Program chairs

Kim G. Larsen, Aalborg, Denmark

Joost-Pieter Katoen, Aachen, Germany

Organizers

Andrzej Wąsowski, Copenhagen, Denmark

Uli Fahrenberg, Aalborg, Denmark

Contact

Uli Fahrenberg
Department of Computer Science, Aalborg University
Selma Lagerløfs Vej 300
9220 Aalborg Øst, Denmark
Phone: +45 99 40 89 28
Email: **uli@cs.aau.dk**

http://qmc.cs.aau.dk/

QMC 2010

Quantitative Model Checking

PhD School
Copenhagen, Denmark
2-5 March 2010
http://www.artist-embedded.org/artist/-Quantitative-Model-Checking-2010-.html

IT University of Copenhagen

Program

Tuesday 2 March **Finite-state model checking**

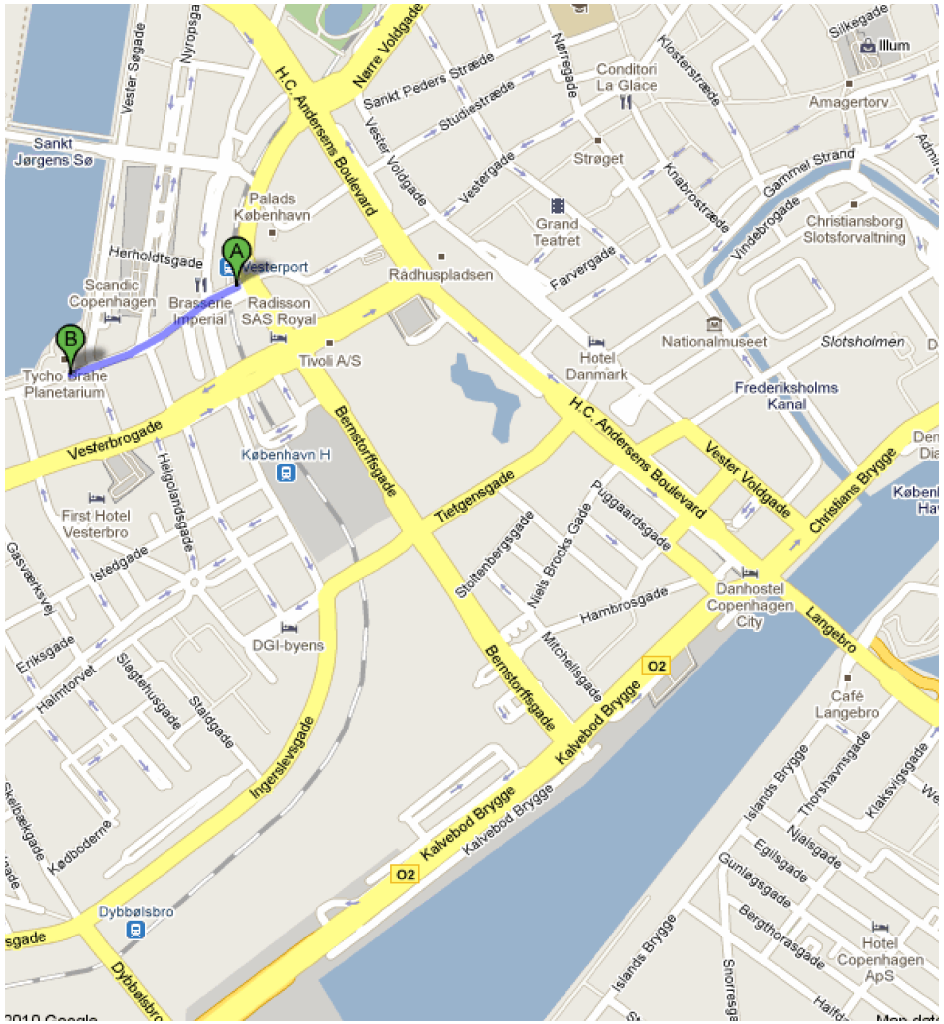
9:00		*Welcome*	**Atrium**
9:30	**Wolper**	Temporal logics and explicit-state model checking	
10:30		*Break*	
11:00	**Wolper**		
12:00		*Lunch*	**Atrium**
13:00	**Heljanko**	Bounded model checking for finite-state systems	**Aud. 2**
14:00		*Break*	
14:10	**Heljanko**		
15:10		*Coffee*	
15:40	**Cimatti**	Symbolic model checking	**Game lab**
17:40		*End*	

Wednesday 3 March **Real-time model checking**

Lectures by Bouyer-Decitre, Larsen, Markey

9:00	Introduction to timed automata	**Aud. 2**
9:45	*Coffee*	
10:15	Decidability and undecidability results	
11:05	*Break*	
11:10	Timed temporal logics	
12:00	*Lunch*	**Scrollbar**
13:00	UPPAAL, data structures and algorithms	**Aud. 2**
13:40	*Break*	
13:45	UPPAAL	**4A54, 4A56**
14:40	*Coffee*	
15:00	Timed games	**Aud. 2**
15:50	*Break*	
16:00	Priced timed automata	
16:45	Open problems	
17:15	*End*	

Social event



Social event

Wednesday 3 March we will visit the **Tycho Brahe Planetarium** in the center of Copenhagen. We will watch a 3D movie, you will be able to see their exhibition, and we will finish the evening by a dinner.

The social event starts at **18:00** and ends around 22:30. You will have to organize transportation from University yourself. As lectures finish at 17:15 this day, you will probably not have time to go by your hotel first.

The address of the planetarium is **Gammel Kongevej 10, 1610 Copenhagen**. You can get there by public transport as follows:

1. Walk from IT University to the DR Byen metro station.
2. Take a metro towards Vanløse and get off at Nørreport station (4 stops).
3. Switch to take an S-train towards København H and get off at Vesterport station (1 stop).
4. Leave Vesterport station by walking along the train direction forwards. Once at street level, turn right into Gammel Kongevej and continue for 5 minutes until you see the planetarium. See map on the facing page.

You will need a 2-zone ticket for this trip. At the metro station there is a vending machine at which you can buy such a ticket for 23 kr. It accepts coins and some credit cards. You can also buy a 10-trip-2-zone card *(klippekort)* for 135 kr, and use it on your other transfers. Note that those *klippekort* need to be validated (in the yellow machines) before each trip.

The trip takes just about 45 minutes including walking, so please set off immediately after the last lecture.

Program

Thursday 4 March — Probabilistic model checking

9:00	**Baier**	Probabilistic computation tree logic and quantitative linear-time properties	**Aud. 2**
10:00	*Coffee*		
10:30	**Baier**		
11:30	**Open session**		
12:00	*Lunch*		**Scrollbar**
13:00	**Katoen**	Model checking and abstraction of continuous-time Markov chains	**Aud. 2**
13:50	*Break*		
14:00	**Katoen**		
14:50	*Coffee*		
15:20	**Parker**	Probabilistic model checking in practice	**Aud. 2**
15:50	*Break*		
16:00	**Parker**		**2A52, 2A54**
17:30	*End*		

Friday 5 March — Hybrid model checking

9:00	**Raskin**	An introduction to hybrid automata	**Scrollbar**
9:45	*Coffee*		
10:15	**Raskin**		
11:00	*Break*		
11:05	**Frehse**	Tools for hybrid systems reachability	
11:45	*Break*		
11:55	**Frehse**		**Game lab**
13:00	*Lunch*		**Canteen**
14:00	**Fränzle**	Automatic analysis of hybrid systems	**Aud. 4**
14:45	*Break*		
14:55	**Fränzle**		
15:40	*Coffee*		
16:00	*School ends*		

Abstracts

Pierre Wolper

Université de Liege, Belgium

Temporal logics and explicit-state model checking

This lecture will introduce linear and branching time temporal logics and discuss the link between these logics and finite automata. Focusing on the linear-time case, the construction of automata from temporal logic formulas will be fully described as well as the classical automata-based model checking algorithms.

Keijo Heljanko

Aalto University, Helsinki, Finland

Bounded model checking for finite-state systems

This lecture will cover bounded model checking (BMC) for finite state systems. BMC is a symbolic model checking technique applying propositional satisfiability (SAT) solving techniques to model checking. We will cover the basics of BMC, encoding transition relations, encoding linear temporal logic formulas, as well as advanced topics in BMC.

Alessandro Cimatti

Istituto per la Ricerca Scientifica e Tecnologica, Trento, Italy

Symbolic model checking

This lecture will introduce various approaches to symbolic model checking for finite state systems. We will cover various simplification and abstraction techniques (guided reachability analysis, localization, CEGAR), and the use of various logical engines (Binary Decision Diagrams, SAT, and SMT). The lecture will include a hands-on session with the NuSMV model checker.

Abstracts

Martin Fränzle

Universität Oldenburg, Germany

Automatic analysis of hybrid systems

Within this lecture, we concentrate on automatic verification and analysis of hybrid systems, with a focus on symbolic methods manipulating both the discrete and the continuous state components symbolically by means of predicative encodings and dedicated constraint solvers. We provide a brief introduction to hybrid discrete-continuous systems, demonstrate the use of predicative encodings for compactly encoding operational high-level models, and continue to a set of constraint-based methods for automatically analyzing different classes of hybrid discrete-continuous dynamics. Covering the range from non-linear discrete-time hybrid systems to probabilistic hybrid systems, advanced arithmetic constraint solvers will be used as a workhorse for manipulating large and complex-structured Boolean combinations of arithmetic constraints arising in their analysis tasks.

Sponsors

MT-LAB
A VKR CENTRE OF EXCELLENCE

artist

Center for Indlejrede Software Systemer

IT University of Copenhagen

Abstracts

David Parker

University of Oxford, UK

Probabilistic model checking in practice

This tutorial will cover some of the practical aspects of probabilistic model checking. In particular, it will focus on the use of PRISM, a widely used tool for verification of discrete-time Markov chains, continuous-time Markov chains, Markov decision processes and extensions of these models with costs and rewards. The tutorial will cover PRISM's modeling and property specification languages and give an overview of its underlying techniques. It will also offer hands-on experience with the tool itself.

Jean-François Raskin

Université Libre de Bruxelles, Belgium

An introduction to hybrid automata

In this lecture, I will review hybrid automata: their syntax, semantics, and basic semi-algorithms for their analysis.

Goran Frehse

Université Joseph Fourier Grenoble 1, France

Tools for hybrid systems reachability

This lecture covers algorithms for computing the reachable states of two fundamental classes of hybrid systems, linear hybrid automata and piecewise affine systems, and how they can be extended to more general classes. The lecture is accompanied by a hands-on lab session.

Abstracts

Introduction to timed automata

In this first lecture we introduce the syntax and semantics of timed automata due to Rajeev Alur and David Dill. The usefulness of the formalism for describing interesting aspects of real-time systems will be illustrated through a number of introductory examples.

Patricia Bouyer-Decitre

École Normale Supérieure de Cachan, France

Decidability and undecidability results

In this lecture we highlight the region construction providing the means of several decidability results related to timed automata, including reachability, model checking as well as checking for a number of behavioral equivalences and preorders. The lecture also identifies the frontier of decidability for timed automata, including undecidability of timed trace inclusion and (un)decidability of reachability for timed automata extended with generalized updates and stopwatches.

Timed temporal logics

This lecture covers timed extensions of branching time logics (TCTL) as well as linear time logics (MTL, MITL) and the associated model checking problems.

Kim G. Larsen

Aalborg University Denmark

UPPAAL, data structures and algorithms

In this lecture the full modeling and specification formalism of the tool UPPAAL will be demonstrated on several examples. Also, a full account of the development of efficient data structuress (so-called zones represented by DBMs, CDDs, ..) and algorithms for the symbolic exploration of timed automata will be given as well as their availability via verification options in UPPAAL. This lecture will contain a hands-on exercise to be dealt with by participating students.

Abstracts

Priced timed automata

Priced Timed Automata (PTA) are an extension of timed automata with one (or more) continuous price variables allowing e.g. energy consumption to be modeled in a rather direct manner. PTA have proved useful for performance analysis and optimization of real-time systems with quantitative information. The lecture shows how the notions of regions and zones are extended with price-information allowing for decidability and efficient analysis of PTA.

Nicolas Markey

École Normale Supérieure de Cachan, France

Timed games

Timed games are games between two players, the controller and the opponent, played on a timed automaton. The possible moves of the two players are indicated by marking transitions of the timed automaton as either controllable or uncontrollable (i.e. under the control of the opponent). Given a control objective, specified in one of the timed temporal logics presented in lecture 3, the problem is to determine (and synthesize) the existence of a "winning" strategy of the controller that will guarantee the control objective to be satisfied regardless of the behavior of the opponent. Having introduced the notion of timed game, the lecture demonstrates decidability of winning strategies for safety, reachability as well as time-optimal reachability strategies. More challenging and practically useful, we consider the problem of synthesizing winning strategies for timed games with only partial observability. Decidability here depends heavily on the kind of observations that may be made. We also illustrate the application of timed games to settle behavioral equivalences and preorders.

Open problems

In the final lecture we point to a number of open problems remaining to be solved within the context of real-time model checking.

Abstracts

Christel Baier

Technische Universität Dresden, Germany

Probabilistic computation tree logic and quantitative linear-time properties

The tutorial provides an introduction to the model checking techniques for probabilistic systems modeled by finite-state discrete-time Markov chains and Markov decision processes. It considers the logic probabilistic computation tree logic and the quantitative analysis against linear-time properties using omega-automata, explains the treatment of fairness assumptions and the main ideas of partial order reduction for probabilistic parallel systems.

Joost-Pieter Katoen

RWTH Aachen, Germany

Model checking and abstraction of continuous-time Markov chains

This lecture will provide an introduction to the verification of CTMCs, a model that combines discrete probabilistic branching with random state residence times. CTMCs are prominent in performance and dependability evaluation, occur as semantic model of high-level modeling formalisms such as stochastic Petri nets and process algebras, and are frequently used in systems biology. We will introduce a branching-time logic on CTMCs, and explain in detail how the validity of these logical formulas can be model-checked on finite CTMCs. In order to handle large, or even infinite CTMCs, we introduce an abstraction technique that fits within the realm of three-valued abstraction methods. The key ingredients of this technique are a partitioning of the state space combined with an abstraction of transition probabilities by intervals. We will present the underlying theory of this abstraction, some examples, and indicate how such abstraction can be applied in a compositional manner.