IST-004527 ARTIST2
Network of Excellence
on Embedded Systems Design

Activity Progress Report for Year 4

JPRA-NoE Integration

Quantitative Testing and Verification

Clusters:

Testing and Verification

Activity Leader:

Professor Ed Brinksma (University of Twente, Embedded Systems Institute)
http://wwwhome.cs.utwente.nl/~brinksma/

Policy Objective (abstract)

The objective is to combine the efforts and skills of the individual leading researchers in Europe into a world-class virtual team, for advancing the state-of-the-art in verification and testing methodologies.

Achieving this objective requires development of theory, methods and tools for testing and verification of embedded systems with an emphasis on quantitative aspects (e.g. real-time and stochastic phenomena), that are of particular importance for the correctness of embedded systems.

A particular effort will be made to transfer knowledge, methods and tools to industry, including integration of the techniques developed into existing tools.

Table of Contents

1. Overview of the Activity

1.1 ARTIST Participants and Roles

Team Leader: Kim G. Larsen (BRICS/Aalborg)
Real-time and probabilistic verification and testing.

Team Leader: Ed Brinksma (University of Twente)
Model-based testing, stochastic modelling and verification.

Team Leader: Pierre Wolper (Centre Fédéré de Verification)
Model checking.

Team Leader: Philippe Schnoebelen (LSV)
Model checking.

Team Leader: Thierry Jéron (INRIA/Rennes)
Real-time testing.

Team Leader: Yassine Lakhnech (Verimag)
Infinite-state model checking.

Team Leader: Wang Yi (Uppsala)
Real-time verification and schedulability.

Team Leader: Tom Henzinger (EPFL)
Model checking algorithms for stochastic, real-time, and hybrid systems

Team Leader: Werner Damm (OFFIS)
Modelling and validation of safety-critical systems.

1.2 Affiliated Participants and Roles

Team Leader: Tretmans (Nijmegen)
testing

Team Leader: Bouajjani (LIAFA)
real-time and hybrid model checking

Team Leader: Lubos Brim (University Brno)
distributed model checking

Team Leader: Tommy Ericsson (Telelogic)
testing tool provider.

Team Leader: Sven H. Sørensen (Motorola A/S)
Areas of his team's expertise: development of embedded systems using model-driven methodology.

Team Leader: Christer Nordstöm （ABB Automation)
Areas of his team's expertise: Modelling and validation of industrial robotics.

Team Leader: Jan Lindblad (Enea Embedded Technology)
Areas of his team's expertise: Real Time Operating Systems and Testing.

Team Leader: Alain Ourghanlian (EDF Recherche et Développement)
Areas of his team's expertise: static analysis and model checking .

1.3 Starting Date, and Expected Ending Date

Start date September 1st,2004. Expected ending date August 31th 2008.

1.4 Baseline

The long-term ambition of the Testing and Verification cluster is to improve current industrial practice for developing embedded systems applications by continuous dissemination and improvement of existing powerful testing and verification techniques. For embedded systems – besides functional correctness – properties concerning quantitative aspects including real-time constraints and constraints on quality of services are of utmost importance. It is therefore our aim to provide modelling formalisms, methods and tools which will allow such quantitative aspects to be dealt with at early design stages and utilized in a systematic (and ideally automatic) approach in the testing phase. Also, based on existing powerful (real-time) verification techniques new research challenges of industrial importance is taken-up including optimal scheduling, monitoring and fault diagnosis, coverage metrics, controller synthesis, analysis of hybrid models (allowing to take into account the physical environment in which an application is used) and robustness and implementability of timed models. The involved partners include leading European teams with responsibility for some of the most mature methods and tools for testing and verification of functional, timing and QoS properties.

There are several ongoing collaborations, including:

- The start of the STREP projects *Quasimodo* and *Multiform* mark significant collaboration between several partners of the cluster (Aalborg, Twente, CFV, LSV and Aalborg, Twente, Verimag respectively). Also a number of teams affiliated with the cluster are partners in the proposal (Aachen, Saarlandes, ESI, Nijmegen).
- Numerous collaborations between LSV and Verimag on national projects (Eva, Prouvé, Rossignol, Action Spécifique du CNRS).
- Aalborg and Uppsala has since 95 continuously collaborated on the development of the tool UPPAAL in parallel with the development of Kronos at Verimag. In particular the collaboration has lead to a spin-off company (officially named UP4ALL).
- LSV, Aalborg and Twente are collaborating on problems related to optimal control and scheduling for real-time systems.
- CVF and Aalborg are collaborating on controller synthesis for real-time systems under partial observability and for efficient realization of synthesis algorithms within the verification tool UPPAAL.
- CVF, LSV and Aalborg are partners in the newly started ESF project GASICS: Games for Analysis and Synthesis of Interactive Computational Systems.
- Twente and INRIA have long been collaborating on testing methodologies and tools;
- INRIA and Verimag has for a long time collaborated on developing the testing tool TGV, and are currently collaborating on connecting IF and TGV within the Agedis IST project and the national project AS Testic
- CVF and EPFL have numerous collaborations on controller synthesis as well as analysis of stochastic model.

1.5 Problem Tackled in Year 4

The long-term ambition of the Testing and Verification cluster is to improve current industrial practice by continuous dissemination and improvement of existing powerful testing and verification techniques. Within the Quantitative Testing and Verification activity our aim is to provide modelling formalisms, methods and tools which will allow *quantitative* aspects to be dealt with and utilized for verification and performance analysis at early design stages as well as for systematic approaches to the testing phase.

The objectives for the 18 months period February 2007 until August 2008 included contined effort towards efficient tool components for controller synthesis, intiation of work on property-preserving code generation, development of generic frameworks using abstraction and compositionality for efficient analysis of quantitative models and new debugging and analysis techniques based on various combinations of testing and verification techniques.

Also, based on existing powerful (real-time) verification techniques work towards maturing and further development of important topics such as optimal scheduling, monitoring and fault diagnosis, controller synthesis, robustness and implementability of quantitative models and analysis of hybrid models, stochastic and timed models has been planned.

In more detail the following problems has been dealt with during the 18 months period (following closely the milestones of the 18 months period – see section 3.1 for more information):

Verification

- Verification and termination of a variety of channels systems
- Expressiveness and complexity of real-time model checking
- improved algorithms for the automata-based approach to model-checking
- Efficient model-checking based on symmetry reduction
- Application of infinite state-space exploration techniques (i.e., acceleration methods) to hybrid systems and timed automata.
- Study of the properties of automata-based symbolic representations of sets of integer and real vectors
- Development of efficient methods for iterating transducers (in the context of Regular Model Checking).
- Analysis of array systems and counter automata.
- Model-checking one-clock priced timed automata.
- Slicing for UPPAAL timed automata modelling formalism.

Testing

- Efficient testing of distributed systems based on adapted symbolic LTL or CTL model-checking of partial-order traces
- Development of new algorithms for monitoring timed and hybrid temporal properties, expressed in the logic MITL, against Boolean and analog signals
- Integration of verification and conformance testing.
- Model-based real-time testing using model checking (UPPAAL) or by synthesizing winning or coorperative test strategies (UPPAAL TIGA) from timed game formulations.
- Quantitative testing framework that takes into account measurement imprecisions.
- Testing scenario for probabilistic processes that exactly characterizes trace distribution equivalence.

Compositionality

- Component-based desing and analysis
- Compositional Verification for Component-based Systems
- Real-time Interface theory using I/O Timed Games, and using UPPAAL TIGA to settle compatibility, refinement and consistency between specifications.
- Architectural language to reason about (quantitative) dependability properties at an architectural level.

Abstraction and Approximate Analysis

- Refinement of abstract domains
- Incremental Component-Based Construction and Verfication of a Robotic System
- Generating Analog-Clock Real-Time Testers Using Action Refinement Techniques
- CEGAR (Counter-Example Guided Abstraction / Refinement) for linear hybrid systems with parameters.

Robustness and Implementability

- Robust analysis of timed systems.
- Robustness in timed parity games.

Controller Synthesis, Scheduling & Diagnosis

Numerous results within these areas have been achieved during the last year and from from several groups. The contributions include both several important theoretical results (settling the boundary between undecidability and decidability) as well as truly significant algorithmic advances in tool performance.

- Synthesis problems with budget constraints
- Synthesis under partial observability
- Stochastic games for the synthesis of systems with uncertain environments.
- model-checking of a Timed Alternating Temporal Logic on Timed Concurrent Game Structures
- Diagnosis for discrete event systems
- Diagnosis and control synthesis for information flow security
- Modular control synthesis
- Cost-optimal reachability strategies for one-clock priced timed automata.
- State Identification Problems for Input/Output Transition Systems
- Development of a method for synthesing controllers from Bounded-Response Properties expressed in MITL
- Definition and study a new class of scheduling problems where a request generator (a timed language) models the requests which are themselves, partially-ordered sets of tasks (jobs) that may require different types of resources.
- The tool UPPAAL TIGA is now completely integrated with UPPAAL allowing the full use of discrete datastructures (records and arrays) as well as user-defined datatypes and functions.
- By the end of the year an extension of UPPAAL TIGA supporting synthesis under partial observability will be made available.

Probabilistic and Stochastic Models

- Reachability analysis and general verification of probabilistic systems
- Definition of alternative (probabilistic and topological) semantics for timed automata (cooperation with LSV, INRIA-IRISA and TU Dresden)
- Decidability questions on probabilistic Büchi automata
- Probabilistic timed automata. An extension of UPPAAL is under development supporting the analysis of probability- and time-bounded reachability properties.

Priced Timed Automata, Resource and Quantitative Models

- Cost-bounded infinite runs for priced timed automata with negative and positive weights.
- Optimal infinte runs for priced timed automata using discounting or limit-ratio in the cost-measure.
- Infinite scheduling and optimal reachability for multi-priced timed automata.
- Relating Timed Petri Nets and Timed Automata
- Quantitative verification problems, where quantities may represent cost or resource use.
- Verification of Systems with Queues and Stacks
- Schedulability analysis for multiprocessor platforms
- Modular Performance Analysis
- Model-based validation of QoS properties
- Quantitative metrics and logics that respectively quantify how similar two systems are and how much a certain property is satisfied by a certain system.

1.6 Comments From Year 3 Review

1.6.1 Reviewers' Comments

For this activity there have been no particular comments.

1.6.2 How These Have Been Addressed

Since there were no specific comments, we did not take specific measures.

2. Summary of Activity Progress

2.1 Previous Work in Year 1

Work carried out in the first months include:

- The Vertecs team of INRIA has worked on test generation for models of infinite state systems with control and data. Systems are modelled with ioSTS (e.g. automata extended with data). Test generation from specification models and test purposes is based on syntactic transformations guided by approximate co-reachability analysis. The main achievements has been a new formalisation of symbolic test generation and a combination of verification and testing for safety properties.
- Uppsala has shown that the schedulability problem will be undecidable if tasks execution times may vary within an interval (representing the best and worst case execution times). They also developed an algorithm to compute the worst-case response times of non-uniformly recurring fixed-priority tasks. For systems containing only periodic tasks, the algorithm performs as well as the classic method for Rate-Monotonic Analysis. These results have been implemented in the TIMES tool for automated schedulability checking.
- A number of improvements have been made on the UPPAAL real-time model checker (www.uppaal.com). This includes the possibility to enrich the timed automaton models with C code. (Aalborg) This has given an important increase in the expressiveness of the modelling tool, e.g. the possibility to include advanced data types. During the period, the tool has been applied for off-line test generation on a connectivity testing framework.
- An extension of UPPAAL (UPPAAL Cora, Aalborg), dedicated to solving optimal scheduling and planning problems, has been introduced. This version is based on a version of the classical timed automaton formalism extended with auxiliary cost variables and with a modified version of the UPPAAL verification engine to take the accumulation of cost into account. During the period, several new algorithms have been designed for transforming the cost optimisation problem into a max-flow problem (in stead of a linear programming problem), and they will be introduced in forthcoming versions of the tool.
- Twente has carried out work on
    - Scheduling by reachability analysis: The feasibility of using search techniques from model checking to synthesize and analyse scheduling problems of industrial relevance was established.
    - Integrated quantitative analysis: The usefulness of model checking techniques for Markov chain analysis was further extended by application to Markov reward modelling. An industrial case study was carried out concerning an availability monitoring algorithm for self-configuring networks, with analysis carried out using the MODEST modelling formalism and the Moebius tool.
    - Modelling of hybrid systems: A process algebraic formalism for the modelling and analysis of hybrid systems has been developed.
    - Real-time testing: A real-time testing theory for quiescent systems has been formulated, implemented as a TorX extension, and extended to multi-channel interfaces.

- Information on formal methods relevant for industrial applications have been collected by OFFIS, and support was given to industrial partners to perform case studies on formal verification tools (commercial ones as well as academic ones). The work on case studies showed that it actually is possible to formally prove safety properties of e.g. existing car steering control software.
- Uppsala has developed a sampling semantics for timed automata, and shown that the new semantics gives rise to a natural notion of digitalization for timed lnguages. A recent result shows that the language inclusion problem in this setting is decidable, which in turn implies that for any timed automaton, a digital machine can be constructed systematically, which accepts the digitalized language of the automaton.
- A version of UPPAAL (UPPAAL Tron, Aalborg), dedicated to online testing of real time systems, has been announced. By using UPPAAL Tron, one can extend the testing power of traditional tools substantially, partly because one can run tests for a very long time, and also because Uppaal Tron gives the possibility to build various stochastic criteria into the test selection algorithm. During the period, further performance improvements have been made, and also a first realistic industrial case study has been made. The purpose of the study was to test the functionality of an existing electronic cooling thermostate, and several inconsistencies wrt. the product specification were revealed.
- Cachan has designed techniques for computing the convex hull of Presburger-definable sets of tuples of integers. These abstraction techniques are used in model-checking of complex counter systems; Improved techniques for verification of communicating systems including half-duplex channel systems and probabilistic lossy channel systems; Introduced the concept of "flat acceleration", a powerful generic algorithmic approach for the symbolic computation of reachability sets in regular model checking; In-depth study of the descriptive power of formalisms based on timed-automata and extensions, constrasted with verification costs; Model checking sets of paths: an approach that sits in between test and model checking. Also, quantitative analysis of priced timed automata, and used timed automata as a tool for fault diagnosis; Designed new probabilistic models supporting improved verification algorithms; Extensions of temporal logic formalisms, and associated verification techniques; Used UPPAAL for the verification of a multitask automation system.

2.2 Previous Work in Year 2

The following is a short summary of work carried out in the second year:

- We have released UPPAAL 4.0 being the result of over two and half years of development and contains many new features, additions to the modelling language, performance improvements, enhancements and polish to the easy to use graphical user interface, and libraries are available free of charge for academic, educational and evaluation purposes
- We have studied channel systems whose behaviour (sending and receiving messages via unbounded FIFO channels) must follow given timing constraints specifying the execution speeds of the local components.
- We have presented an algorithm for inferring a timed-automaton model of a system from information obtained by observing its external behavior. In this work, the full class of event-recording automata has been considered.

- We have worked on symbolic test selection for extended automata using abstract interpretation.
- We have worked on symbolic Determinisation of Extended Automata.
- We have implemented tool support for off-line test generation for real-time systems in UPPAAL Cover and UPPAAL Tron and applied it to a case study where a model-based approach to black-box testing is applied to verify that a Wireless Application Protocol (WAP) gateway conforms to its specification. The WAP gateway is developed by Ericsson and used in mobile telephone networks to connect mobile phones with the Internet.
- We have worked on conformance testing of programs with floating point numbers with respect to its specification with real numbers.
- We have worked on black-box testing of cryptographic protocols, using a compositional approach for checking secrecy and authenticity properties of cryptographic protocols integrating ideas from verification, conformance testing, and learning, with application to biometric passports.
- Work on verification of communication protocols using abstractiInterpretation of FIFO queues.
- Work on supervisory control of infinite symbolic systems using abstract interpretation.
- We have worked on analysis of priced (Weighted) timed automata, in particular settling decidability of optimal reachability in presence of multi cost functions and proved undecidability of model checking and optimal control in general (for priced timed automata with 3 or more clocks)
- We have worked on efficient implementation of cost-optimal reachability for priced timed automata using a symbolic A* algorithm in UPPAAL Cora.
- Work on robustness issues for timed and hybrid automata by introduction of a parametric semantics for timed controllers called the ASAP semantics.
- We have worked on analysis of O-minimal Hybrid Systems, refinement of abstraction for affine hybrid automata and development of an acceleration method suited for linear hybrid automata.
- We have developed and implemented an efficient on-the-fly algorithm for solving timed games wrt reachability and safety propertis. The implementation is available in the tool UPPAAL Tiga.
- For finite games we have proposed a fixed point theory of anti-chains to efficiently solve games of imperfect information.
- We have proposed algorithms for the verification of infinite state systems including rectangular abstrqctions of hybrid automata.
- We haved defined Quantitative similarity between timed systems and proposed logics for real-time games allowing to specify objectives.
- We have defined the formalism of Symbolic Transition Systems (STS) in order to support testing of systems with data. Also to support testing of communication protocls a testing theory allowing for action refinement was proposed.
- We have developed a framework for test coverage semantics as well as a testing theory for probabilistic processes.

- We have developed on-line testing of real-time systems in the tool UPPAAL Tron as well as a theory for conformance testing for real-time systems as used in the tool TTG.
- We have developed a framework for compositional reasoning about qualitatitve system properties.
- We have proposed a symbolic algorithm for the analysis of the robustness of timed automata, that is the correctness of the model in presence of small drifts on the clocks or imprecision in testing guards.
- We have presented a novel approach to synthesize good schedules for a class of scheduling problems that is slightly more general than certain existing scheduling problems.
- We have presented an (semi-decision procedure) algorithm for cost-bounded probabilistic reachability problem.
- We have studied the state identification problems for finite-state transducers, and the fundamental observation problem of decentralized observation.'
- We have provided an automatic method for calculating the path condition for programs with real time constraints. This method can be used for the semiautomatic verification of a unit of code in isolation, i.e., without providing the exact values of parameters with which it is called.
- We have showed how Allen's logic can be translated to LTL and how to synthesize automatically monitors for specifications in this logic.
- We have developed a framework for development and validation of product lines. In the approach families of embedded discrete finite state programs are modeled using input-enabled alternating transition systems. One model describes all functionality, while each variant is defined by an environment, describing its possible uses.
- We have worked on compositional verification using I/O-Automata

2.3 Previous Work in Year 3

The following is a short summary of what was carried out during the third year:

- We implemented efficient on-the-flh algorithms for solving timed games with respect to reachability and safety properties. The implementation has resulted in the branch UPPAAL TIGA providing a mature and fully integrated tool capable of synthesizing winning strategies for models exploiting the full modeling language of UPPAAL 4.0. Strategies can be represented as BDDs and CDDs providing compact formats for control code.
- An industrial application of UPPAAL TIGA to the synthesis of a climate controller for pig stables have been conducted. The resulting tool chain (UPPAAL TIGA and Simulink) has been applied by the company Skov A/S.
- We have shown decidability, provided an efficient on-the-fly algorithm for synthesizing stategies for timed games with partial observability. Also applications have been provided. The algorithm will be integrated with UPPAAL TIGA by the end of 2008.
- We have shown how to reduce various simulation and alternating simulation preorders between timed automata and timed game automata to safety games.

- The game-theretic approach has been applied to the testing of uncontrollable real-time systems. Specifying test purposes as TCTL fomulas, UPPAAL TIGA has been exployided to synthesis test strategies. Case studies have been conducted.
- Lower and upper bound complexity results for refinement and consistency of modal transition systems have been obtained.
- Model checking and optimal reachability for one-clock priced timed automata and priced tmed games have been shown decidable.
- Improved state space search algorithms for timed automata models – exploiting heuristic search and agent based techniques – have been described and implemented within the UPPAAL verification engine.
- The DBM Library of UPPAAL – which provided efficient implementation of the symbolic datastructure used for exploring timed automata state spaces – has been significantly improved.
- We have given formal semantics for dynamic fault trees (DFT) and developed compositional analysis techniques aliviating the state space explosion problem.
- We have developed equivalence relations and metrics for concurrent, stochastic games.
- We have proposed a symbolic algorithm for the analysis of the robustness of timed automata. Prototype implementation within UPPAAL has been made.
- We developed an semi-decision algorithm for cost-bounded probabilistic reachability in timed automata extended with prices (on edges and locations) and discrete probabilistic branching.
- We developed a framework for quantitative reasoning about quantitative systems. More precisely, we developed quantitative logics and quantitative refinement relations and showed their connections.
- A temporal interface for a component is a finite automaton that specifies the legal sequences of calls to functions that are provided by the component. We compared and evaluated three different algorithms for automatically extracting temporal interfaces from code
- We provided efficient methods for synthezing control for reactive systems by solving two-player games on graphs. The efficiency of the method avoids expensive determinization of (Büchi) automata.
- We have extended the game logic ATL with time quantifiers allowing objectives on real-time games to be specified.
- We studied stochastic graph games with omega-regular winning conditions specified as Rabin or Streett objectives. We developed a compositional theory of system verification, where specifications assign real-numbered costs to systems.
- We presented a verification methodology for cooperating traffic agents covering analysis of cooperation strategies, realization of strategies through control, and implementation of control.
- We analysed a flap controller (high-lift) case study. This application is derived from a case study for Airbus, a controller for the flaps of an aircraft.

- We analysed a distributed controller for Tramways. The "Single-tracked Line Segment" (SLS) case study stems from an industrial partner of the UniForM-project.
- The STG tool (test generation for models with control and data) has been improved, and a number of cases studies have been developped and experimented.
- A methodology integrating verification and conformance testing for the formal validation of reactive systems has been described.
- We have proposed an extension of the model of communicating automata (CFSM): Symbolic Communicating Machines (SCM), where messages carry data in infinite domains, and an approximate reachability analysis method on this model, based on lattice automata.
- Approximation and abstraction methods for the validation of for timed systems with respect to particular resource-related issues covering a broad range of resources such as processors, buffers and memory blocks etc.
- Ideas underlying our new algorithms for controller synthesis under imperfect information has recently been extended to solve classical problems in automata theory for finite word languages and infinite words.
- We have defined an new abstract fixpoint checking algorithm with automatic refinement by backward completion in Moore closed abstract domains. We have studied the properties of our algorithm and prove it to be more precise than the counterexample guided abstract refinement algorithm (CEGAR).
- Development of an acceleration method suited for the analysis of linear hybrid automata.
- Testing distributed systems through symbolic model checking
- Study of the properties of automata-based representation of sets of real numbers.
- A new technique for computing the convex hull of an automaton-represented finite set of integers was introduced.
- Development of a method for test case generation for ultimately periodic paths.
- Development of a conformance testing framework for hybrid systems, including definition of coverage measure for hybrid systems and algorithms for coverage-guided test generation.
- We developed new composability and compositionality techniques for deadlock-freedom implemented within BIP.
- Developed an abstract domain extending differerence-bounded matrices with disequality constraints.

2.4 Final Results

2.4.1 Technical Achievements

Aalborg

A Game-Theoretic Approach to Real-Time System Testing

This work presents a game-theoretic approach to the testing of uncontrollable real-time systems. By modelling the systems with Timed I/O Game Automata and specifying the test purposes as Timed CTL formulas, we employ a recently developed timed game solver UPPAAL-TIGA to synthesize winning strategies, and then use these strategies to conduct black-box conformance testing of the systems. The testing process is proved to be sound and complete with respect to the given test purposes. Case study and preliminary experimental results indicate that this is a viable approach to uncontrollable timed system testing.

Infinite Runs in Weighted Timed Automata with Energy Constraints with LSV, ENS Cachan, France

We study the problems of existence and construction of infinite schedules for finite weighted automata and one-clock weighted timed automata, subject to boundary constraints on the accumulated weight. More specifically, we consider automata equipped with positive and negative weights on transitions and locations, corresponding to the production and consumption of some resource (e.g. energy). We ask the question whether there exists an infinite path for which the accumulated weight for any finite prefix satisfies certain constraints (e.g. remains between 0 and some given upper-bound). We also consider a game version of the above, where certain transitions may be uncontrollable.

Complexity of Decision Problems for Mixed and Modal Specifications with ITU, Copenhagen and Imperial College London, UK

We consider decision problems for modal and mixed transition systems used as specifications: the common implementation problem (whether a set of specifications has a common implementation), the consistency problem (whether a single specification has an implementation), and the thorough refinement problem (whether all implementations of one specification are also implementations of another one). Common implementation and thorough refinement are shown to be PSPACE-hard for modal, and so also for mixed, specifications. Consistency is PSPACE-hard for mixed, while trivial for modal specifications. We also supply upper bounds suggesting strong links between these problems.

Testing Real-Time Systems Using UPPAAL with Uppsala University

This chapter presents principles and techniques for model-based black-box conformance testing of real-time systems using the Uppaal model-checking tool-suite. The basis for testing is given as a network of concurrent timed automata specified by the test engineer. Relativized input/output conformance serves as the notion of implementation correctness, essentially timed trace inclusion taking environment assumptions into account. Test cases can be generated offline and later executed, or they can be generated and executed online. For both approaches this chapter discusses how to specify test objectives, derive test sequences, apply these to the system under test, and assign a verdict.

Fast Directed Model Checking Via Russian Doll Abstraction, with University of Friburg and University of Innsbruck

Directed model checking aims at speeding up the search for bugs in a system through the use of heuristic functions. Such a function maps states to integers, estimating the state's distance to the nearest error state. The search gives a preference to states with lower estimates. The key issue is how to generate good heuristic functions, i.e., functions that guide the search quickly to an error state. An arsenal of heuristic functions has been developed in recent years. Significant progress was made, but many problems still prove to be notoriously hard. In particular, a body of work describes heuristic functions for model checking timed automata in Uppaal, and tested them on a certain set of benchmarks. Into this arsenal we add another heuristic function. With previous heuristics, for the largest of the benchmarks it was only just possible to find some (unnecessarily long) error path. With the new heuristic, we can find provably shortest error paths for these benchmarks in a matter of seconds. The heuristic function is based on a kind of Russian Doll principle, where the heuristic for a given problem arises through using Uppaal itself for the complete exploration of a simplified instance of the same problem. The simplification consists in removing those parts from the problem that are distant from the error property. As our empirical results confirm, this simplification often preserves the characteristic structure leading to the error.

Model-checking one-clock priced timed automata, with LSV, ENS Cachan, France

We consider the model of priced (a.k.a. weighted) timed automata, an extension of timed automata with cost information on both locations and transitions, and we study various model-checking problems for that model based on extensions of classical temporal logics with cost constraints on modalities. We prove that, under the assumption that the model has only one clock, model-checking this class of models against the logic WCTL, CTL with cost-constrained modalities, is PSPACE-complete (while it has been shown undecidable as soon as the model has three clocks). We also prove that model-checking WMTL, LTL with cost-constrained modalities, is decidable only if there is a single clock in the model and a single stopwatch cost variable (i.e., whose slopes lie in {0,1}).

Optimal infinite scheduling for multi-priced timed automata, with LSV, ENS Cachan, France and ESI, Eindhoven NL

This paper is concerned with the derivation of infinite schedules for timed automata that are in some sense optimal. To cover a wide class of optimality criteria we start out by introducing an extension of the (priced) timed automata model that includes both costs and rewards as separate modelling features. A precise definition is then given of what constitutes optimal infinite behaviours for this class of models. We subsequently show that the derivation of optimal non-terminating schedules for such double-priced timed automata is computable. This is done by a reduction of the problem to the determination of optimal mean-cycles in finite graphs with weighted edges. This reduction is obtained by introducing the so-called corner-point abstraction, a powerful abstraction technique of which we show that it preserves optimal schedules.

In this paper, we prove the decidability of the minimal and maximal reachability problems for multi-priced timed automata, an extension of timed automata with multiple cost variables evolving according to given rates for each location. More precisely, we consider the problems of synthesizing the minimal and maximal costs of reaching a given target location. These problems generalize conditional optimal reachability, i.e., the problem of minimizing one

primary cost under individual upper bound constraints on the remaining, secondary, costs, and the problem of maximizing the primary cost under individual lower bound constraints on the secondary costs. Furthermore, under the liveness constraint that all traces eventually reach the goal location, we can synthesize all costs combinations that can reach the goal.

Optimal reachability for multi-priced timed automata

The decidability of the minimal reachability problem is proven by constructing a zone-based algorithm that always terminates while synthesizing the optimal cost tuples. For the corresponding maximization problem, we construct two zone-based algorithms, one with and one without the above liveness constraint. All algorithms are presented in the setting of two cost variables and then lifted to an arbitrary number of cost variables.

Development of UPPAAL

In 2008 the concrete simulator of UPPAAL-TIGA was improved. It is now more stable and its interface has been updated. Another completely new algorithm was implemented to handle timed games with partial observability. It is now possible to define actions on edges inside the graphical editor and define observations when checking properties. The implementation only need to have loop detections added to be complete w.r.t. our paper. A second new major feature was also the implementation of timed games with Buchi accepting states, while avoiding some other bad states. This new algorithm is on-the-fly in the sense that it can stop whenever it has found a stable set of accepting and winning states. It works in two stages where a fix-point on the set of winning states is computed and then a fix-point on the set of winning states that are Buchi accepting.

Concerning UPPAAL, the engine is now able to merge DBMs dynamically when exploring the state-space. This is a transparent feature for the user. This is done automatically whenever possible.

Another new feature has been the addition of stop-watches. It is now possible to add to locations expressions of the form "x'==expr" where x is a clock and expr an expression evaluating to 0 or 1. There is no other needed syntax additions. Any clock can be stopped. However, the algorithm used becomes an over-approximation whenever a state that is stopping a clock is reached.

We also mention that the DBM library has been updated internally to cope with the extensions we have made. A new version will be released soon.

Discount-Optimal Infinite Runs in Priced Timed Automata

Discount-optimal infinites scheduling for priced timed automata has been shown decidable using region-based techniques (so-called corner-point abstraction). Using discounting in optimization criteria is a often used in Control Theory, and leads to a simple fixed-point characterization in the setting of weighted timed automata. The fixed-point characterization suggests an efficient algorithm in contrast to limit-ratio optimality.

Off-line test case generation

Off-line test-case generation from I/O timed automata models using increasingly techniques depending on properties of the given model. The techniques ranges from model checking (suitable if the model is controllable, i.e. deterministic and has neither timing uncertainty nor conflicting outputs), synthesis of testing strategy (suitable if the model is deterministic but fully

observable), synthesis of strategy under partial observability. Also, testing strategies with respect to a given test purpose but relying on corporation from the system under test have been given.

Timed Interface Theory

An interface theory for real-time systems using timed games have beedn developed. Here UPPAAL TIGA supports compatibility, refinement as well as consistency checking of component specifications.

Slicing for UPPAAL.

The focus of this thesis is to introduce slicing for Uppaal. Slicing is a technique based on static analysis used to reduce the syntactic size of models or applications. In this thesis, we show how slicing may be used to construct reachability preserving reductions of Uppaal models possibly improving the performance of the tool. Using automated slicing in Uppaal will eliminate the need for users to manually optimize models for faster verification of a certain property. Moreover, it allows less experienced users of Uppaal, which unknowingly may design models, containing unnecessary large amounts of data, to verify properties which \uppaal otherwise would have been unable to check.

Design Verification Patterns

Design Verification Patterns are formal specifications that define the semantics of design patterns. For each design pattern, the corresponding verification pattern give a set of proof obligations. They must be discharged for a correct implementation of the pattern. Additionally there is a set of properties that may be used in the design and verification of applications that employ the pattern. The concept is illustrated by examples from general software engineering and more specialised properties for embedded software.

Model-based Schedulability Analysis of Safety Critical Hard Real-Time Java Programs

This report describes the implementation of SARTS, a model based schedulability analysis tool used for hard real-time systems. SARTS is used to translate hard real-time systems, implemented in Java, to a timed automata model in UPPAAL.

The system being analyzed must be implemented in SCJ2, a safety critical profile for Java developed in this project, based on SCJ. The target hardware is the time predictable Java processor JOP, developed specifically for hard real-time systems.

Several experiments have been conducted to illustrate the accuracy of SARTS compared to existing tools. It is shown how the model based approach can result in a more accurate analysis, than possible with traditional analyses.

Twente

Quantative reasoning frameworks

We have further refined and extended our quantitative verification framework. This framework allows us to reason about quantitative properties in a quantitative way, that is, rather than saying if a property holds for a system, we express to what extend the property holds for a system We have developed a quantitative logic QLTL, which is a quantitative counterpart of QLTL; and in we have further developed our work on metrics for games and on the logic QLTL.

Moreover, we have used this framework to define a testing theory that describes how to test systems in the presence of measurement imprecisions.

Architectural dependability analysis

We have developed an architectural language that allows one to reason about dependability properties at an architectural level. By annotating an existing system design with dependability information (such as failure rater for components, or recovery times for processes), our method allows the designer to extract automatically analytical models from which various dependability measures (such as availability) can be computed automatically.

Testing Probabilistic Processes

We have defined a testing scenario that characterized in when two probabilistic processes exhibit the same visible behavior, i.e. when these are trace distribution equivalent. Along the way, we establish an Induction-Approximation theorem, which is of independent interest, since it related properties of infinite executions to properties of finite executions.

CVF

Synthesis with incomplete information (cooperation with EPFL, U Aalborg, and EC Nantes)

We have continued our collaboration with EPFL on algorithms for the synthesis of controller with imperfect information. In this research, we have proposed a fixed point theory to solve games of imperfect information. Â The fixed point theory is defined on the lattice of antichains of sets of states. Â Contrary to the classical solution proposed by Reif, our new solution does not involve determinization. Â As a consequence, it is readily applicable to classes of systems that do not admit determinization. Notable examples of such systems are timed and hybrid automata. Â As an application, we show that the discrete control problem for games of imperfect information defined by rectangular automata is decidable. This result extends a result by Henzinger and Kopke.

Those results have been extended to stuttering invariant and observation based strategy in collaboration with U Aalborg and EC Nantes. These results should be integrated into the tool UppAal-Tiga in 2008.

Improved algorithms for the automata-based approach to model-checking (in collaboration with EPFL)

Ideas underlying our new algorithms for controller synthesis under imperfect information have recently been extended to solve classical problems in automata theory for finite word languages and infinite words. With this new method, inclusion between two nondeterministic automata can be solved much more efficiently than with previously known algorithms. Those results should lead to the development of a new model-checking tool for linear time specifications expressed in LTL or using nondeterministic Buechi automata.

Fixed point based abstraction refinement (in collaboration with ENS Paris)

We have defined an new abstract fixpoint checking algorithm with automatic refinement by backward completion in Moore closed abstract domains. Â We have studied the properties of our algorithm and prove it to be more precise than the counterexample guided abstract

refinement algorithm (CEGAR). Â Contrary to several works in the literature, our algorithm does not require the abstract domains to be partitions of the state space. We have shown that our automatic refinement technique is compatible with so-called acceleration techniques. Â Furthermore, the use of Boolean closed domains does not improve the precision of our algorithm.

Development of an acceleration method suited for linear hybrid automata.

This method generalizes previous work on acceleration of integer-based systems, and provides a semi-algorithm for exploring the Â state-space of general linear hybrid automata, without abstracting away parts of the system or performing approximations. This method has been shown to be complete over the specific subclass of timed automata, but is also applicable to a much broader class of systems.

New efficient approximate verification based on symmetry markers.

This new verification technique can be used in various model-checker and in particular the spin tool. Â It exploits state-space symmetries induced by scalarset values used in a model. The technique involves efficiently computing a marker for each state encountered during search. A complete verification method only partially exploits symmetry; an approximate verification method fully exploits symmetry. We describe how symmetry markers can be efficiently computed and integrated into the SPIN tool. Â An empirical evaluation of our technique shows very good performance results and a high degree of precision for the approximate method (i.e. very few non-symmetric states receive the same marker). We also identify a class of models for which the approximate technique is precise.

Testing Distributed Systems through Symbolic Model Checking

The observation of a distributed system´s finite execution can be abstracted as partial order trace. We show that testing that such a distributed execution satisfies some global property amounts therefore to model check the corresponding trace. We provide an efficient symbolic CTL model checking algorithm for traces. This method is based on a symbolic data structure, called Interval Sharing Trees, allowing efficiently representing and manipulating sets of k-uples of naturals. Efficient symbolic operations are defined on this data structure in order to deal with all CTL modalities.

Study of the properties of automata-based representations of sets of real numbers

Automata-based representations of sets of real vectors are useful for manipulating the sets of configurations of infinite-state systems during state-space exploration. We have established that the sets of real vectors that can be represented by weak deterministic automata in all integer bases are exactly those thatare definable in first-order additive arithmetic. This generalizes to real numbers the well-known Cobham's theorem on the representability of sets of integers, and provides a theoretical justification to the use of weak deterministic automata as data structures for representing sets of reals in actual verification applications.

Alternative semantics for timed automata.

Like most models used in model-checking, timed automata are an idealized mathematical model used for representing systems with strong timing requirements. In such mathematical models, properties can be violated, due to unlikely (sequences of) events. We propose two

new semantics for the satisfaction of omega-regular properties, one based on probabilities, and the other one based on topology, to rule out these sequences. We prove that the two semantics are equivalent and lead to a PSPACE-Complete qualitative model-checking problem for LTL over finite executions. We also obtain decidability results for both qualitative and quantitative problems on infinite runs for one-clock timed automata.

Model-checking TATL on TCGS.

We propose a new model for timed games, based on concurrent game structures (CGSs). Compared to the classical timed game automata Â of Asarin et al., our timed CGSs are "more concurrent", in the sense that they always allow all the agents to act on the system, independently of the delay they want to elapse before their action. Timed CGSs weaken the "element of surprise" of timed game automata reported by de Alfaro et al. We prove that our model has nice properties, in particular that model-checking timed CGSs against timed ATL is decidable via region abstraction, and in particular that strategies are "region-stable" if winning objectives are. We also propose a new extension of TATL, containing ATL*, which we call TALTL. We prove that model-checking this logic remains decidable on timed CGSs. Last, we explain how our algorithms can be adapted in order to rule out Zeno (co-)strategies, based on the ideas of Henzinger et al.

Study of the properties of automata-based representations of sets of real numbers

Automata-based representations of sets of real vectors are useful for manipulating the sets of configurations of infinite-state systems during state-space exploration. We have established that the sets of real vectors that can be represented by infinite-word automata in all integer bases are exactly those that are definable in first-order additive arithmetic. As an important corollary, such sets can also be represented by weak deterministic automata, which can be used as actual data structures in verification applications. This result generalizes to real numbers the well-known Cobham's theorem on the representability of sets of integers, and provides a theoretical justification to the use of weak deterministic automata.

EPFL

Timed parity games: Complexity and robustness, EPFL

We considered two-player games played in real time on game structures with clocks and parity objectives. The games are concurrent in that at each turn, both players independently propose a time delay and an action, and the action with the shorter delay is chosen. To prevent a player from winning by blocking time, we restricted each player to strategies that ensure that the player cannot be responsible for causing a zeno run. First, we presented an efficient reduction of these games to turn-based (i.e., nonconcurrent) finite-state (i.e., untimed) parity games. The states of the resulting game are pairs of clock regions of the original game. Our reduction improved the best known complexity for solving timed parity games. Moreover, the rich class of algorithms for classical parity games can now be applied to timed parity games. Second, we considered two restricted classes of strategies for the player that represents the controller in a real-time synthesis problem, namely, limit-robust and bounded-robust strategies. Using a limit-robust strategy, the controller cannot choose an exact real-valued time delay but must allow for some nonzero jitter in each of its actions. If there is a given lower bound on the jitter, then the strategy is bounded-robust. We showed that exact strategies are more powerful than limit-robust strategies, which are

more powerful than bounded-robust strategies for any bound. For both kinds of robust strategies, we presented efficient reductions to standard timed automaton games. These reductions provide algorithms for the synthesis of robust real-time controllers.

Trading infinite memory for uniform randomness in timed games, EPFL

We considered concurrent two-player timed automaton games with omega-regular objectives specified as parity conditions. These games offer an appropriate model for the synthesis of real-time controllers. Earlier works on timed games focused on pure strategies for each player. We studied, for the first time, the use of randomized strategies in such games. While pure (i.e., nonrandomized) strategies in timed games require infinite memory for winning even with respect to reachability objectives, we showed that randomized strategies can win with finite memory with respect to all parity objectives. Also, the synthesized randomized real-time controllers are much simpler in structure than the corresponding pure controllers, and therefore´easier to implement. For safety objectives we proved the existence of pure finite-memory winning strategies. Finally, while randomization helps in simplifying the strategies required for winning timed parity games, we proved that randomization does not help in winning at more states.

Minimum-time reachability in timed games, EPFL and CVF

We considered the minimum-time reachability problem in concurrent two-player timed automaton game structures. We showed how to compute the minimum time needed by a player to reach a target location against all possible choices of the opponent. We did not put any syntactic restriction on the game structure, nor did we require any player to guarantee time divergence. We only required players to use receptive strategies which do not block time. The minimal time is computed in part using a fixpoint expression, which we showed can be evaluated on equivalence classes of a non-trivial extension of the clock-region equivalence relation for timed automata.

Quantitative languages, EPFL:Quantitative generalizations of classical languages, which assign to each word a real number instead of a boolean value, have applications in modeling resource-constrained computation. We used weighted automata (finite automata with transition weights) to define several natural classes of quantitative languages over finite and infinite words; in particular, the real value of an infinite run is computed as the maximum, limsup, liminf, limit average, or discounted sum of the transition weights. We defined the classical decision problems of automata theory (emptiness, universality, language inclusion, and language equivalence) in the quantitative setting and study their computational complexity. As the decidability of language inclusion remains open for some classes of weighted automata, we introduced a notion of quantitative simulation that is decidable and implies language inclusion. We also gave a complete characterization of the expressive power of the various classes of weighted automata. In particular, we showed that most classes of weighted automata cannot be determinized.

Controller synthesis with budget constraints, EPFL:

We studied the controller synthesis problem under budget constraints. In this problem, there is a cost associated with making an observation, and a controller can make only a limited number of observations in each round so that the total cost of the observations does not exceed a given fixed budget. The controller must ensure some

omega-regular requirement subject to the budget constraint. Such budget constraints arise in designing and implementing controllers for resource-constrained embedded systems, where a controller may not have enough power, time, or bandwidth to obtain data from all sensors in each round. Budget constraints lead to games of imperfect information, where the unknown information is not fixed a priori, but can vary from round to round, based on the choices made by the controller how to allocate its budget. We showed that the budget-constrained synthesis problem for omega-regular objectives is complete for exponential time. In addition to studying synthesis under a fixed budget constraint, we studied the budget optimization problem, where given a plant, an objective, and observation costs, we have to find a controller that achieves the objective with minimal accumulated cost (or minimal peak cost). We showed that this problem is reducible to a game of imperfect information where the winning objective is a conjunction of an omega-regular condition and a long-run average condition (or a least max-cost condition), and this again leads to an exponential-time algorithm. Finally, we extended our results to games over infinite state spaces, and show that the problems are decidable for infinite state games with stable quotients of finite index. Consequently, the discrete time budget-constrained synthesis and budget optimization problems are decidable for rectangular hybrid automata.

Model checking omega-regular properties of interval Markov chains, EPFL

We studied the problem of model checking Interval-valued Discrete-time Markov Chains (IDTMC). IDTMCs are discrete-time finite Markov Chains for which the exact transition probabilities are not known. Instead in IDTMCs, each transition is associated with an interval in which the actual transition probability must lie. We considered two semantic interpretations for the uncertainty in the transition probabilities of an IDTMC. In the first interpretation, an IDTMC represents a (possibly uncountable) family of (classical) discrete-time Markov Chains, where each member of the family is a Markov Chain whose transition probabilities lie within the interval range given in the IDTMC. We call this semantic interpretation Uncertain Markov Chains (UMC). In the second semantics for an IDTMC, which we call Interval Markov Decision Process (IMDP), the uncertainty is resolved through non-determinism. In other words, each time a state is visited, we adversarially pick a transition distribution that respects the interval constraints, and take a probabilistic step according to the chosen distribution. We introduced a logic, omega-PCTL, that can express liveness, strong fairness, and omega-regular properties (such properties cannot be expressed in PCTL). We showed that the omega-PCTL model checking problem for Uncertain Markov Chain semantics is decidable in PSPACE (same as the best known upper bound for PCTL), and for Interval Markov Decision Process semantics it is decidable in coNP (improving the previous known PSPACE bound for PCTL). We also showed that the qualitative fragment of the logic can be solved in coNP for the UMC interpretation, and can be solved in polynomial time for a sub-class of UMCs. We also proved lower bounds for these model checking problems. We showed that the model checking problem of IDTMCs with LTL formulas can be solved for both UMC and IMDP semantics by reduction to the model checking problem of IDTMC with omega-PCTL formulas

.

Equivalence of labeled Markov chains, EPFL and CVF

We considered the equivalence problem for labeled Markov chains (LMCs), where each state is labeled with an observation. Two LMCs are equivalent if every finite sequence of observations has the same probability of occurrence in the two LMCs. We showed that equivalence can be decided in polynomial time, using a reduction to the equivalence problem for probabilistic automata, which is known to be

solvable in polynomial time. We provided an alternative algorithm to solve the equivalence problem, which is based on a new definition of bisimulation for probabilistic automata. We also extended the technique to decide the equivalence of weighted probabilistic automata. Then, we considered the equivalence problem for labelled Markov decision processes (LMDPs), which asks given two LMDPs whether for every scheduler (i.e., way of resolving the nondeterministic decisions) for each of the processes, there exists a scheduler for the other process such that the resulting LMCs are equivalent. The decidability of this problem remains open. We showed that the schedulers can be restricted to be observation-based, but may require infinite memory.

Stochastic limit-average games are in EXPTIME, EPFL:

We showed that the value of a finite-state two-player zero-sum stochastic game with limit-average payoff can be approximated to within epsilon in time exponential in a polynomial in the size of the game times polynomial in logarithmic in 1/epsilon.

Value iteration, EPFL:

We surveyed value iteration algorithms on graphs. Such algorithms can be used for determining the existence of certain paths (modelchecking), the existence of certain strategies (game solving), and theprobabilities of certain events (performance analysis). We classified the algorithms according to the value domain (boolean, probabilistic,or quantitative); according to the graph structure (nondeterministic,probabilistic, or multi-player); according to the desired property of paths (Borel level 1, 2, or 3); and according to the alternation depth and convergence rate of fixpoint computations.

Brno

DeVinE Tool

We have further optimized the DiVinE tool and evaluated its performance and scalability on large-scale parallel systems. For preliminary experiments we have used the Distributed ASCI Supercomputer, a wide-area distributed system for Computer Science research in the Netherlands. As a next step we plan to make large scalability tests on the new cluster in Aaalborg. We have extended the DiVinE tool with I/O efficient verification algorithms, which allow to exploit parallel hard disks. Quantitative LTL mode-checking has been added to the probabilistic version of DiVinE.

IRISA

Verification of Systems with Queues and Stacks:

Many scientific studies analysed the FIFO channel systems, but none offered a fully satisfying solution. We proposed to tackle this problem within the abstract interpretation framework, by defining some abstract lattices adapted to this kind of systems. We considered systems with a finite alphabet of messages, then more complex systems, with an infinite alphabet of messages. This leads us to define and to study a new kind of automata: the lattice automata. Those automata are also useful for the analysis of programs with a call stack.

Quantitative Model-Checking of One-Clock Timed Automata:

We have defined two relaxed semantics (one based on probabilities and the other one based on the topological notion of largeness) for LTL over infinite runs of timed automata which rule out unlikely sequences of events. We proved that these two semantics match in the framework of single-clock timed automata (and only in that framework), and proved that the corresponding relaxed model-checking problems are PSPACE-Complete. Moreover, we proved that the probabilistic non-Zenoness can be decided for single-clock timed automata in NLOGSPACE.

We consider the quantitative model-checking problem for omega-regular properties: we aim at computing the exact probability that a given timed automaton satisfies an omega-regular property. We develop a framework in which we can compute a closed-form expression for this probability; we furthermore give an approximation algorithm, and finally prove that we can decide the threshold problem in that framework.

Probabilistic Büchi Automata:

Probabilistic Büchi automata (PBA) are finite-state acceptors for infinite words where all choices are resolved by fixed distributions and where the accepted language is defined by the requirement that the measure of the accepting runs is positive. The main contribution of this paper is a complementation operator for PBA and a discussion on several algorithmic problems for PBA. All interesting problems, such as checking emptiness or equivalence for PBA or checking whether a finite transition system satisfies a PBA-specification, turn out to be undecidable. An important consequence of these results are several undecidability results for stochastic games with incomplete information, modelled by partially-observable Markov decision processes and omega-regular winning objectives. Furthermore, an alternative semantics for PBA is discussed where it is required that almost all runs for an accepted word are accepting, which turns out to be less powerful, but has a decidable emptiness problem.

Diagnosis and predictability:

We studied the problem of predicting the occurrences of a pattern in a partially-observed discrete-event system. The system is modeled by a labeled transition system. The pattern is a set of event sequences modeled by a finite-state automaton. The occurrences of the pattern are predictable if it is possible to infer about any occurrence of the pattern before the pattern is completely executed by the system. An off-line algorithm to verify the property of predictability is presented. The verification is polynomial in the number of states of the system. An on-line algorithm to track the execution of the pattern during the operation of the system is also presented. This algorithm is based on the use of a diagnoser automaton.

Diagnosis and control synthesis for infomation flow security:

We have been interested in constructing monitors for the detection of confidential information flow in the context of partially observable discrete event systems. We focus on the case where the secret information is given as a regular language. We first characterized the set of observations allowing an attacker to infer the secret behaviors. We considered the general case where the attacker and the administrator have different partial views of the system. Further, based on the diagnosis of discrete event systems, we provide necessary and sufficient conditions under which detection and prediction of secret information flow can be ensured and a construction of a monitor ensuring this task.

Given a finite transition system and a regular predicate, we also addressed the problem of computing a controller enforcing the opacity of the predicate against an attacker (that partially observes the system), supposedly trying to push the system to reveal the predicate. Assuming that the controller can only control a subset of the events it observes (possibly different

from the ones of the attacker), we showed that an optimal control always exists and provide sufficient conditions under which it is regular and effectively computable. These conditions rely on the inclusion relationships between the observable alphabets of the attacker and the controller and the controllable alphabet.

Modular control synthesis:

In this work sufficient conditions for modular (supervisory) control synthesis are presented which equal global control synthesis. In modular control synthesis a supervisory control is synthesized for each module separately and the supervisory control consists of the parallel composition of the modular supervisory controls. The general case of the specification that is indecomposable and not necessarily contained in the plant language, which is often the case in practice, is considered. The usual assumption that all shared events are controllable is relaxed by introducing two new structural conditions relying on the global mutual controllability condition. The novel concept used as a sufficient structural condition is strong global mutual controllability. The main result uses a weaker condition called global mutual controllability together with local consistency of the specification. An example illustrates the approach.

Integration of verification and testing:

A methodology integrating verification and conformance testing for the formal validation of reactive systems has been proposed. A specification of a system - an extended input-output automaton, which may be infinite-state - and a set of safety properties (“nothing bad ever happens”) and possibility properties (“something good may happen”) are assumed. The properties are first tentatively verified on the specification using automatic techniques based on approximated state-space exploration, which are sound, but, as a price to pay for automation, are not complete for the given class of properties. Because of this incompleteness and of state-space explosion, the verification may not succeed in proving or disproving the properties. However, even if verification did not succeed, the testing phase can proceed and provide useful information about the implementation. Test cases are automatically and symbolically generated from the specification and the properties, and are executed on a black-box implementation of the system. The test execution may detect violations of conformance between implementation and specification; in addition, it may detect violation/satisfaction of the properties by the implementation and by the specification. In this sense, testing completes verification.

Discrete controller synthesis for modular reactive systems

We here focused on the exploitation of particularities of modular reactive systems for the application of discrete controller synthesis (DCS). We have proposed a schema of integration of DCS techniques into the modular compilation of an extended synchronous language. In this extended language, the modularity is expressed by nodes, representing components associated with modular synthesis objectives ; we can then obtain, by application of DCS tools on these components, some synchronous controllers controlling parts of programs. In this framework, we implemented a translation schema of a subset of the Lucid Synchrone language into dynamic systems, for further application of Sigali, as DCS tool. Future work will consist in applying decentralized control methods, together with modular distribution of synchronous

programs, in order to obtain automatically, from an annotated synchronous program, a distributed controlled system.

This work is part of the post-doc of Gwenael Delaval funded by Artist 2 and a cooperative work between the pop-art EPI (Inria Grenoble) and the VerTeCs EPI (Inria Rennes).

Control of Infinite Symbolic Transitions Systems under Partial Observation

We have proposed algorithms for the synthesis of memoryless controllers through partial observation of infinite state systems modelled by STS. We provide models of safe controllers both for potentially blocking and non blocking controlled systems. To obtain algorithms for this problems, we use abstract interpretation techniques which provide overapproximations of the transitions set to disable. To our knowledge, with the hypotheses taken, the improved version of our algorithm provides a better solution than what was previously proposed in the literature. Our tool SMACS allowed us to make an empirical validation of our methods to show their feasibility and usability.

This work has been done in cooperation with T. Massart and G. Kalyon (Université libre de Bruxelles, Belgium).

STG symbolic test generation tool:

The tool has been improved and a new version can be downloaded from Inria Gforge: https://gforge.inria.fr/plugins/scmsvn/viewcvs.php/?root=bjeannet

VERIMAG

Logics for programs with integer arrays

Programs with integer arrays poses interesting challenges for the existing methods and tools for software verification. In particular, logics for reasoning about infinite state spaces modeling unbounded arrays are required by e.g. predicate abstraction, abstract interpretation or Hoare-style program proofs. Moreover, push-button verification needs decidable logics in which program properties can be expressed.

We have developped two decidable logics in which universally quantified array properties can be expressed. These logics enhance the expressivity of existing logics by allowing arithmetic comparisons between adjacent elements of arrays (such as difference bounds constraints, or octagonal constraints). The decision procedures for the logics are based on translations to classes of counter automata, for which the emptiness problem (existence of a run leading to some final state) is decidable.

Monitoring Real-Time Properties:

Formal verification is a very ambitious activity due to its exhaustiveness and for this reason testing/simulation is still the most commonly used validation techniques. Nevertheless, testing can be made more formal by employing a precise formal specification logic based on temporal logic, against which simulation traces can be checked. This technique called monitoring or

runtime verification is gaining popularity and it does not suffer from the state explosion and other difficulties associated with traditional model checking. Motivated by the verification of analog circuits we have developed new algorithms for monitoring timed and hybrid temporal properties, expressed in the logic MITL, against Boolean and analog signals. And, we have developed a tool that has rised interest in several companies (ST, Freescale, Rambus and Mentor Graphics) and applied it to several case studies.

Synthesis from Bounded-Response Properties:

The problem is synthesizing controllers directly from high-level specification is very challenging and proposed theoretical solutions are prohibitively complex. We have suggested a simpler variant of this problem where we restrict ourselves to bounded-response properties which are equivalent to safety properties and hence do not require complex omega automata and their determinization. We express these properties in the real-time logic MITL and then, by transforming them to past properties we can rely on our previous results we construct a deterministic timed automaton to which synthesis algorithms are applied.

Scheduling Policies for Streams of Structured Jobs:

The need to process efficiently streams of tasks that arrive nondeterministically is a crucial problem in embedded system design. Traditional models of real-time systems are not always appropriate for these situations as they often treat periodic and independent tasks. We have defined a new class of scheduling problems where a request generator (a timed language) models the requests which are themselves, partially-ordered sets of tasks (jobs) that may require different types of resources. On these models we prove some fundamental results concerning schedules and scheduling strategies of bounded backlog and latency [DM08].

Formal Verification of Linear Hybrid Automata with PHAVer:

Linear hybrid automata (LHA) are characterized by piecewise constant bounds on the derivatives. They are of interest in formal verification because their dynamics are so simple that basic operators such as discrete and continuous successor states can be computed with exact integer arithmetic, and relatively efficiently over an infinite time horizon. Our tool PHAVer uses exact polyhedral computations to compute the set of reachable states and to verify equivalence and abstraction between automata using assume/guarantee reasoning. Its characteristic is the ability to conservatively overapproximate polyhedra with polyhedra of substantially lower complexity. It is also able to handle more complex dynamics, a generalized form of piecewise affine dynamics, by overapproximation. On-the-fly, adaptive partitioning allows us to target the accuracy of the overapproximation to relevant parts of the state space. Forward/backward refinement, in which the partitioning is iteratively refined while alternating forward and backward reachability, has allowed us to formally verify oscillation of a nonlinear circuit model with three state variables.

Another verification approach for LHA avoids polyhedral computations altogether. It exploits the fact that for LHA reachability along a given path can be formulated as satisfiability of a conjunction of linear constraints, and can therefore be computed very efficiently using linear programming techniques. Various methods of counter example guided abstraction refinement have been proposed in literature to verify safty. We have generically extended these methods to parameter synthesis.

Quantitative verification of embedded software :

We have worked on three research directions concerning quantitative verification of embedded software.

The design and implementation of software-intensive embedded product lines requires dealing with a variety of constantly changing application- and system-dependent quantitative non-functional requirements and constraints that need to be verified throughout the development process. Moreover, because product lines are built upon a set of core services which are improved, customized, extended and integrated to come up with differentiated products, there is a need to resort to component-based approaches. However, many embedded applications (e.g., video compression) are most likely specified in a transformational data-oriented style. The componentization of such applications is therefore deferred to the implementation phase, where performance and platform constraints are taken into account. In [YADZB08] we presented a formally-grounded method to carry on this process. The approach consists in integrating (1) the component-based language and execution engine BIP, and (2) the coordination language and code-generation infrastructure FXML/Jahuel. This enables verification of quantitative properties using the associated BIP tool-suite.

-AADL is an aerospace standard for model-driven design of complex real-time embedded systems. Currently, behavioral properties of AADL models can be specified inside the system description using AADL concepts or outside it using external textual languages, and verified using schedulability analysis or (Time Petri Net-based) model-checking tools. Our work [MOYB08] (1) proposes Visual Timed Scenarios (V TS) as a graphical property specification language for AADL, and (2) devises an effective translation from V TS to Time Petri Nets (TPN) which enables model-checking properties expressed in V TS over AADL models using TPN-based tools integrated into AADL-compliant IDEs (e.g., TOPCASED).

-3. In [3] we have developed a technique to compute symbolic polynomial approximations of the amount of dynamic memory required to safely execute a method without running out of memory, for Java-like imperative programs. Given an initial configuration of the stack and the heap, the peak memory consumption is the maximum space occupied by newly created
objects in all states along a run from it. We over-approximate the peak memory consumption using a scoped-memory management where objects are organized in regions associated with the lifetime of methods. We model the problem of computing the maximum memory occupied by any region configuration as a parametric polynomial optimization problem over a polyhedral domain and resort to Bernstein basis to solve it. We apply the developed tool to several benchmarks.

Compositional Verification for Component-based Systems

We have presenedt a compositional method for the verification of component-based systems described in a subset of the BIP language encompassing multi-party interaction without data transfer. The method is based on the use of two kinds of invariants. Component invariants which are over-approximations of components' reachability sets. Interaction invariants which are constraints on the states of components involved in interactions. Interaction invariants are obtained by computing traps of finite-state abstractions of the verified system. The method is applied for deadlock verification in the D-Finder tool. D-Finder is an interactive tool that takes as input BIP programs and applies proof strategies to eliminate potential deadlocks by computing increasingly stronger invariants. The experimental results on non-trivial examples allow either to prove deadlock-freedom or to identify very few

deadlock configurations that can be analyzed by using state space exploration.

Incremental Component-Based Construction and Verfication of a Robotic System

Autonomous robots are designed to perform high level tasks on their own , or with very limited external control. They are needed in situations where human control is either infeasible or not cost-effective. Designing and developing software for an autonomous robot is quite a challenging and complex task.

In [BGLN08], [BIS08] we present an evolution of the LAAS Architecture for Autonomous System and its tool GenoM. This evolution is based on the BIP component based design framework which has been successfully used in other domains (e.g. embedded systems). In this study, we show how we seamlessly integrate BIP in the preexisting methodology. We present the componentization of the functional level of a robot, the synthesis of an execution controller as well as validation techniques for checking essential safety properties. This approach has been integrated in the LAAS architecture and we have performed a number of experiment in simulation but also on a real robot (DALA).

Generating Analog-Clock Real-Time Testers Using Action Refinement Techniques:

In a previous work we proposed a method for generating digital-clock tests for real-time systems using action refinement techniques. We have extended this method for generating analog-clock testers. Analog-clock testers are testers which can observe real-time with precision. Our goal from testing is to check the conformance of a given implementation with respect to a given specification (the model). The main benefit of the method is to save memory space needed to build and to store tests. One important contribution of this work is a simplified way for both modelling and testing real-time systems. We first write a (high-level) simplified version of the model of the system, as an input-output transition system (IOTS) and then we refine it into a more detailed (low-level) model as a timed input-output transition system (TIOTS). This same mechanism applies to the test generation procedure.

Automatic Generation of Path Conditions for Concurrent Timed Systems: We concentrate on the automatic generation of test cases for concurrent real-time systems. In order to test a particular behavior of the system, we generate path conditions for (concurrent real-time) execution paths. Instantiating such path conditions allows us to test the desired path. We do not assume finite state systems. Hence our modeled systems may reference unbounded variables in tests and assignments (when we ignore the particular word length in a given machine). Such a precondition characterizes all the states from which we can execute the path. However, there may be other possible executed paths, due to nondeterministic choice, which can be eliminated by adding further synchronization. The path condition calculation can be used in a model checking search, hunting for a path satisfying a given temporal property. It allows us to verify a procedure or collection of procedures in isolation, without providing initial values. Using the weakest precondition calculation, verification is performed symbolically, or ?for all parameters at once?. The temporal property is translated into an automaton and contributes to calculation of the path condition (i.e., it is a condition for executing a path while satisfying the temporal property).
For the real-time case, we need to generalize the calculation of a path condition, taking into account only the essential conditions to follow a particular path in the execution. For example, if the path is abcd, we may constrain only a to precede b, for being on the same process, c to precede d, again, for being on the other process, and b to precede d, for referring to the same variable. We start with a given path (in the flow chart, or interleaved from different flow charts for concurrent processes) merely from a practical consideration; it is very simple to specify an interleaved execution sequence. However, we look at the essential partial order, which is consistent with real-time constraints, rather than at the total order. We cannot assume that

transitions must follow each other, unless this order stems from some sequentiality constraints (such as transitions belonging to the same process or using the same variable) or from timing constraints. Thus, with the above restrictions, acbd is equivalent to abcd and represents the same (partial order) execution. For untimed systems, there is no difference between the condition for partial order execution and the condition for executing any of the sequences (linearizations) consistent with it. Because of commutativity between concurrently executed transitions, we obtain the same path condition either way. However, when taking time constraints into account, the actual time and order between occurrences of transitions does affect the path condition (which now includes time information).

OFFIS

Verification of collision avoidance protocols

We have identified common principles underlying such protocols in what could be called “design patterns” for collision avoidance protocols, building on the key concepts of criticality functions and safety envelopes, and demonstrated the wide applicability with examples from avionics, rail, and automotive applications.

We have provided methods to explore the design space of such protocols, specifically allowing to analyze the interdependencies between system-parameters (e.g. the maximal response times in reaction to external events to ensure collision freedom) and emergent non-functional parameters of components (such as the maximal breaking force, or maximal communication delay for wire-less channels), based on a parametric hybrid logic.

We have developed a verification methodology for such applications using a variant of the ETCS protocol as running example addressing both derived safety and stability requirements.

Verification on non-linear hybrid systems

We have provided abstraction refinement based approaches to the verification of both discrete time and dense time models of non-linear hybrid systems exploiting robustness.

Verification of hybrid systems with large discrete state spaces

We have stretched the limits of maintaining precise abstraction in the verification of hybrid systems with large discrete state spaces by lifting AIG based verification to AIG(T) based model-checking algorithms for both discrete and dense time models of hybrid systems. We have been developing an abstraction refinement based verification technique for open-loop discrete-time models with large discrete state spaces and demonstrated its scalability on industrial size models.

2.4.2 Individual Publications Resulting from these Achievements

Aalborg

Kim Guldstrand Larsen, Jacob Illum Rasmussen: Optimal reachability for multi-priced timed automata. Theor. Comput. Sci. 390(2-3): 197-213 (2008)

Patricia Bouyer, Ed Brinksma, Kim Guldstrand Larsen: Optimal infinite scheduling for multi-priced timed automata. Formal Methods in System Design 32(1): 3-23 (2008)

Patricia Bouyer, Kim Guldstrand Larsen, Nicolas Markey: Model Checking One-clock Priced Timed Automata CoRR abs/0805.1457: (2008)

Sebastian Kupferschmid, Jörg Hoffmann, Kim Guldstrand Larsen: Fast Directed Model Checking Via Russian Doll Abstraction. TACAS 2008: 203-217

Anders Hessel, Kim Guldstrand Larsen, Marius Mikucionis, Brian Nielsen, Paul Pettersson, Arne Skou: Testing Real-Time Systems Using UPPAAL. Formal Methods and Testing 2008: 77-117

Adam Antonik, Michael Huth, Kim Guldstrand Larsen, Ulrik Nyman, Andrzej Wasowski: Complexity of Decision Problems for Mixed and Modal Specifications. FoSSaCS 2008: 112-126

Patricia Bouyer, Ulrich Fahrenberg, Kim Guldstrand Larsen, Nicolas Markey, Jirí Srba: Infinite Runs in Weighted Timed Automata with Energy Constraints. FORMATS 2008: 33-47

Alexandre David, Kim Guldstrand Larsen, Shuhao Li, Brian Nielsen: A Game-Theoretic Approach to Real-Time System Testing. DATE 2008: 486-491

Franck Cassez, Alexandre David, Kim Guldstrand Larsen, Didier Lime, Jean-François Raskin: Timed Control with Observation Based and Stuttering Invariant Strategies. ATVA 2007: 192-206

Adam Antonik, Michael Huth,Kim G. Larsen, Ulrik Nyman, Andrzej Wąsowski.: 20 Years of Modal and Mixed Specifications. In Concurrency Column of Bulletin of EATCS no 95, 2008.

Alexandre David, Larsen, Kim Guldstrand Larsen, Shuhao Li, Brian Nielsen: Timed Testing with Partial Observability. Under submission.

John Knudsen, Anders P. Ravn, Arne Skou: Design Verification Patterns. Formal Methods and Hybrid Real-Time Systems 2007: 399-413.

Zhenbang Chen, Zhiming Liu, Volker Stolz, Lu Yang, Anders P. Ravn: A Refinement Driven Component-Based Design. ICECCS 2007: 277-289

Thomas Bøgholm, Henrik Kragh-Hansen, Petur Olsen, Bent Thomsen, Kim G. Larsen: Model-Based Schedulability Analysis of Safety Critical Hard Real-Time Java Programs. JTRES 2008: Proceedings of the 6th International Workshop on Java Technologies for Real-Time and Embedded Systems, p. 106—114, 2008.

Claus Thrane, Uffe Sørense. Slicing for UPPAAL. 2008 Annual IEEE Conference. Aalborg, Denmark, 2008. Best Student Paper Award.

Jirí Srba: Comparing the Expressiveness of Timed Automata and Timed Extensions of Petri Nets. FORMATS 2008: 15-32

Petr Jancar, Jirí Srba: Undecidability of bisimilarity by defender's forcing. J. ACM 55(1): (2008)

Twente

L. de Alfaro and M. Faella and M.I.A. Stoelinga: Linear and Branching System Metrics. IEEE Transactions on Software Engineering, 2008 (to appear)

H. Bohnenkamp and M.I.A. Stoelinga: Quantitative Testing. EMSOFT 08. 8th ACM & IEEE Conference on Embedded Software, ACM, 2008

Luca De Alfaro and Rupak Majumdar and Vishwanath Raman and Marielle Stoelinga: Game Refinement Relations and Metrics. Logical Methods in Computer Science

H. Boudali and P. Crouzen and B.R.H.M. Haverkort and M. Kuntz and M.I.A. Stoelinga: Architectural dependability evaluation with Arcade. Proceedings of the 38th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN2008). IEEE Society, 2008.

H. Boudali and P. Crouzen and B.R.H.M. Haverkort and M. Kuntz and M.I.A. Stoelinga: Arcade - A Formal, Extensible, Model-based Dependability Evaluation Framework. Proceedings of the 13th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS). 2008.

H. Boudali and P.Crouzen and and B.R.H.M. Haverkort and M. Kuntz and M.I.A. Stoelinga: Rich Interfaces for Dependability: Compositional Methods for Dynamic Fault Trees and Arcade models. 2008. Proceedings of the 2nd International Workshop on Foundations of Interface theories (FIT'08).

M.Faella and A.Legay and M.I.A. Stoelinga: Model checking Quantitative Linear Time Logic. 2007. Proceedings of the Sixth Workshop on Quantitative Aspects of Programming Languages (QAPL'08).

L. Cheug and M. I. A. Stoelinga and F. W. Vaandrager: A testing scenario for probabilistic processes. Journal of the ACM. Volume 54, no 6, ACM, 2007.

CVF

Pierre Ganty, Gilles Geeraerts, Jean-François Raskin, Laurent Van Begin. Méthodes algorithmiques pour l'analyse des réseaux de Petri. To appear in Journal Techniques et sciences informatiques. 2009.

M. De Wulf, L. Doyen, N. Markey, and J.F. Raskin. Robust Safety of Timed Automata. To appear in Formal Methods in System Design, 2008.

Martin De Wulf, Laurent Doyen, Nicolas Maquet and Jean-François Raskin. LTL Satisfiability, Alternating Büchi Automata Emptiness, and Model-Checking with Alaska. To appear in ATVA'08. 6 pages. 2008.

Jean-François Raskin and Frédéric Servais. Visibly Pushdown Transducers. In ICALP'08. 386-397. 2008.

Pierre Ganty, Jean-François Raskin, Laurent Van Begin. From Many Places to Few: Automatic Abstraction Refinement for Petri Nets. Invited extended version. To appear in Journal of Fundamenta Informaticae. 28 pages. 2008.

Laurent Doyen, Tom Henzinger, Jean-François Raskin. An equivalence relation for Markov Chains. Invited paper. In International Journal of Foundations of Computer Science,19(3):549-563, 2008.

Martin De Wulf, Laurent Doyen, Nicolas Maquet and Jean-François Raskin. Antichains: Alternative Algorithms for LTL Satisfiability and Model-Checking. In TACAS'08, LNCS, Springer, 63-77, 2008.

Véronique Bruyère, Emmanuel Dal'ollio, and Jean-François Raskin. Durations and Parametric Model-Checking in Timed Automata. In Transactions on Computational Logic,9(2):1-23, ACM press, 2008.

Bernard Boigelot, Julien Brusten and Véronique Bruyère. On the Sets of Real Numbers Recognized by Finite Automata in Multiple Bases. In ICALP'08, Lecture Notes in Computer Science, 5126, Springer, pp. 112--123, 2008.

Axel Legay: T(O)RMC: A Tool for (omega)-Regular Model Checking. CAV 2008: 548-551
François Cantin, Axel Legay, Pierre Wolper: Computing Convex Hulls by Automata Iteration. CIAA 2008: 112-121.

Axel Legay, Andrzej S. Murawski, Joël Ouaknine, James Worrell: On Automated Verification of Probabilistic Programs. TACAS 2008: 173-187

LSV

C. Baier, N. Bertrand and Ph. Schnoebelen. Verifying nondeterministic probabilistic channel systems against omega-regular linear-time properties. ACM Transactions on Computational Logic 9(1), 2007.

P. Chambart and Ph. Schnoebelen. Mixing Lossy and Perfect Fifo Channels. In Proceedings of the 19th International Conference on Concurrency Theory (CONCUR'08), Toronto, Canada, August 2008, LNCS 5201, pages 340-355. Springer.

P. Chambart and Ph. Schnoebelen. The Ordinal Recursive Complexity of Lossy Channel Systems. In Proceedings of the 23rd Annual IEEE Symposium on Logic in Computer Science (LICS'08), Pittsburgh, PA, USA, June 2008, pages 205-216. IEEE Computer Society Press.

P. Chambart and Ph. Schnoebelen. The omega-Regular Post Embedding Problem. In Proceedings of the 11th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'08), Budapest, Hungary, March-April 2008, LNCS 4962, pages 97-111. Springer.

P. Bouyer, N. Markey, J. Ouaknine, Ph. Schnoebelen and J. Worrell. OnTermination for Faulty Channel Machines. In Proceedings of the 25th Annual Symposium on Theoretical Aspects of Computer Science (STACS'08), Bordeaux, France, February 2008, pages 121-132.

P. Chambart and Ph. Schnoebelen. Post Embedding Problem is not Primitive Recursive, with Applications to Channel Systems. In Proceedings of the 27th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'07), New Delhi, India, December 2007, LNCS 4855, pages 265-276. Springer.

P. Bouyer, N. Markey, J. Ouaknine and J. Worrell. On Expressiveness and Complexity in Real-time Model Checking. In Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP'08) - Part II, Reykjavik, Iceland, July 2008, LNCS 5126, pages 124-135. Springer.

F. Laroussinie, N. Markey and G. Oreiby. On the Expressiveness and Complexity of ATL. Logical Methods in Computer Science 4(2:7), 2008.

P. Bouyer, N. Markey and P.-A. Reynier. Robust Analysis of Timed Automata via Channel Machines. In Proceedings of the 11th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'08), Budapest, Hungary, March-April 2008, LNCS 4962, pages 157-171. Springer.

P. Bouyer, S. Haddad and P.-A. Reynier. Timed Petri Nets and Timed Automata: On the Discriminating Power of Zeno Sequences. Information and Computation 206(1), pages 73-107, 2008.

INRIA/Rennes
J. Komenda, J. van Schuppen, B. Gaudin, H. Marchand, Supervisory Control of Modular Systems with Global Specification Languages, Automatica, 44:1127-1134, 2008.

C. Constant, T. Jeron, H. Marchand, V. Rusu, Validation of Reactive Systems, in Modeling and Verification of Real-TIME Systems - Formalisms and software Tools, S. Merz, N. Navet (eds.), Chapter 2, Pages 51-76, HermÃ¨s Science, January 2008.

Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, Marcus Groesser, Almost-Sure Model Checking of Infinite Paths in One-Clock Timed Automata, in Proceedings of the 23rd Annual IEEE Symposium on Logic in Computer Science (LICS'08), Pittsburgh, PA, USA, June 2008.

J. Dubreil, Ph. Darondeau, H. Marchand, Opacity Enforcing Control Synthesis, in Workshop on Discrete Event Systems, WODES'08, Gothenburg, Sweden, March 2008.

T. Jeron, H. Marchand, S. Genc, S. Lafortune, Predictability of Sequence Patterns in Discrete Event Systems, in IFAC World Congress, Seoul, Korea, July 2008.

J. Dubreil, T. Jeron, H. Marchand, Monitoring Information flow by Diagnosis Techniques, Research Report IRISA, No 1901, August 2008.

T. Le Gall, Abstract Lattices for the Verification of Systems with Queues and Stacks, PhD Thesis Universitat de Rennes 1, July 2008.

VERIMAG
Saddek Bensalem, Doron Peled, Hongyang Qu and Stavros Tripakis. Automatic Generation of Path Conditions for Concurrent Timed Systems. In Theoritical Computer Science, Volume 404, number 3, 28 Septembre 2008.

Ananda Basu, Matthieu Gallien, Charles Lesire, Thanh-Hung Nguyen, Saddek Bensalem, Felix Ingrand and Joseph Sifakis. Incremental Component-Based Construction and Verfication of a Robotic System. ECAI 2008 The 18th European Conference on Artificial Intelligence, Patras, Greece, July 21 - 25, 2008.

Saddek Bensalem, Felix Ingrand and Joseph Sifakis. Autonomous Robot Software Design Challenge 6th IARP/IEEE-RAS/EURON, Joint International Workshop on Technical Challenge for Dependable Robots in Human Environments, Pasadena, USA, May 17-18, 2008.

Saddek Bensalem, Moez Krichen and Stavros Tripakis. Generating Analog-Clock Real-Time Testers Using Action Refinement Techniques. In the International Conference ROGICS 2008, Mahdia, Tunisia.

Saddek Bensalem, Moez Krichen and Stavros Tripakis. State Identification Problems for Input/Output Transition Systems. In WODES'08, the 9th international Workshop on Discrete Event Systems, May 28-30, 2008, Goteborg, Sweden.

Saddek. Bensalem, Marius. Bozga, Matthieu. Gallien, Felix. Ingrand, Moez. Krichen and Stavros Tripakis. Automatic Generation of Observers for the Dala Robot with TTG. In the International Conference CISA 2008, Annaba, Algeria.

Matthieu Gallien, Fahmi Gargouri, Imen Kahloul, Moez Krichen, Thanh-Hung Nguyen, Saddek Bensalem, Felix Ingrand. D'une approche modulaire a une approche orientee composant pour

le developpement de systemes autonomes : defis et principes. In the 3rd Workhosp CAR 2008, Bourges, France.

Ananda Basu, Matthieu Gallien, Charles Lesire, Thanh-Hung Nguyen, Saddek Bensalem, Felix Ingrand and Joseph Sifakis. Incremental Component-Based Construction and Verification of a Robotic System. International Workshop on Current Software frameworks in Cognitive Robotics integrating different computational paradigms, Sept. 22nd 2008, Nice, France.

S. Yovine, I. Assayad, F.-X. Defaut, M. Zanconi, A. Basu. A formal approach to derivation of concurrent implementations in software product lines. To appear in Process Algebra for Parallel and Distributed Processing, M. Alexander & W. Gardner, eds., Chapman and Hall/CRC Press, Taylor and Francis Group LLC, 2008.

D. Monteverde, A. Olivero, S. Yovine, V. Braberman. VTS-based Specification and Verification of Behavioral Properties of AADL Models. In: Model Based Architecting and Construction of Embedded Systems (ACES-MB 2008), Toulouse, France, September 29, 2008.

V. Braberman, F. Fernandez, D. Garbervetsky, S. Yovine. Parametric Prediction of Heap Memory Requirements. ISMM'08, June 7-8, 2008, Tucson, Arizona, USA. ACM 2008.

P. Habermehl, R. Iosif and T. Vojnar. What else is decidable about integer arrays? In Proc. FoSSaCS 2008, LNCS, pp. 474-489

P. Habermehl, R. Iosif and T. Vojnar. A Logic of Singly Indexed Arrays. In Proc. LPAR 2008, to appear.

A. Degorre, O. Maler, On Scheduling Policies for Streams of Structured Jobs, FORMATS'08, 2008

Goran Frehse. A Counterexample-Guided Approach to Parameter Synthesis for Linear Hybrid Automata. In HSCC'08, 2008

O. Maler, A. Pnueli, D. Nickovic, Checking Temporal Properties of Discrete, Timed and Continuous Behaviors, Pillars of Computer Science, Springer, 2008

Uppsala

Nan Guan, Wang Yi, Zonghua Gu and Ge Yu. New Schedulability Test Conditions for Non-Preemptive Scheduling on Multiprocessor Platforms. Accepted by the 29th IEEE Real-Time Systems Symposium, Barcelona.

Bengt Jonsson, Simon Perathoner, Lothar Thiele, and Wang Yi. Cyclic dependencies in modular performance analysis. Accepted by the 8th International Conference on Embedded Software, Atlanta, USA, 2008.

John Håkansson, Jan Carlson, Aurelien Monot, Paul Pettersson and Davor Slutej. Component-Based Design and Analysis of Embedded Systems with UPPAAL PORT. 6th International Symposium on Automated Technology for Verification and Analysis, Springer-Verlag, Seoul, South Korea, October, 2008.

Simon Tschirner, Liang Xuedong and Wang Yi. Model-Based Validation of QoS Properties of Biomedical Sensor Networks. Accepted by the 8th International Conference on Embedded Software, Atlanta, USA, 2008.

Parosh Abdulla, Pavel Krcal and Wang Yi .R-automata. In the proceedings of the 19th International Conference on Concurrency Theory, Toronto, Canada, 2008. Lecture Notes in Computer Science, Volume 5021, pages: 67-81.

Nan Guan, Zonghua Gu, Wang Yi and Ge Yu. Improving Scalability of Model-Checking for Minimizing Buffer Requirements of Synchronous Dataflow Graphs. Regular paper accepted by the 14th Asia and South Pacific Design Automation Conference, Jan. 19-22 2009. Yokohama, Japan.

Parosh Abdulla, Pavel Krcal and Wang Yi. R-automata with value copying. In the proceedings of 10th International Workshop on Verification of Infinite-State Systems, Toronto, Canada, 23rd of August 2008 (to appear in Electronic Notes in Theoretical Computer Science).

Elena Fersman, Pavel Krcal, Paul Pettersson and Wang Yi. Task Automata: Schedulability, Decidability and Undecidability. In Journal: Information and Computation, 205(8), 2007. pages: 1149-1172.

EPFL

Krishnendu Chatterjee, Thomas A. Henzinger, and Vinayak Prabhu, Timed parity games: Complexity and robustness, Proceedings of the Sixth International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS), Lecture Notes in Computer Science, Springer, 2008.

Krishnendu Chatterjee, Laurent Doyen, and Thomas A. Henzinger, Quantitative languages, Proceedings of the 17th International Conference on Computer Science Logic (CSL), Lecture Notes in Computer Science, Springer, 2008.

Krishnendu Chatterjee, Thomas A. Henzinger, and Vinayak S. Prabhu, Trading infinite memory for uniform randomness in timed games, Proceedings of the 11th International Workshop on Hybrid Systems: Computation and Control (HSCC), Lecture Notes in Computer Science 4981, Springer, 2008, pp. 87-100.

Krishnendu Chatterjee, Rupak Majumdar, and Thomas A. Henzinger, Controller synthesis with budget constraints, Proceedings of the 11th International Workshop on Hybrid Systems: Computation and Control (HSCC), Lecture Notes in Computer Science 4981, Springer, 2008, pp. 72-86.

Krishnendu Chatterjee, Koushik Sen, and Thomas A. Henzinger, Model checking omega-regular properties of interval Markov chains, Proceedings of the 11th International Conference on Foundations of Software Science and Computational Structures (FOSSACS), Lecture Notes in Computer Science 4962, Springer, 2008, pp. 302-317.

Krishnendu Chatterjee, Rupak Majumdar, and Thomas A. Henzinger,Stochastic limit-average games are in EXPTIME, International Journal of Game Theory, 2008.

Krishnendu Chatterjee and Thomas A. Henzinger, Reduction of stochastic parity to stochastic mean-payoff games, Information Processing Letters 106:1-7, 2008.

Krishnendu Chatterjee and Thomas A. Henzinger, Value iteration, in 25 Years of Model Checking, Lecture Notes in Computer Science, Springer, 2008.

OFFIS

V. Sofronie-Stokkermans and C. Ihlemann. Automated reasoning in some local extensions of ordered structures. Journal of Multiple-Valued Logic, 2007. Accepted for publication; 17 pages.

W. Damm, S. Disch, H. Hungar, S. Jacobs, J. Pang, F. Pigorsch, C. Scholl, U. Waldmann, and B. Wirtz. Exact state set representations in the verification of linear hybrid systems with large

discrete state-space. In Proceedings of the 5th Symposium on Automated Technology for Verification and Analysis, Lecture Notes in Computer Science. Springer, 2007.

S. Jacobs and V. Sofronie-Stokkermans. Applications of hierarchical reasoning in the verification of complex systems. Electronic Notes in Theoretical Computer Science,174(8):pp. 39–54, 2007.

Marc Segelken. Abstraction and counterexample-guided construction of omega- automata for model checking of step-discrete linear hybrid models. In Werner Damm and Holger Hermanns, eds., Proceedings of the 19th International Conference on Computer Aided Verification – CAV 2007, vol. 4590 of LNCS. Springer, 2007.

V. Sofronie-Stokkermans. Sheaves and geometric logic and applications to modular verification of complex systems. Electronic Notes in Theoretical Computer Science, p. 25 p., 2007. Accepted for publication.

Brno

J. Barnat and L. Brim and S. Edelkamp and D. Sulewski and P. Simecek: Can Flash Memory Help in Model Checking, 13th International Workshop on Formal Methods for Industrial Critical Systems (FMICS 2008), 2008, 159--174.

J. Barnat and L. Brim and P. Rockai: DiVinE Multi-Core -- A Parallel LTL Model-Checker, ATVA 2008, to appear., 2008.

J. Barnat and L. Brim and I. Cerna and M. Ceska and J. Tumova: Local Quantitative LTL Model Checking, 13th International Workshop on Formal Methods for Industrial Critical Systems (FMICS 2008), 2008, 63--78.

J. Barnat and L. Brim and I. Cerna and S. Drazan and D. Safranek: Parallel Model Checking Large-Scale Genetic Regulatory Networks with DiVinE. ENTCS, volume 194(3), 2008, 35--50.

J. Barnat and L. Brim and P. Simecek and M. Weber: Revisiting Resistance Speeds Up I/O-Efficient LTL Model Checking, Tools and Algorithms for the Construction and Analysis of Systems (TACAS)., Springer, 2008, volume 4963 of LNCS, 48-62.

J. Barnat and P. Rockai: Shared Hash Tables in Parallel Model Checking. ENTCS, volume 198(1), 2008, 79—91.

L. Brim and J.Barnat: Squeeze All the Power Out of Your Hardware to Verify Your Software! ISOLA 2008, Springer Verlag, 2008, volume 17 of CCIS, 604-618.

2.4.3 Interaction and Building Excellence between Partners

Together with Edmund M. Clarke and Allen Emerson, Joseph Sifakis (director of ARTIST2) received the 2007 Turing Award, widely considered the most prestigious award in computing, for their original and continuing research on model checking.

Kim Larsen was awarded Doctor Honoris Causa at ENS Cachan acknowledging his regular collaboration with LSV. Kim Larsen also spent a month as an invited professor at LSV.

Patricia Bouyer and Nicolas Markey have ongoing collaborations with Thomas Brihaye (CFV). A PhD student at LSV, Arnaud da Costa, is currently spending 7 months in the group of Thomas Brihaye.

Patricia Bouyer has an active collaboration with Nathalie Bertrand (IRISA).

EPFL worked with the Universite Libre de Bruxelles on algorithms for establishing equivalences between probabilistic systems, and on the minimum-time reachability problem in timed games.

From Aalbrog to CFV (Brussels): one week visit of Prof. Kim Larsen to the team of Prof. JF Raskin.

From CFV (Brussels) to EPFL (Henzinger): Dr. Laurent Doyen formerly in CFV is post-doct at EPFL.

From CFV (Brussels) to EPFL (Henzinger): several visits during 2007-2008 by Prof. JF Raskin.

From EPFL (Henzinger) to CFV (Brussels): several visits during 2007-2008 by Dr. L Doyen.

From CFV (Brussels) to INRIA-IRISA, three months visit of Prof. T. Massart and G. Kaylon to Dr T. Jéron at IRISA Rennes.

From CFV (Mons) to LSV: several visits during 2007-2008 by Thomas Brihaye.

From INRIA-IRISA and LSV to CFV (Mons): one week visit of Nathalie Bertrand (INRIA-IRISA), Patricia Bouyer (LSV) and Nicolas Markey (LSV).

From Nijmegen to Inria Rennes: one week visit of Jan Tretmans to Irisa for participation to the summer school EJCP.

From CFV (Brussels) to Inria Rennes: two month visit of Gabriel Kalyon and one month visit of Thierry Massart.

From Irisa to CFV (Brussels) one week visit of T. Legall to ULB followed by post-doc started in September 2008.

In collaboration with LSV and Univ. Dresden, INRIA/Rennrs studied the topologic and probabilistic semantics of timed-automata and the model of probabilistic Büchi automata.

Through various research visits, Twente has interacted with Frits Vaandrager from University of Nijmegen

Through various research visits, Twente has interacted with Henrik Bohnenkamp from RWTH Aachen (affliated partner) and Pepijn Crouzen from Saarland University, Saarbruecken (affiliated partner).

M. Stoelinga (Twente) took part as a member in the PhD committee of Ulrik Nyman (Aalborg University) and Olga Grinchstein (University of Uppsala).

Ghassan Oreiby wil after his position as PhD student at LSV go to Aalborg University for a post doc position starting November 1, 2008.

Organization and hosting of the International Workshop FORMATS in Salzburg (in connection with the ESWeek), October 2007.

Organization and hosting of the International Workshop FORMATS in Saint Malo (in collaboration with QEST), September 2008.

2.4.4 Joint Publications Resulting from these Achievements

LSV & CVF

M. De Wulf, L. Doyen, N. Markey and J.-F. Raskin. Robust Safety of Timed Automata. Formal Methods in System Design, 2008. To appear.

P. Bouyer, Th. Brihaye, V. BruyÃ¨re and J.-F. Raskin. On the optimal reachability problem on weighted timed automata. Formal Methods in System Design 31(2), pages 135-175, 2007.

N. Bertrand, P. Bouyer, Th. Brihaye and N. Markey. Quantitative Model-Checking of One-Clock Timed Automata under Probabilistic Semantics. In Proceedings of the 5th International Conference on Quantitative Evaluation of Systems (QEST'08), Saint Malo, France, September 2008, pages 55-64. IEEE Computer Society Press.

C. Baier, N. Bertrand, P. Bouyer, Th. Brihaye and M. GrÃ¶ÃŸer.Almost-Sure Model Checking of Infinite Paths in One-Clock Timed Automata. In Proceedings of the 23rd Annual IEEE Symposium on Logic in Computer Science (LICS'08), Pittsburgh, PA, USA, June 2008, pages 217-226. IEEE Computer Society Press.

C. Baier, N. Bertrand, P. Bouyer, Th. Brihaye and M. GrÃ¶ÃŸer. Probabilistic and Topological Semantics for Timed Automata. InProceedings of the 27th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'07), New Delhi, India, December 2007, LNCS 4855, pages 179-191. Springer.

Th. Brihaye, M. Ghannem, N. Markey and L. Rieg. Good friends are hard to find! In Proceedings of the 15th International Symposium on Temporal Representation and Reasoning (TIME'08), MontrÃ©al, Canada, June 2008, pages 32-40. IEEE Computer Society Press.

Th. Brihaye, F. Laroussinie, N. Markey and G. Oreiby. Timed Concurrent Game Structures. In Proceedings of the 18th International Conference on Concurrency Theory (CONCUR'07), Lisbon, Portugal, September 2007, LNCS 4703, pages 445-459. Springer.

P. Bouyer, Th. Brihaye, M. Jurdzinski, R. Lazic and M. Rutkowski. Average-Price and Reachability-Price Games on Hybrid Automata with Strong Resets. In Proceedings of the 6th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'08), Saint-Malo, France, September 2008, LNCS 5215, pages 63-77. Springer.

Aalborg & LSV

P. Bouyer, U. Fahrenberg, K. G. Larsen, N. Markey and J. Srba. Infinite Runs in Weighted Timed Automata with Energy Constraints. In Proceedings of the 6th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'08), Saint-Malo, France, September 2008, LNCS 5215, pages 33-47. Springer.

P. Bouyer, K. G. Larsen and N. Markey. Model Checking One-clock Priced Timed Automata. Logical Methods in Computer Science 4(2:9), 2008.

P. Bouyer, E. Brinksma and K. G. Larsen. Optimal Infinite Scheduling for Multi-Priced Timed Automata. Formal Methods in System Design 32(1), pages 2-23, 2008.

EPFL & CFV

Laurent Doyen, Thomas A. Henzinger, and Jean-Francois Raskin, Equivalence of labeled Markov chains, International Journal of Foundations of Computer Science 19:549--563, 2008.

Thomas Brihaye, Thomas A. Henzinger, Vinayak S. Prabhu, and Jean-Francois Raskin, Minimum-time reachability in timed games, Proceedings of the 34th International Colloquium on Automata, Languages, and Programming (ICALP), Lecture Notes in Computer Science 4596, Springer, 2007, pp. 825-837.

Martin De Wulf, Laurent Doyen, Nicolas Maquet and Jean-François Raskin. Antichains: Alternative Algorithms for LTL Satisfiability and Model-Checking. In TACAS'08, LNCS, Springer, 63-77, 2008.

Laurent Doyen, Tom Henzinger, Jean-François Raskin. An equivalence relation for Markov Chains. Invited paper. In International Journal of Foundations of Computer Science,19(3):549-563, 2008.

Franck Cassez, Alexandre David, Kim Larsen, Didier Lime and Jean-François Raskin. Timed Control with Observation Based and Stuttering Invariant Strategies}. In ATVA'07, Lecture Notes in Computer Science, 4762, pp. 192--206, Springer, 2007.

INRIA & LSV & CVF

C. Baier, N. Bertrand, P. Bouyer, Th. Brihaye, M. Groesser, Probabilistic and Topological Semantics for Timed Automata, in Proceedings of the 27th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'07), V. Arvind, Sanjiva Prasad (eds.), Volume 4855, New Delhi, India, December 2007.

Christel Baier, Nathalie Bertrand, Marcus Groesser, On Decision Problems for Probabilistic BÃ¼chi Automata, in Proceedings of the 11th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'08), Budapest, Hungary, March 2008.

N. Bertrand, P. Bouyer, Th. Brihaye, N. Markey, Quantitative Model-Checking of One-Clock Timed Automata under Probabilistic Semantics, in Proceedings of the 5th International Conference on the Quantitative Evaluation of SysTems (QEST'08), Saint Malo, France, September 2008.

2.4.5 Keynotes, Workshops, Tutorials

Keynotes and tutorials

Nicolas Markey. Timed Systems - Model Checking and Games, Invited tutorial, 8th School on Modelling and Verifying Parallel Processes (MOVEP'08), Nouan-le-Fuzelier, France, 2008.

Patricia Bouyer. Model-Checking Timed Temporal Logics. Invited talk at TFIT'08 (Taipei, Taiwan, March 2008)

Patricia Bouyer. Quantitative timed games. Invited tutorial at theGAME'08 annual meeting (Warsaw, Poland, September 2008)

Thomas A. Henzinger, Games in System Design and Verification, invited keynote lecture, Eighth International Conference on Logic and the Foundations of Game and Decision Theory (LOFT), Amsterdam, The Netherlands, July 2008.

Thomas A. Henzinger, Three Sources of Infinity in Computation: Nontermination, Real Time, and Probabilistic Choice, invited lecture, First International Conference on Infinity in Logic and Computation (ILC), Cape Town, South Africa, November 2007.

Jean-Francois Raskin. Invited Talk. ``Timed automata: verification, control and optimality". Artist 2 Summer School in China. Shanghai. July 2008.

Jean-Francois Raskin. Invited Talk. Fixpoint-based Abstraction Refinements, LABRI, U Bordeaux, France, June 12, 2008.

Jean-Francois Raskin. Invited Talk. "An Introduction to Games Played on Graphs", University of Luxembourg, May 26, 2008.

Jean-Francois Raskin. Invited Talk. ``Controller Synthesis using Lattice Theory". IEEE CDC2007. December 2007. New-Orleans, USA.

Jean-Francois Raskin. Invited Talk. "Fixpoint-based Abstraction Refinements", LIAFA, U Paris 7, France, October 8, 2007.

Thomas Brihaye. Invited Talk "Optimal reachability problem for weighted timed automata and games", October 2007, in the AlgoSyn meeting organized by the University of Aachen in Rolduc (Netherlands)

Bernard Boigelot. Invited Talk. A Generalization of Cobham's Theorem to Automata over Real Numbers, LABRI, U Bordeaux, France, January 2008.

Bernard Boigelot. Invited Talk. A Generalization of Cobham's Theorem to Automata over Real Numbers, LIAFA, U Paris 7, France, January 2008.

Bernard Boigelot. Invited Talk. On the Sets of Real Numbers Recognized by Finite Automata in Multiple Bases, Dagstuhl Seminar ``Beyond the Finite", Dagstuhl, Germany, April 2008.

Bernard Boigelot. Invited Talk. Automata-based Representations of Arithmetic Sets, Automata and Verification Workshop, Mons, Belgium, August 2008.

Veronique Bruyere and Jean-Francois Raskin. Organization of the workshop "Automata and Verification", University of Mons-Hainaut, Belgium, August 25-26, 2008.

T. Jeron, invited talk at SBMF 2008, Symbolic model-based test selection, In Brazilian Symposium on Formal Methods (SBMF 2008), Salvador, Bahia, Brazil, August 2008, to appear in ENTCS.

Kim G. Larsen. 'Timing and Performance Analysis: Static Analysis versus Model Checking'. Invited Talk on the Honoris Causa to Professor Dr. Reinhard Wilhelm from RWTH Aachen. Germany. October 24, 2008.

Kim G. Larsen. Model-driven Test and Verification of Real-Time and Embedded Systems. Test Conference, Aalborg University. Denmark. October 20, 2008.

Kim G. Larsen: Verification, Performance Analysis, and Controller Synthesis for Real-Time Systems. Invited talk. Marktoberdorf Summerschool. Marktoberdorf, Germany. August 5-16, 2008.

Kim G. Larsen. Quantitative Verification and Synthesis for Embedded Systems. Invited Talk. ARTIST2 Summer School Autrans (near Grenoble), France. September 8-12, 2008.

Ed Brinksma. Quantitative Testing Theory. Invited Talk. ARTIST2 Summer School Autrans (near Grenoble), France. September 8-12, 2008.

Kim G. Larsen. Priced Timed Automata and Games. Automata and Verification Workshop University of Mons-Hainaut. Mons, Belgium. August 25, 26, 2008.

Kim G. Larsen. Modeling, Verification and Synthesis of Timed Systems. Invited Talk at The Centre for Interdisciplinary Computational and Dynamical Analysis (CICADA) Launch Event. Manchester University, England. July 1, 2008.

Kim G. Larsen. Model-driven Testing of Real-Time and Embedded Systems. Invited Talk at Pan-European Conference Systematic Testing. Berlin, Germany. June 5, 2008.

Kim G. Larsen. Playing Games with Timed Interfaces. Invited Talk. Foundation for Interface Theory (FIT). Budapest, Hungary. April 5, 2008.

Kim G. Larsen. Model Checking Embedded and Real Time Systems. Invited Talk. 9th International Workshop on Discrete Event Systems (WODES). Gothenburgh, Sweden. May 28-30, 2008.

Kim G. Larsen. Validation, Performance Analysis and Synthesis of Embedded Systems. Invited Talk. 3rd intl Workshop on Systems Software Verification (SSV08). Sidney, Austrailia. February 25-27, 2008.

Kim G. Larsen. Performance analysis, scheduling and synthesis of embedded systems . Invited Talk. Final Workshop of Centre for Dependable Computing (CDC). Tallinn, Estonia. January 21-22, 2008.

Kim G. Larsen. Verification, optimization and synthesis for timed systems: from theory to tools. Invited talk given at the receipt of Dr Honoris Causa from LSV, ENS Cachan. Cachan, France. November 26, 2007.

Workshops

10th International Workshop on Verification of Infinite-State Systems (INFINITY'08)

15th International Symposium on Temporal Representation and Reasoning (TIME'08)

7th International Workshop on Parallel and Distributed Methods in verifiCation - PDMC 2008. Affiliated to ETAPS 2008, March 29-April 6, 2008.

Dagstuhl seminar on Distributed Verification and Grid Computing. 10.08.08 - 14.08.08, Seminar 08332, Organized by: Henri E. Bal (Vrije Universiteit Amsterdam, NL),Lubos Brim (Masaryk University, CZ), Martin Leucker (TU München, DE)

Co-organization of the summer school Movep (http://www.univ-orleans.fr/movep2008/) about modeling and verifying parallel processes in June 2008, partially funded by Artist 2.

RTSS08 track on Design and Verification of Embedded Real-Time Systems,
the 29th IEEE Real-Time Systems Symposium. Barcelona, Spain. November 30 - December 3, 2008. (organizer Wang Yi)

ARTIST summer school on embedded systems design, Su Zhou, China. Aug. 1-12, 2007. (co-organizer Wang Yi)

Foundations of Interface Technologies, FIT 2008. Satellite events of ETAPS 2008 and sponsored by ARTIST2. Budapest, Hungary. April 5, 2008.

The conference series CAV, Computer-Aided Verification, is a key conference for this cluster and with members of the cluster taking a leading role. As such the first conference in the series was held in Grenoble in 1989.
Later – in 1991 the conference was hosted in Aalborg. In 2007 Werner Damm (OFFIS) was co-chair for CAV 2007.
In 2009, Verimag will organise CAV, the Conference on Computer-Aided Verification in Grenoble, where it had all started in 1989. This will be the 21th in a series dedicated to the advancement of the theory and practice of computer-aided formal analysis methods for hardware and software systems. CAV considers it vital to continue its leadership in hardware verification, maintain its recent momentum in software verification, and consider new domains such as biological systems. http://www-cav2009.imag.fr/

3. Milestones, and Future Evolution Beyond the NoE

3.1 Milestones

Year3: **Development of algorithms and implementation of tools for optimal controller synthesis, robust model checking, coverage-based test selection and code generation.** Existing verification tools and test generation tools are more strongly connected, including stronger links between academic and industrial tools.

During the third year significant amount of research was carried out with respect to development of efficient algorithms for controller synthesis. In particular the work using the lattice-theoretic approach points to truly significant gains in performance for model-checking of finite state systems compared with existing implementations. For priced games the division between decidability and undecidability is now made more clear leaving only the case of models with 2 clocks open. For pure timed games efficient implementations of on-the-fly synthesis has been obtained.

Robust model checking (replacing simple boolean answers with quantitative verdicts) has been developed as well as methods for coverage-based test selection. The anticipated work on code generation was not been pursued, the main reason being that the STREPS proposal that would have funded this work for many cluster partners was not granted in the FP6 Call.

Link between academic and industrial tools has been demonstrated (in particular linking UPPAAL Tiga with Simulink.

Year4: **Completion of efficient tool components for controller synthesis. Initiation of work on property-preserving code generation and industrial applications. Development of generic framework using abstraction and compositionality for efficient analysis of quantitative models.** Emergence of a range of new powerful debugging and analysis techniques based on various combinations of testing and verification techniques.

Substantial research advancing state-of-the-art of controller synthesis has been made covering synthesis with budget constraints, partial observability, modular and distributed synthesis, cost-optimality, synthesis with respect to bounded response properties. The tool UPPAAL TIGA is now completely integrated with UPPAAL allowing timed game models to make the full use of discrete datastructures as well as user-defined datatypes and functions. By the end of the year an extension of UPPAAL TIGA supporing partial observability will be made available.

The tool-chain UPPAAL, UPPAAL-TIGA and Simulink has been applied on other industrial application, including synthesizing the controller for a hydraulic molding machine (provided by Hydac Electronic within the Quasimodo projects).

During the fourth year, a number analysis methods for a variety of quantitative models have been provided including models with reources (e.g. cost, energy, memory consumption, stack size, etc) as well as stochastic aspects. Methods supporting compositionality and component based development using rich interfaces (i.e. quantitative component specifications) have been provided and abstraction of quantitative models (both stochastic and hybrid) have provided the basis for efficient Counter-Example-Guided-Abstraction-Refinement (CEGAR) methods.

Novel debugging and analysis techniques have been provided including techniques based on heuristic and guided search, off-line test generation using games as well as symbolic model checking techniques (using BDDs and DBMs) to deal with data and time respectively.

The application of heuristic search for off-line test generation of test sequences has been applied in a commercial tool (V+) powerd by the UPPAAL verification engine and marketed by the company Scott/Tiger Validate. The method of the tool are now been applied by the company Novo Nordisk in systematic testing of medical devices within the Danish project DaNES.

Property-preserving code generation – and the related topic on robustness – has been subject to less activity than anticipated. However, with the Quasiomod project this topic will be pursued during the next years.

3.2 Indicators for Integration

A significant sign of integration is the succesful STREP proposals *Quasimodo* and *Multiform* in which several partners of the cluster take part and which will pursue several of the challenges identified by the activity. In *Quasimodo* emphasis is on providing tool plugins supporting analysis of quantitative models exhibiting both real-time and stochastic aspects. In the *Multiform* project emphasis is in providing an integrated tool environment envolving Simulink as well as a number of academic tools (e.g. UPPAAL and Phaver).

Also, the newly started ESF project GASICS involves a number of partners from ARTIST2 and will be have an emphasis on the foundation of game theory and controller synthesis as well as efficient tool support.

Finally , as indicated by the concrete list of interations reported in section 2.4.3 and the numerous joint publications listed in section 2.4.4 the partners have established long-term and extensive collaborations on all the topics covered by the activity.

3.3 Main Funding

< Should be copied and adapted from the Year 3 deliverable. >

- European Project Quasimodo. FP7 STREP. www.quasimodo.aau.dk/
- European Project Multiform. FP7 STREP.
- Danish Network of Embedded Systems, DaNES. Danish national project sponsored by the Danish Advanced Technology Foundation.
- MT-LAB, Danish national project sponsored by Villum-Kahn Rasmussen Foundation.
- MoDES, Danish national project sponsored by the Strategic Research Council.
- GASICS. European Project sponsored by European Science Foundation (ESF).
- French national project ARA DOTS
- Indo-French project P2R MODISTE-COVER
- French RNTL project Testec (Testing of real-time embedded control-commandsystems) with Lurpa (Ens Cachan), Inria (Rennes), I3S (Nice) , Labri (Bordeaux), EDF R&D, TNI Software.
- CREDO (http://www.cwi.nl/projects/credo/, for Uppsala), supported by EU, Modeling and analysis of evolutionary structures for distributed services.

- SAVE++ (http://www.mrtc.mdh.se/SAVE/, for Uppsala) supported by Swedish strategic research. Component Based Design of Safety Critical Vehicular Systems
- Modeling and verification of timed systems (for Uppsala) financed by the Swedish research council
- COMBEST (COMponent-Based Embedded Systems design Techniques) is an European research project funded by the European Community's (2008-2010) under the grant agreement number ST STREP 215543.
- GENESYS is an European research project funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under the grant agreement number FP7-213322.
- MARAE, is an industrial project on robust methods to develop autonomous systems (2008-2010), in collaboration with ASTRIUM (EADS) and LAAS. This project is funded by FNRAE ("Fondation Nationale pour la Recherche en Aéronautique et l'Espace")
- MAES is an ANR SSIA project in colaboration with LAAS and LIAFA (2005-2008).
- French RNTL AVERILES for the analysis and verification of embedded software systems with dynamic memory structures. www.lsv.ens-cachan.fr/rntl-averiles/
- IST PROSYD project for the design of a standard, integrated property-based paradigm for the design of electronic systems building upon the emerging standard property specification language PSL/Sugar. http://www.prosyd.org/
- IST project MULTIFORM, Integrated Multi-formalism Tool Support for the Design of networked Embedded Control Systems, http://www.multiform.bci.tu-dortmund.de/

3.4 Future Evolution Beyond the Artist2 NoE

The activities of this topic will be pursued beyond the end of the ARTIST2 NoE within the ARTIST Design Noe activity on *Modeling and Validation,* with a emphasis on

- modeling formalisms spanning the areas of computer science, control, hardware and networks covering all aspects of embedded systems.
- efficient means for analysis of models including simulation, testing, static analysis, model-checking, run-time verification, monitoring, diagnosability, controller synthesis.
- compositional methodologies allowing new complex systems to be assembled from already constructed and validated components.
- Realization of coherent tool chain

As already stated the newly started FP7 STREP projects *Quasimodo* and *Multiform* will provide for a, where several of the partners of the cluster may pursue the challenges identified with respect to testing and verification of embedded systems.

4. Internal Reviewers for this Deliverable

Contributions and internal review has been made by Bruno Bouyssounouse (UJF/Verimag), Kim G. Larsen (Aalborg) and Arne Skou (Aalborg).