Systems

Bernard Lambeau
bernard.lambeau@uclouvain.be

April 5, 2011

Abstract

Acknowledgements

I would like to thank ...

Contents

Abstract	i
Acknowledgements	ii
1 Introduction	2
2 A multi-view modeling framework	3
2.1 Event-based behavior models	3
2.1.1 Message Sequence Charts for instance behaviors . . .	3
2.1.2 Labelled Transition Systems for class behaviors . . .	3
2.2 State-based abstractions	3
2.2.1 Capturing state information with Fluents	3
2.2.2 Guards in behavior models	3
2.2.3 Decorations on behavior models	3
2.3 Intentional models as goal graphs on fluents	3
2.3.1 Goals and Fluent Linear Temporal Logic (FLTL) . . .	3
2.3.2 Linking FLTL and LTS: property and tester automata	3
3 LTS trace semantics of guarded hMSC	4
3.1 From guarded hMSC to guarded LTS	4
3.2 From guarded LTS to pure LTS	4
3.3 Model analysis perspectives of deductive synthesis	4
4 Inductive synthesis of LTS models from MSC and hMSC models	5
4.1 From grammar induction to model synthesis	5
4.2 Interactive induction of LTS models from MSC	5
4.2.1 Pruning the induction space with state information . .	5
4.2.2 Pruning the induction space with goals	5
4.3 Induction of LTS models from hMSC	5
5 Evaluation	6
5.1 Experimental results on case studies	6
5.2 Experimental results on synthetic data	6

6	Towards an evaluation platform of inductive model synthesis	7
6.1	Motivation and overview	7
6.2	Scientific setup	7
6.2.1	State machines	7
6.2.2	Training and test samples	7
6.2.3	Submission and scoring	7
6.2.4	Baseline: Blue-Fringe performance	7
6.3	Competition results	7
7	Tool Support	8
7.1	A FLTL Model-checker for g-hMSC Models	8
7.2	Interactive State-Machine Induction from Scenarios	8
7.3	The Gisele Clinical Pathway Analyzer	8
8	Discussion and Related Work	9
9	Conclusion	10
	Bibliography	10

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 2

A multi-view modeling framework

2.1 Event-based behavior models

2.1.1 Message Sequence Charts for instance behaviors

2.1.2 Labelled Transition Systems for class behaviors

2.2 State-based abstractions

2.2.1 Capturing state information with Fluents

2.2.2 Guards in behavior models

2.2.3 Decorations on behavior models

2.3 Intentional models as goal graphs on fluents

2.3.1 Goals and Fluent Linear Temporal Logic (FLTL)

2.3.2 Linking FLTL and LTS: property and tester automata

Chapter 3

LTS trace semantics of guarded hMSC

- 3.1 From guarded hMSC to guarded LTS
- 3.2 From guarded LTS to pure LTS
- 3.3 Model analysis perspectives of deductive synthesis

Chapter 4

Inductive synthesis of LTS models from MSC and hMSC models

- 4.1 From grammar induction to model synthesis
- 4.2 Interactive induction of LTS models from MSC
 - 4.2.1 Pruning the induction space with state information
 - 4.2.2 Pruning the induction space with goals
- 4.3 Induction of LTS models from hMSC

Chapter 5

Evaluation

5.1 Experimental results on case studies

5.2 Experimental results on synthetic data

Chapter 6

Towards an evaluation platform of inductive model synthesis

6.1 Motivation and overview

6.2 Scientific setup

6.2.1 State machines

6.2.2 Training and test samples

6.2.3 Submission and scoring

6.2.4 Baseline: Blue-Fringe performance

6.3 Competition results

Chapter 7

Tool Support

- 7.1 A FLTL Model-checker for g-hMSC Models
- 7.2 Interactive State-Machine Induction from Scenarios
- 7.3 The Gisele Clinical Pathway Analyzer

Chapter 8

Discussion and Related Work

Chapter 9

Conclusion