

Alexander Malkis



CONTACT INFORMATION

Chair of Software Engineering
University of Freiburg
Georges-Köhler-Allee 052
79110 Freiburg im Breisgau
Germany
Phone: (+49) 761 203 6953
Email: alexmalk@informatik.uni-freiburg.de
Web: <http://swt.informatik.uni-freiburg.de/~alexmalk>

OBJECTIVE

Full-time research or teaching position.

EDUCATION

Autumn 2000 —

University of Saarland, Germany:

- Study of mathematics and computer science at the university of Saarland, Germany. Supported by the renowned “Studienstiftung des Deutschen Volkes”. Among the best in mathematics and computer science.
- Diploma thesis: “Polyominoes, polyedges and the Digit game”. Advisor: Raimund Seidel.

— January 2005
2004 — 2006

Grade: very good.

- PhD student at the Max-Planck-Institut für Informatik. Advisor: Andreas Podelski.

Albert-Ludwigs University of Freiburg, Germany:

2006 — 2009
July 2009

- Continuing PhD studies. Supported by a DFG-fellowship for graduates.
- PhD thesis submitted.

May — July 2008

Microsoft Research, Redmond, USA:

- Internship. Proving safety of multithreaded drivers via history specifications. Mentors: Shaz Qadeer, Shuvendu Lahiri.

Microsoft Research, Cambridge, UK:

April — June 2009
September 2009 –

- Internship. Security language SecPAL for privacy. Mentor: Moritz Y. Becker.
- Internship. Concurrency and Shape Analysis. Mentor: Byron Cook.

RESEARCH STATEMENT

Verification of multithreaded programs.

I've proven that three different verification methods have the same precision: the Owicki-Gries proof rule, thread-modular reasoning and Cartesian abstract interpretation.

I've shown how to remove their common precision limit without enlarging the shared state space. For that, I discovered the technique of so-called *exception sets*, which get subtracted before and get added back after an approximation.

I've shown how to automatically find a suitable parameter (= exception set) by a counterexample-guided abstraction scheme. Implementation AMTV fights the state-explosion problem in the number of threads for finite-state programs.

I've discovered a complete algorithm (CEGAR with exceptions) which is polynomial on all programs with thread-modular proofs and on a practically interesting class of programs without thread-modular proofs. The implementation shows fast behavior also in practice.

For the underlying decision queries, I've discovered a resolution technique for Cartesian products to solve inclusion queries between unions of Cartesian products.

The described work was done in the course of my PhD studies with Podelski and Rybalchenko.

For dealing with large multithreaded programs by reasoning about their small models I've discovered how to verify soundness of thread abstractions (with Qadeer and Lahiri).

Privacy and security.

I've shown how to use an authorization policy language (SecPAL) for specifying and enforcing privacy policies and preferences regarding users' personal information in networks. The work is the first one that creates a thoroughly formal model for a privacy policy language (with Becker).

- RESEARCH INTERESTS
- **Software verification and program analysis:**
 - Concurrency: modularity, compositionality, model-checking of multithreaded programs
 - Specification: thread abstractions
 - Abstract interpretation: smooth refinement and completeness.
 - **Privacy and security:**
 - Specification languages, formal models, logic, policies and preferences, privacy-enabled dynamic networks
 - Access control, delegation, authorization.
 - **Combinatorics:** enumeration.
 - Future research:
 - Polynomial verification of large program classes
 - Emptiness of language intersection: complexity and algorithms
 - Thread simulations, liveness, procedure abstractions under concurrency
 - A working verifier for multithreaded C
 - Verifying multithreaded programs with richer structure: with probabilism, recursion, for multicore systems.
- PROFESSIONAL AND TEACHING ACTIVITIES
- Lecture (led by Prof. Dr. Podelski): Verification, Uni Freiburg, Winter 2007-08.
 - Seminar: Abstract Interpretation, Uni Freiburg, Winter 2007-08.
 - Seminar: Advanced topics in Model Checking, Uni Freiburg, Summer 2007.
 - Seminar: Software Model Checking, Uni Freiburg, Winter 2006-07.
 - Seminar: Software Model Checking, MPII, Summer 2005.
 - Reading Group at MPII, Winter 2005 - Summer 2006.
 - Teaching assistant in lectures in mathematics and computer science during diploma studies.
 - Programmer at TelComNet: automatic database-driven CD production web service (1999).
- COMPUTER SKILLS
- OS: Unix/Linux, Windows, MS DOS.
 - Languages: C/C++, Java, ML (Ocaml, MosML, SML), Fortran 77, HTML, x86 assembler, Pascal (Turbo, Delphi), Basic (GW, Turbo, Quick, Visual), Lisp, C#/Spec#, BoogiePL.
 - Development: 3D-multiplayer network game, verifier AMTV.
 - Website design.
- PUBLICATIONS
- Submitted: PhD thesis.
 - Submitted: Counterexample-Guided Refinement for Thread-Modular Verification.
 - *Abstract Threads*, VMCAI 2010, Madrid, Springer. With Lahiri, Qadeer.
 - *Precise Thread-Modular Verification*, SAS 2007, Kongens Lyngby, Springer. With Podelski, Rybalchenko.
 - *Thread-Modular Verification is Cartesian Abstract Interpretation*, ICTAC 2006, Tunis, Springer. Longer version *Thread-Modular Verification and Cartesian Abstraction* presented at the TV 2006 workshop in Seattle. With Podelski, Rybalchenko.
 - *Polyominoes, polyedges and the digit game*, Diploma thesis, Saarbrücken, 2004.
- LANGUAGES
- English, German, Russian.
- REFERENCES
- Available upon request.
- PERSONAL INTERESTS
- Karate