Software Modeling and Verification

Staff

- Professors:

  Prof. Dr. Ir. Joost–Pieter Katoen PD
  Prof. em. Dr. Klaus Indermark
  `http://moves.rwth-aachen.de/`

- Secretary:

  Elke Ohlenforst

- Lecturer:

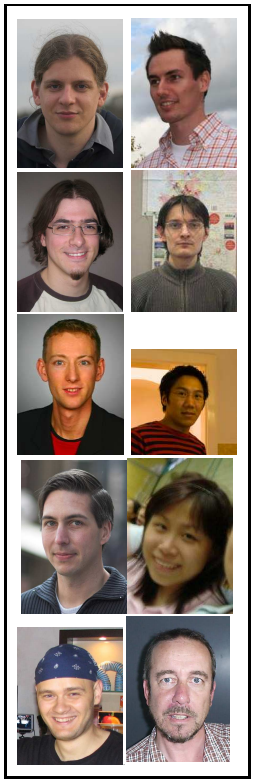
  Akademischer Oberrat Priv.–Doz. Dr. Thomas Noll

- Researchers:

  Dr. Erika Ábrahám (until 08/2007)
  Dr. Henrik Bohnenkamp
  Dr. David Jansen (from 01/2007 until 08/2007)
  Tingting Han, M.Sc. (funded by the NWO)
  Dipl.-Inform. Jonathan Heinen (from 05/2007)
  Dipl.-Inform. Carsten Kern
  Dipl.-Inform. Daniel Klink (DFG funding)
  Alexandru Mereacre, MSc (from 11/2007, DFG funding)
  Dipl.-Inform. Martin Neuhäußer (funded by the NWO)
  Viet Yen Nguyen, M.Sc. (from 02/2008, ESA funding)
  Dipl.-Inform. Stefan Rieger
  Dipl.-Inform. Haidi Yue (from 05/2008, DFG funding)
  Ivan Zapreev, M.Sc. (until 03/2008, funded by the NWO)

- Technical Staff:

  Arnd Gehrmann



- Diploma/Master Students:

  Lars Helge Haß
  Alexandru Mereacre

- Student Researchers:

  Yen Cao
  Falko Dulat
  Ulrich Hölscher
  Johanna Nellen
  Christina Jansen
  Denise Nimmerrichter
  Christian Lücking
  Maximilian Odenbrett
  David Piegdon
  Franziska Roloff
  Ulrich Schrempp
  Stefan Schulz
  Ulrich Schmidt-Goertz
  Benjamin Zimmermann

- Visiting Scientists:

  Varun Aggarwala (Indian Inst. of Techn. Guwahati, India)
  Dr. Suzana Andova (TU Eindhoven, NL)
  Miguel Andres (Radboud University Nijmegen, NL)
  Machiel van der Bijl (Univ. Twente, NL)
  Eric Bodden (McGill Univ, CA)
  Dr. Benedikt Bollig (LSV Cachan, F)
  Dr. Dragan Bosnacki (Technical Univ. Eindhoven, NL)
  Prof. Dr. Ed Brinksma (Emb. Systems Inst., Eindh., NL)
  Dr. Manuela Bujorianu (Uni Twente, NL)
  Dr. Lucia Cloth (University Twente, NL)
  Berteun Damman (University of Twente, NL)
  Wan Fokkink (Vrije Universiteit Amsterdam, NL)
  Dr. Sven Johr (Universität Saarbrücken, D)
  Marijn Jongerden (University of Twente, NL)
  Stephan Tobies (Europ. Microsoft Innov. Center, Aachen)
  Dr. Tomas Krilavicius (Vytautas Magnus Univ., Lithuania)
  PD Dr. Martin Leucker (TU Munich, D)
  Anne Remke (University of Twente, NL)
  Dr. Wolfram Schulte (Microsoft Res., Redmond USA)
  Jeremy Sproston (Univ. Torino, Italy)
  Mani Swaminathan (Uni Oldenburg, D)
  Dr. Hagen Voelzer (IBM Zurich, CH)
  Mikhail Volkov (Ural State University, Russia)
  Dr. Verena Wolf (EPFL, CH)

Overview

The years 2007 and 2008 have been extremely dynamic and active. First and foremost, we successfully acquired a project funded by the European Space Agency (ESA) which is aimed at system software co-engineering and focuses on the combination of performance and verification, an active research field at our chair. With the Italian research institute FBK and the French company Thales Alenia Space, we have two years to convince ESA that formal methods are pivotal to model and analyze both correctness and efficiency aspects of aerospace systems. A true and interesting challenge indeed. Several students are actively participating in this project.

Since the beginning of 2008, we are also involved in the EU project Quasimodo that focuses on quantitative aspects of embedded systems. Issues like resource usage, uncertainty, response times, and so forth, are central in this project. It covers modeling formalisms, model checking techniques, as well as formal testing techniques to assess and model these aspects at various levels of the software development process. The project is led by Aalborg University (Kim G. Larsen), and involves besides six academic partners three industrial partners that contribute with case studies to be tackled. A wonderful testbed to apply formal methods in practice!

Due to these projects, as well as our participation in the Research Training Group ALGOSYN and the excellence cluster UMIC, several new researchers joined our chair: Jonathan Heinen, Alexandru Mereacre, Viet Yen Nguyen, and Haidi Yue. Ivan Zapreev successfully defended his dissertation in March 2007 and joined the group of Jan van Schuppen (hybrid systems) at CWI, Amsterdam.

What else? Too much to all be mentioned. Let's pick some of the highlights. New courses have been set up, such as Testing of Reactive Systems, where the focus is on formal methods towards system testing, and Foundations of the UML, a course that attempts to pinpoint at semantical and analysis issues in notations such as sequence diagrams, statecharts and the OCL. The book "Principles of Model Checking", a joint (and longlasting) project with Christel Baier (TU Dresden) was completed and has been published by MIT Press. We were involved in the organization of several events, such as the Workshop on Automata and Logics (WAL'07) in 2007, an event that was organized as a birthday salute to colleague Wolfgang Thomas and that attracted more than 160 participants, a Dagstuhl seminar on quantitative methods for embedded systems (organized together with colleagues Boudewijn Haverkort (Twente) and Lothat Thiele (ETH Zurich)), a Quasimodo Workshop in Aachen, and a Lorentz Centre (Leiden) Workshop on the Validation of Stochastic Systems with more than 50 participants. Last, but definitely not at least, I mention the enormous productivity of the researchers at the chair. Various high-quality papers have been produced, and important advances have been achieved, both in theory as well as in tool development. It's a real pleasure and very stimulating to work with such an active and talented team!

At the time of reading of this annual report, the chair has been extended with a junior professorship on the topic of “Theory of Hybrid Systems”, an important field of research on the edge of mathematics, computer science, and various engineering disciplines. This position has been financed by the excellence initiative of the DFG and is occupied since October 2008 by Prof. Dr. Erika Ábrahám (formerly at the Research Centre at Julich). We warmly welcome Erika and are looking forward to a prosperous cooperation.

Joost-Pieter Katoen.

Research Projects

Smyle Modeling Approach

J.-P. Katoen, C. Kern, B. Bollig (ENS Cachan, F), M. Leucker (TU Munich)

(partially funded by the DAAD PROCOPE 2008 project)

The Smyle Modeling Approach (SMA) is a model-based software development methodology which is centered around Smyle, a dedicated learning procedure to support engineers to interactively obtain design models from requirements, characterized as either being desired (positive) or unwanted (negative) system behavior. The learning approach is complemented by scenario patterns where the engineer can specify clearly desired or unwanted behavior. This way, user interaction is reduced to the interesting scenarios limiting the design effort considerably. In SMA, the learning phase is complemented by an effective analysis phase that allows for detecting design flaws at an early design stage.

Learning Residual Finite-State Automata

J.-P. Katoen, C. Kern, B. Bollig (ENS Cachan, F),
P. Habermehl (ENS Cachan, Univ. Paris 7, F), M. Leucker (TU Munich)

(partially funded by the DAAD PROCOPE 2008 project)

The class of residual finite-state automata (RFSA) is a subclass of non-deterministic finite automata (NFA). It was shown by Denis et al. that for each regular language a unique RFSA (a so-called canonical RFSA) can be determined which may be exponentially more succinct than the corresponding minimal deterministic finite automaton (DFA) for this language. Together with the LSV (ENS Cachan) and the Technical University Munich, we developed a learning algorithm for inferring RFSA. It is an incremental learning algorithm and closely related to Angluin's learning algorithm L* for DFA. As we employ non-determinicity our approach allows for learning compact representations of regular languages. Many fields of application are available where the need for compact representations is by far more important than the determinicity of the model. Moreover, we developed a first prototype that implements our learning approach and showed its functionality in several small-size examples. However, larger size case studies are left for future work.

QUPES: Verification of Quantitative Properties of Embedded Software

T. Han, J.-P. Katoen, M. Neuhäußer

(funded by the NWO)

Embedded software typically executes on devices that, first and foremost, are not personal computers. Due to its embedded nature, its robustness is of prime importance, and timely reactions to stimuli from its – mostly physical – environment are essential. The aim of the QUPES project is to assess these quantitative aspects (e.g., timeliness and robustness) as an integral part of the embedded software validation phase.

To accomplish this, probabilistic model-checking techniques can be applied for models that are equipped with randomness and variants thereof which also exhibit nondeterminism. Based on efficient numerical methods and abstraction techniques, quantitative properties can be checked automatically even on large state space with millions of states using dedicated tools. Oppose to, amongst others, the essential feature of model checking, where evidences will be provided on a property refutation, counterexamples generation in probabilistic model checking is almost not developed. We provide the theoretical and algorithmic foundations for counterexample generation in probabilistic model checking, in particular for discrete-time Markov chains. One of the key principles is the casting of the concepts of strongest evidence and smallest counterexample as (variants of) shortest path problems. This enabled the use of efficient and well-studied graph algorithms for counterexample generation. These results can be extended to Markov chains with rewards, to Markov decision processes (MDPs), to LTL model checking, and have been recently been adopted in probabilistic counterexample-guided abstraction-refinement (CEGAR) techniques for MDPs as well as in counterexample generation for continuous-time Markov chains (CTMC) and cpCTL logic. Compact representation of a counterexample by regular expressions are also studied.

Further, compositional reasoning is a key strategy in analyzing complex systems as it allows the use of hierarchical and modular modeling formalisms like stochastic process algebras, stochastic activity networks or generalized stochastic Petri nets. Continuous-time Markov Decision processes (CTMDPs) are the nondeterministic counterpart of the aforementioned CTMCs and are well suited for compositional verification techniques. We define stochastic logics (like CSL) on CTMDPs and provide their measure-theoretic basis. Further, well-known equivalences like strong and weak bisimulation relations are adapted to CTMDPs which considerably reduce the state-space needed for quantitative analysis.

Dependable Global Computing

J.-P. Katoen, R. De Nicola (Univ. Florence, I), D. Latella (CNR-ISTI, I), M. Loreti (Univ. Florence, I), M. Massink (CNR-ISTI, I)

(funded by the DAAD and CNR-ISTI)

Due to their enormous size —networks typically consist of thousands or even millions of nodes— and their strong reliance on mobility and interaction, performance and dependability issues are of utmost importance for "network-aware computing". Spontaneous computer crashes may easily lead to failure of remote execution or process movement, while spurious network hick ups may cause loss of code fragments or unpredictable delays. The enormous magnitude of computing devices involved in global computing yields failure rates that no longer can be ignored. The presence of such random phenomena implies that correctness of global computing software and their privacy guarantees are no longer rigid notions like: "either it is safe or it is not" but have a less absolute nature, e.g.: "in 99.7% of the cases, privacy can be ensured". The intrinsic complexity of global computers, though, complicates the assessment of these issues severely. Systematic methods, techniques and tools, all based on solid mathematical foundations i.e., formal methods, are therefore needed to establish performance and dependability requirements and guarantees.

This project attempts to make a considerable step into this direction by extending a successful programming and specification formalism for global computing, KLAIM, with random delays, and by developing a novel stochastic spatial temporal logic as property specification language for performance and dependability guarantees.

MC=MC: Model Checking Infinite-State Markov Chains

J.-P. Katoen, I.S. Zapreev, B. Haverkort (Univ. Twente, NL), A. Remke (Univ. Twente, NL)

(funded by the NWO)

Model-based performance evaluation aims at forecasting system behaviour in a quantitative way, starting from an abstract system model. Due to the ever-increasing size and complexity of modern computer and communication systems, performance models that are directly amenable for a numerical solution are often generated from high-level modelling languages, based, e.g., on stochastic Petri nets or stochastic process algebras. For a significant class of systems, these models turn out to be infinite state,

and need to be analysed by specific techniques, such as matrix-geometric methods.

Recently, extensions to temporal logics have been developed to ease the specification of important measures-of-interest (like reponse times, or the probability to reach deadlines) over performance models, and logic-based verification algorithms have been integrated with numerical means to automatically check these properties. This novel approach is, however, still restricted to finite-state systems.

This project aims to establish a cross-fertilization between (i) performance evaluation techniques for infinite-state systems and (ii) logic-based model-checking algorithms for Markov chains. The goal is to develop algorithms and a prototype software tool for the specification and automated evaluation of performance measures for infinite-state Markov chains, and to apply these to case studies with realistic complexity. In particular, Markov chains with a regular structure will be investigated (Jackson QNs and Quasi-Birth-Death processes), and discrete event simulation techniques will be developed for model checking. These techniques are realized in the model checker MRMC.

System and Software Co-Engineering: Performance and Verification

A joint project together with the groups of Alessandro Cimatti (Fondazione Bruno Kessler, Centre for Scientific and Technological Research, Trento, Italy), and Xavier Olive (Thales Alenia Space, On Board Software Department, Cannes, France)

(This project is funded by the European Space Agency, ESA)

We develop a model-based approach to system-software co-engineering which is tailored to the specific characteristics of critical on-board systems for the space domain. The approach is supported by a System-Level Integrated Modeling (SLIM) Language in which engineers are provided with convenient ways to specify a.o. nominal hardware, as well as software operations, (probabilistic) faults and their propagation, error recovery and degraded modes of operation. This language is strongly based on AADL and its error annex which allows for the modeling of error behavior. A kernel of the SLIM language is equipped with a formal semantics that provides the interpretation of SLIM specifications in a precise and unambiguous manner. Systems are considered as a hierarchy of (hardware and software) components where components are defined by their type (interface) and implementation. Components communicate via ports allowing for message and continuous communication. The internal structure of a component implementation is specified by its decomposition into subcomponents, together with their HW/SW bindings and their interaction via connections over ports. Component behavior is described by a textual description of mode-transition diagrams. System reconfiguration is supported by mode-dependent presence of subcomponents and their

connections. Error behaviour is described by probabilistic finite state machines, where error delays may be governed by continuous random variables.

Correctness properties, safety guarantees, and performance and dependability requirements are specified using requirement specification patterns which act as parameterized “templates” to the engineers and thus offer a comprehensible and easy-to-use framework for requirement specification.

The properties are checked on the SLIM specification using formal analysis techniques such as model checking and probabilistic variants thereof. The precise character of these techniques and the SLIM semantics yield a trustworthy modeling and analysis framework for system and software engineers. The formal analysis is based on state-of-the-art model checking techniques such as bounded SAT-based and symbolic model checking, and extensions of model checking with numerical and simulative means to reason about quantitative requirements such as performance and dependability. The analysis facilities support, among others: automated derivation of dynamic (i.e., randomly timed) fault trees, FMEA tables, assessment of FDIR, and automated derivation of observability requirements.

The realization of an integrated platform tool set is currently under development. Evaluation of this novel approach to system-software co-engineering will take place by means of selected case studies representing critical on-board computer-based systems.

Verifying Pointer Programs with Unbounded Heap Structures

J. Heinen, J.-P. Katoen, Th. Noll, S. Rieger

The incorrect use of pointers is one of the most common sources of software errors. This especially applies to concurrent systems whose nondeterministic behavior rises additional challenges. Proving the correctness of concurrent pointer-manipulating programs with unbounded heap, let alone algorithmically, is a highly non-trivial task. This project attempts to develop automated verification techniques and accompanying tool support for concurrent programs with dynamic thread creation and memory allocation

that handle linked data structures which are potentially unbounded in their size. More concretely, two issues are addressed:

In a first phase, we concentrated on (possibly cyclic) singly-linked list data structures. We allow to express correctness properties of programs by combining a simple pointer logic for specifying heap properties with linear-time (LTL) operators for reasoning about system executions. To obtain a finitary representation of the system, we developed a technique which abstracts from non-interrupted sublists in the heap, resulting in a finite-state representation of the data space. In a second abstraction step, using an intermediate Petri-net representation, we also derive a finite-state representation of the control flow, which then allows us to employ standard LTL model checking techniques. Thus our approach stays within the realm of traditional (linear-time) model checking. This facilitates the usage of standard model checkers for validating temporal properties addressing absence of memory leaks, dereferencing of null pointers, dynamic creation of cells, and simple and position-dependent aliasing.

Next we extended our approach to analyze programs that handle more complex dynamic data structures. We developed a novel abstraction framework that employs graph grammars, more precisely context-free hyperedge replacement grammars, as an intuitive formalism for abstractly modeling dynamic data structures. The key idea is to use the replacement operations which are induced by the grammar rules in two directions. By a *backward* application of some rule, a subgraph of the heap can be condensed into a single nonterminal edge, thus obtaining an *abstraction* of the heap. By applying rules in *forward* direction, certain parts of the heap which have been abstracted before can be *concretized* again, which avoids the necessity for explicitly defining the effect of pointer-manipulating operations on abstracted parts of the heap. Altogether this method again allows to extend finite-state verification techniques to handle pointer-manipulating programs operating on complex dynamic data structures that are potentially unbounded in their size. We demonstrated how our framework can be employed for analysis and verification purposes by giving its instantiation for binary trees, and by applying this instantiation to the well-known Deutsch-Schorr-Waite traversal algorithm. Our approach is supported by a prototype tool, enabling the quick verification of essential properties such as heap invariants, completeness, and termination.

Equational Abstractions for Software Model Checking

M. Neuhäußer, Th. Noll, L. Haß, P. Tawiah

The combinatorial explosion of state spaces is the biggest challenge in applying model-checking methods to concurrent systems. The goal of this project is to develop a new state-space reduction technique that is tailored to system specifications in *Rewriting Logic*, a unified semantic framework for concurrency which is based on conditional

term rewriting modulo equational theories. The idea is to hide "unimportant" details of the system's behavior (such as internal computations) in the equations, and to represent only "interesting" state changes (such as communication operations) by explicit transitions. Our results show that this optimization can be implemented by transforming the Rewriting Logic specification, avoiding the intermediate construction of the full state space. The correctness of our technique can be established by proving that the original and the reduced system are weakly bisimilar.

The usability of this approach was demonstrated by applying it to the concurrent functional programming language Erlang, which is designed for implementing open, distributed telecommunication software. The inherent complexity and nondeterminacy of such systems impedes the use of validation methods which are purely based on testing. Therefore we developed a formalization of this language in the Rewriting Logic framework, employing equations for defining abstraction mappings on the state space of the system. This specification was implemented in the *Maude* system, and its model checker was employed to verify simple system properties.

Formal Models of Microcontroller Systems

Th. Noll, G. Herberich, B. Schlich, C. Weise

Embedded systems usually operate in uncertain environments, giving rise to a high degree of nondeterminism in the corresponding formal models. This, together with other effects, leads to the well-known *state explosion problem*, meaning that the models of those systems grow exponentially in size as the number of components increases. Careful handling of nondeterminism is therefore crucial for obtaining efficient tools for analysis and verification.

The goal of this project, carried out in close cooperation with the Embedded Software Laboratory of our department, is to develop formal computation models and state-space reduction techniques to tackle this problem. A first approach was taken by defining a general automata-based model for microcontrollers, taking into account both the hardware, the software, and the environment of the system. This model was used to prove the correctness of a particular abstraction method, called *delayed nondeterminism*, which resolves nondeterministic behavior only if and when this is required by the application code. More concretely, a simulation relation between the concrete and the abstract state space was established, thus showing the soundness of delayed nondeterminism with respect to "path-universal" verification logics such as ACTL and LTL.

Current efforts concentrate on extending the model to cover further abstraction techniques, and on the implementation of a tool component which automatically produces a state-space generator from the given microcontroller model.

Quasimodo

H. Bohnenkamp, J.-P. Katoen, D. Klink, H. Yue

Embedded systems are hidden computer components of many devices and appliances used in daily life: washing machines, air conditioners, cars, and GPS navigation systems, to name only a few. Embedded systems are highly complex, which poses a challenge for their design and implementation. In particular, such systems have to meet a multitude of quantitative constraints, such as available computation resources, power consumption, memory usage, communication bandwidth, arrival rates, timing constraints, QoS, availability, fault tolerance, etc.

Since January 2008, the MOVES group participates in the European research project "Quasimodo", funded by the European Commission under the IST framework programme 7 for Information and Communication Technology, ICT.

The objective of the Quasimodo project is to develop theory, techniques and tool components for handling quantitative (e.g. real-time, hybrid and stochastic) constraints in model-driven development of real-time embedded systems. Ultimate aim is to increase the competitiveness of European industrial companies which develop, implement and deploy embedded systems.

More specifically, the project aims are:

1. Improving the modelling of diverse quantitative aspects of embedded systems.
2. Providing a wide range of powerful techniques for analysing models with quantitative information and for establishing abstraction relations between them.
3. Generating predictable code from quantitative models.
4. Improving the overall quality of testing by using suitable quantitative models as the basis for generating sound and correct test cases.
5. Applying the techniques to real-life case-studies and disseminating the results to industry.

Project partners are universities, research institutes, and companies in Germany, The Netherlands, Denmark, Belgium, and France.

The MOVES Group is currently working on a case-study for a sensor-network gossiping protocol, which is posed by one of the industrial partners.

Model-Based Testing

H. Bohnenkamp, E. Brinksma (ESI, NL), M. Stoelinga (Uni Twente, NL)

Testing is one of the most natural, intuitive and effective methods to increase the reliability of software. Formal methods have been employed to analyse and systematise the testing idea in general, and to define notions of correctness of implementations w.r.t. specifications in particular. The ioco testing theory reasons about black-box conformance testing of software components. The test-case generation and execution algorithms of ioco have been implemented in TorX, a testing tool developed at the University of Twente.

We work on two topics in this area.

1. An extension to TorX to allow testing of real-time properties: real-time testing. Real-time testing means that the decisions whether an implementation under test has passed or failed a test is not only based on which outputs are observed, given a certain sequence of inputs, but also on when the outputs occur, given a certain sequence of inputs applied at predefined times. We use as input models non-deterministic safety timed automata.

2. In timed testing, reaching a verdict depends on time measurements. The imprecision of measurements can lead to false positives (test fails although the implmenation behaved correctly). We work on an extension of the ioco theory, where the verdicts are not binary (pass/fail) but of a quantitative nature: it is measured how close to the specified behaviour the implementation behaves.

The MoDeST Tool Environment

H. Bohnenkamp, J.-P. Katoen, H. Hermanns (Uni Saarland)

The specification language MoDeST covers a wide spectrum of modelling concepts, ranging from plain labelled transition systems to stochastic systems like Generalised Semi-Markov Decision Processes. MoDeST possesses a rigid, process-algebra style semantics, and yet provides modern and flexible specification constructs. MoDeST specifications constitute a coherent starting-point to analyse distinct system characteristics with various techniques, e.g., model checking to assess functional correctness and discrete-event simulation to establish the system's reliability. Analysis results thus refer to the *same* system specification, rather than to different (and potentially incompatible) specifications of system perspectives like in the UML.

The tool MOTOR (MoDeST Tool enviRonment) is aimed to provide the means to analyse and evaluate MoDeST specifications. The tool is written in the C++ programming language. The tool provides (i) interfacing capabilities for connection to existing tools for specific projected models, and (ii) also means for enhancement by *native* algorithms for analysis of (classes) of MoDeST specifications. In earlier work, MOTOR has been connected to MÖBIUS, a performance evaluation tool suite that has been developed at the University of Illinois at Urbana-Champaign, US. The MoDeST/Mobius tandem is currently used and constantly improved in the Quasimodo project case-studies.

Other Activities

J.-P. Katoen

- Member of the Steering Committee of ETAPS (European Joint Conferences on Theory and Practice of Software).
- Member of the Steering Committee of QEST (Quantitative Evaluation of Systems).
- Board Member of the Dutch Society on Theoretical Computer Science (NVTI).
- Member of the Program Committee of the following events: FORMATS 08, SSV 08, DSN 08, QEST 07, SAVCBS 07, PERFORMANCE 07, Verify 07, DSN 07.
- Invited speaker at:
  - Summerschool GLOBAN 08, Warsaw, September 2008
  - IEEE/IFIP Symposium TASE 08, Nanjing, June 2008
  - International Conference FORMATS 07, Salzburg, October 2007
  - ARTIST2 Winterschool MOTIVES 07, Trento, March 2007
- Member of the IFIP Working Group 1.8 on Concurrency Theory.
- Member of the EPSRC Review College (Engineering and Physical Sciences Research Council), UK.
- Co-organizer of the Lorentz Workshop on Validation of Stochastic Systems, Leiden, The Netherlands, November 2007.
- Organizer of the Workshop on Automata and Logics (WAL 2007), Aachen, Germany, December 2007.
- Co-organizer of the Dagstuhl Seminar on Quantitative Aspects of Embedded Systems, Schloss Dagstuhl, Wadern, Germany, March 2007
- Member of several external PhD committees.

Th. Noll

- Organizer of the 7th Workshop on Language Descriptions, Tools and Applications (ETAPS/LDTA 2007)
- Program committee member of the Software Engineering Track at the 23rd Annual ACM Symposium on Applied Computing (SAC 2008)
- Program committee member of the 2nd International Workshop on Harnessing Theories for Tool Support in Software (TTSS 2008)
- Program committee member of the 2nd IEEE International Conference on Secure System Integration and Reliability Improvement (SSIRI 2008)

- Member of the examination boards for Computer Science and Computational Material Science
- Student advisor for the following applied subjects within CS: Electrical Engineering, Civil Engineering, and Medicine
- Organization of teaching service of CS Department (http://www-i2.informatik.rwth-aachen.de/Teaching/Service/)

Talks and Publications

Talks

Henrik Bohnenkamp: *Motor: The Modest Tool EnviRonment,* Talk, VOSS 2 Meeting, Leiden, Nov. 2007. Workshop presentation.

Henrik Bohnenkamp: *Quantitative Testing,* Talk, German Chapter Concur, March 2008.

Henrik Bohnenkamp: *Quantitative Testing,* Talk, Quasimodo Workgroup Meeting, Aachen, June 2008. Workshop presentation.

Tingting Han: *Counterexample generation in probabilistic model checking: Theory and practice,* Talk, QUASIMODO Kick-Off Meeting (Aalborg, Denmark), Jan. 2008.

Tingting Han: *Counterexamples in probabilistic model checking,* Talk, TACAS 2007 (Braga, Portugal), March 2007.

Tingting Han: *Counterexamples in probabilistic model checking,* Talk at the University of Birmingham (UK), March 2007.

Tingting Han: *Providing evidence of likely being on time Counterexample generation for CTMC model checking,* Talk, VOSS 2 workshop (Dresden, Germany), Feb. 2007.

Tingting Han: *Providing evidence of likely being on time Counterexample generation for CTMC model checking,* Talk, ATVA'07 (Tokyo, Japan), Oct. 2007.

Tingting Han: *Time-abstracting bisimulation for probabilistic timed automata,* Talk, TASE'08 (Nanjing, China), June 2008.

Jonathan Heinen: *Graph Grammar Abstraction for Complex Dynamic Data Structures,* Talk, KPS 2007, Timmendorfer Strand, Oct. 2007. Workshop Presentation.

Joost-Pieter Katoen: *Abstraction for Probabilistic Systems,* Talk at Int. Conf. on Formal Analysis and Modeling of Timed Systems (FORMATS), Salzburg (Austria), Oct. 2007. Keynote Presentation.

Joost-Pieter Katoen: *Abstraction for Probabilistic Systems,* Colloquium at the University of Oldenburg, Nov. 2007. Invited Presentation.

Joost-Pieter Katoen: *Bisimulation Minimization and Experimental Comparison of Probabilistic Model Checkers,* Talk at the University of Birmingham, March 2007.

Joost-Pieter Katoen: *Good, cheap, fast: Reachability objectives in randomly timed games,* Talk at Technical University of Eindhoven (NL), May 2008.

Joost-Pieter Katoen: *Hoe werkt Google's zoekmachine?,* Talk, RWTH Wissenschaftsnacht, Nov. 2007.

Joost-Pieter Katoen: *Introduction to Probabilistic Model Checking,* Talk at IFIP WG 2.2 Meeting, Nancy (France), Sept. 2007. Invited Presentation.

Joost-Pieter Katoen: *MOTOR: The MoDeST Tool EnviRonment,* Talk at TACAS 2007, Braga (Portugal), March 2007.

Joost-Pieter Katoen: *Reachability objectives in randomly timed games,* Talk, QUASIMODO Kick-Off Meeting, Jan. 2008. Aalborg (Denmark).

Joost-Pieter Katoen: *Perspectives in probabilistic verification,* Keynote talk, IEEE Symposium on Theoretical Aspects of Software Engineering (TASE), June 2008. Keynote talk.

Joost-Pieter Katoen: *Principles of Model Checking,* Lecture Series at Nanjing University, 2008.

Joost-Pieter Katoen: *Principles of Model Checking,* Lecture Series at Tsinghua University, Beijing, June 2008.

Joost-Pieter Katoen: *Perspectives in probabilistic verification,* Invited Colloquium at Institute of Software, Chinese Academy of Sciences, Beijing, July 2008.

Joost-Pieter Katoen: *Reachability objectives in randomly timed games,* Talk at the Workshop on Automata and Logics: History and Perspectives (WAL), Aachen, Dec. 2007. Invited Presentation.

Joost-Pieter Katoen: *Regular Expressions for PCTL Counterexamples,* Talk at QEST 2008, Sept. 2008.

Joost-Pieter Katoen: *Soft Real Time Scheduling and Quality of Service,* Tutorial at the ARTIST2 Winterschool on MOdelling, TestIng, and Verification for Embedded Systems (MOTIVES), Trento (Italy), Feb. 2007. Invited Presentation.

Joost-Pieter Katoen: *The Dutch PhD Programme,* Talk, Fakultätentag 2007, Saarland University, Nov. 2007. Invited Presentation.

Joost-Pieter Katoen: *The Probabilistic Model Checker MRMC,* Talk at Symposium Two Decades of Probabilistic Verification, Leiden (NL), Nov. 2007.

Joost-Pieter Katoen: *True Concurrency in Concurrent Programming,* Guest lecture at the Saarland University, Dec. 2007.

Joost-Pieter Katoen: *Verifying Probabilistic Phenomena: Theory or Practice?,* Colloquium at the University of Konstanz, Jan. 2007. Invited Presentation.

Carsten Kern: *Replaying Play in and Play out: Synthesis of Design Models from Scenarios by Learning,* Talk, TACAS'07 (Braga, Portugal), March 2007.

Carsten Kern: *Smyle: A Tool for Synthesizing Distributed Models from Scenarios by Learning,* Talk, CONCUR'08 (Toronto, Canada), Aug. 2008.

Carsten Kern: *Smyle - Synthesis of Design Models from Scenarios by Learning,* Talk, Kolloquium Programmiersprachen und Grundlagen der Programmierung (Timmendorfer Strand), Oct. 2007.

Carsten Kern: *Synthesizing Design Models from Scenarios by Learning,* Talk, Workshop Programmiersprachen und Rechenkonzepte (Bad Honnef), May 2007. Workshop Presentation.

Daniel Klink: *Abstraction for Stochastic Systems by Erlang's Method of Stages,* Talk at QUASIMODO Workshop, June 2008.

Daniel Klink: *Abstraction for Stochastic Systems by Erlang's Method of Stages,* Talk at CONCUR'08, Aug. 2008.

Alexandru Mereacre: *Model Checking HML On Piecewise-Constant Inhomogeneous Markov Chains,* Talk at FORMATS'08, Sept. 2008.

Martin Neuhäußer: *Bisimulation and Logical Preservation for Continuous-Time Markov Decision Processes,* Talk, Concur 2007, Sept. 2007.

Thomas Noll: *Approaches to Model-Based Analysis and Verification,* Talk, COMPASS Kickoff Meeting, Noordwijkerhout, The Netherlands, Feb. 2008. Workshop Presentation.

Thomas Noll: *Delayed Nondeterminism in Model Checking Embedded Systems Assembly Code,* Talk, Haifa Verification Conference (HVC 2007), Israel, Oct. 2007. Conference Presentation.

Thomas Noll: *Equational Abstractions for Reducing the State Space of Rewrite Theories,* Talk, 7th Int. Workshop on Rewriting Logic and its Applications (WRLA 2008), Budapest, Hungary, March 2008. Workshop Presentation.

Thomas Noll: *SLIM Language: Modifications & Semantics,* Talk, COMPASS Preliminary Design Review Meeting, Trento, Italy, Sept. 2008. Workshop Presentation.

Thomas Noll: *The System-Level Integrated Modeling Language,* Talk, COMPASS System Requirements Review Meeting, Aachen, May 2008. Workshop Presentation.

Stefan Rieger: *Abstracting Complex Data Structures by Hyperedge Replacement,* Talk, ICGT 2008, Leicester (UK), Sept. 2008. Conference Presentation.

Stefan Rieger: *Composing Transformations to Optimize Linear Code,* Talk, ICTAC 2007, Macao (CN), Sept. 2007. Conference Presentation.

Stefan Rieger: *Heap Abstraction by Graph Reduction with Graph Grammars,* Talk, Universität des Saarlandes, Saarbrücken, Aug. 2007. Invited Presentation.

Stefan Rieger: *Unbounded Tread Creation in Dynamic Pointer Programs,* Talk, KPS 2007, Timmendorfer Strand, Oct. 2007. Workshop Presentation.

Stefan Rieger: *Verifying Concurrent List-Manipulating Programs by LTL Model Checking,* Talk, Workshop Programmiersprachen und Rechenkonzepte (Bad Honnef), May 2007. Workshop Presentation.

Stefan Rieger: *Verifying Dynamic Pointer-Manipulating Threads,* Talk, FM 2008, Turku (FI), May 2008. Conference Presentation.

Stefan Rieger: *Verifying Dynamic Pointer-Manipulating Threads,* Talk, German Chapter Concur, March 2008. Workshop Presentation.

Stefan Rieger: *Verifying List-Manipulating Programs with Unbounded Thread Creation,* Talk, Universität des Saarlandes, Saarbrücken, Aug. 2007. Invited Presentation.

Daniel Willems: *Abstraction for Continuous-Time Markov Chains,* Talk, Dagstuhl Seminar: Quantitative Aspects of Embedded Systems, March 2007.

Ivan S. Zapreev: *Bisimulation minimisation mostly speeds up probabilistic model checking,* Talk. TACAS'07, Braga, Portugal, March 2007.

Publications

Christel Baier, Boudewijn R. Haverkort, Holger Hermanns, and Joost-Pieter Katoen: *Reachability in continuous-time Markov reward decision processes,* In Erich Graedel, Joerg Flum, and Thomas Wilke, editors, Logic and Automata: History and Perspectives. pages 53-72. Volume 2 of Texts in Logics and Games. Amsterdam University Press, 2008.

Christel Baier, and Joost-Pieter Katoen: *Principles of Model Checking,* MIT Press, 2008.

Henrik Bohnenkamp, and Axel Belinfante: *Timed Model-Based Testing,* In Jan Tretmans, editor, Tangram: Model-based integration and testing of complex high-tech systems. pages 115-128. Embedded Systems Institute, The Netherlands, 2007.

Henrik Bohnenkamp, Holger Hermanns, and Joost-Pieter Katoen: *Motor: The MoDeST Tool Environment,* In Proceedings of the 13th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'07). pages 500-504. Volume 4424 of Lecture Notes in Computer Science. Springer-Verlag, 2007.

Henrik Bohnenkamp, and Marielle Stoelinga: *Quantitative Testing,* In Proc. EMSOFT 2008. ACM, 2008.

Benedikt Bollig, Joost-Pieter Katoen, Carsten Kern, and Martin Leucker: *Replaying Play in and Play out: Synthesis of Design Models from Scenarios by Learning,* In Proceedings of the 13th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'07). pages 435-450. Volume 4424 of Lecture Notes in Computer Science. Springer Verlag, 2007.

Benedikt Bollig, Joost-Pieter Katoen, Carsten Kern, and Martin Leucker: *SMATheSmyle Modeling Approach,* In 3rd IFIP TC2 Central and East European Conference

on Software Engineering Techniques (CEE-SET). Lecture Notes in Computer Science. 2008.

Benedikt Bollig, Carsten Kern, Joost-Pieter Katoen, and Martin Leucker: *Smyle: a Tool for Synthesizing Distributed Models from Scenarios by Learning,* In 19th International Conference on Concurrency Theory (CONCUR'08). pages 162-166. Volume 5201 of Lecture Notes in Computer Science. Springer, 2008.

Taolue Chen, Tingting Han, and Joost-Pieter Katoen: *Time-Abstracting Bisimulation for Probabilistic Timed Automata,* In 2nd IEEE International Symposium on Theoretical Aspects of Software Engineering (TASE). pages 177-184. IEEE CS Press, 2008.

Berteun Damman, Tingting Han, and Joost-Pieter Katoen: *Regular Expressions for PCTL Counterexamples,* In Quantitative Evaluation of Systems (QEST). IEEE CS Press, 2008.

Tingting Han, and Joost-Pieter Katoen: *Counterexamples in probabilistic model checking,* In Proceedings of the 13th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'07). pages 60-75. Volume 4424 of Lecture Notes in Computer Science. Springer Verlag, 2007.

Tingting Han, and Joost-Pieter Katoen: *Providing evidence of likely being on time - Counterexample generation for CTMC model checking,* In Proceedings of the 5th International Symposium on Automated Technology for Verification and Analysis (ATVA'07). Lecture Notes in Computer Science. Springer Verlag, 2007.

Tingting Han, Joost-Pieter Katoen, and Alexandru Mereacre: *Compositional Modeling and Minimization of Time-inhomogeneous Markov Chains,* In Hybrid Systems: Computation and Control (HSCC). pages 244-258. Volume 4981 of Lecture Notes in Computer Science. Springer Verlag, 2008.

Lars Helge Ha, and Thomas Noll: *Equational Abstractions for Reducing the State Space of Rewrite Theories,* In Proc. of 7th Int. Workshop on Rewriting Logic and its Applications (WRLA 2008). ENTCS. Elsevier, 2008.

Gerlind Herberich, Thomas Noll, Bastian Schlich, and Carsten Weise: *Proving Correctness of an Efficient Abstraction for Interrupt Handling,* In Proc. 3rd Int. Workshop on Systems Software Verification (SSV '08). pages 133-150. Volume 217 of ENTCS. Elsevier, 2008.

David N. Jansen, Joost-Pieter Katoen, Marcel Oldenkamp, Marielle Stoelinga, and Ivan S. Zapreev: *How fast and fat is your probabilistic model checker? An experimental comparison,* In Proceedings of the 3rd Haifa Verification Conference (HVC 2007). pages 69-85. Volume 4899 of Lecture Notes in Computer Science. Springer, 2008.

Joost-Pieter Katoen: *Abstraction of Probabilistic Systems,* In Formal Methods for Timed Systems (FORMATS'07). pages 1-3. Volume 4763 of LNCS. Springer-Verlag, 2007.

Joost-Pieter Katoen: *Perspectives in Probabilistic Verification,* In 2nd IEEE International Symposium on Theoretical Aspects of Software Engineering (TASE). pages 3-10. IEEE CS Press, 2008.

Joost-Pieter Katoen, Tim Kemna, Ivan S. Zapreev, and David N. Jansen: *Bisimulation minimisation mostly speeds up probabilistic model checking,* In Proceedings of the 13th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'07). pages 76-92. Volume 4424 of Lecture Notes in Computer Science. Springer Verlag, 2007.

Joost-Pieter Katoen, Daniel Klink, Martin Leucker, and Verena Wolf: *Abstraction for Stochastic Systems by Erlang's Method of Stages,* In 19th International Conference on Concurrency Theory (CONCUR'08). pages 279-294. Volume 5201 of Lecture Notes in Computer Science. Springer, 2008.

Joost-Pieter Katoen, Daniel Klink, Martin Leucker, and Verena Wolf: *Three-valued abstraction for continuous-time Markov chains,* In Proceedings of the 19th International Conference on Computer Aided Verification (CAV). pages 311-324. Volume 4590 of Lecture Notes in Computer Science. Springer Verlag, 2007.

Joost-Pieter Katoen, Daniel Klink, Martin Leucker, and Verena Wolf: *Three-Valued Abstraction for Probabilistic Systems,* Technical Report , RWTH Aachen University, 2007.

Joost-Pieter Katoen, and Alexandru Mereacre: *Model Checking HML On Piecewise-Constant Inhomogeneous Markov Chains,* In FORMATS'08. Volume 5215 of Lecture Notes in Computer Science. Springer-Verlag, 2008.

Joost-Pieter Katoen, Thomas Noll, and Stefan Rieger: *Verifying Concurrent List-Manipulating Programs by LTL Model Checking,* In Workshop on Heap Analysis and Verification (HAV 2007)pages 94-113. 2007.

Martin R. Neuhäußer, and Joost-Pieter Katoen: *Bisimulation and Logical Preservation for Continuous-Time Markov Decision Processes,* In 18th International Conference on Concurrency Theory (CONCUR'07). pages 412-427. Volume 4703 of LNCS. Springer-Verlag, 2007.

Martin R. Neuhäußer, and Thomas Noll: *Abstraction and Model Checking of Core Erlang Programs in Maude,* In Proceedings of the 6th International Workshop on Rewriting Logic and its Applications. pages 147-163. Volume 176 of ENTCS. Elsevier, 2007.

Viet Yen Nguyen: *Optimising Techniques for Model Checkers,* Masters Thesis, University of Twente, 2007.

Viet Yen Nguyen, and Theo C. Ruys: *Incremental Hashing for SPIN,* In Proceedings 15th International SPIN Workshop on Model Checking of Software. Lecture Notes in Computer Science. 2008.

Rocco De Nicola, Joost-Pieter Katoen, Diego Latella, Michele Loreti, and Mieke Massink: *Model checking mobile stochastic logic,* Theoretical Computer Science, 382:42-70, 2007.

Thomas Noll, and Bastian Schlich: *Delayed Nondeterminism in Model Checking Embedded Systems Assembly Code,* In Hardware and Software: Verification and Testing (Proc. of 3rd Int. Haifa Verification Conf., HVC 2007). pages 185-201. Volume 4899 of Lecture Notes in Computer Science. Springer, 2008.

Thomas Noll, and Stefan Rieger: *Composing Transformations to Optimize Linear Code,* In Proc. 4th Int. Colloquium on Theoretical Aspects of Computing (ICTAC '07). pages 425-439. Volume 4711 of LNCS. 2007.

Thomas Noll, and Stefan Rieger: *Verifying Dynamic Pointer Programs,* In 15th Int. Symp. on Formal Methods (FM '08). pages 84-99. Volume 5014 of LNCS. Springer, 2008.

Stefan Rieger, and Thomas Noll: *Abstracting Complex Data Structures by Hyperedge Replacement,* In 4th Int. Conference on Graph Transformations (ICGT 2008). pages 69-83. Volume 5214 of LNCS. Springer, 2008.

Volker Stolz: *Temporal assertions for sequential and concurrent programs,* PhD Dissertation, AiB 2007-15, RWTH Aachen University, Dept. of Computer Science, 2007.

Mani Swaminathan, Martin Fraenzle, and Joost-Pieter Katoen: *The Surprising Robustness of (Closed) Timed Automata against Clock-Drift,* In 5th IFIP International Conference on Theoretical Computer Science (IFIP TCS). 2008.

Ivan S. Zapreev: *Model Checking Markov Chains: Techniques and Tools,* PhD Thesis, University of Twente, 2008.

Lijun Zhang, Holger Hermanns, Friedrich Eisenbrand, and David N. Jansen: *Flow faster: Efficient Decision Algorithms for Probabilistic Simulations,* In Orna Grumberg, and Michael Huth, Editors, TACAS 2007. pages 155-169. Volume 4424 of LNCS. Springer, 2007.