Dominic Steinhöfel | Curriculum Vitae

Kahlertstr. 30, 64293 Darmstadt, Germany
steinhoefel.dominic@gmail.com
https://www.dominic-steinhoefel.de
https://scholar.google.com/citations?user=s3VQvR8AAAAJ
https://dblp.uni-trier.de/pers/hd/s/Steinh=ouml=fel:Dominic
https://orcid.org/0000-0003-4439-7129

Research Interests

My research focuses on languages, techniques, and tools for increasing confidence in software correctness. To this end, I work on testing and program proving approaches and design specification languages for the unit and system level. As part of my research, I contributed Abstract Execution [6], an automatic verification framework mainly for unit-level program transformations, and ISLa [1], a language and solver for complex string constraints for the precise testing and analysis of software systems.

Positions and Experience

Researcher (PostDoc), CISPA Helmholtz Center for Information Security — Jan 1st, 2021–?

Full-time researcher in the research group of Prof. Dr. Andreas Zeller.

Research Assistant, Technische Universität Darmstadt — May 1st, 2015–Dec. 31st, 2020

"Software Engineering" group of Prof. Dr. Reiner Hähnle. Responsibilities: Research, Teaching, Administration (State Position).

Student Assistant, Technische Universität Darmstadt — 2014–2015

Enforcement of security properties for distributed systems, modeling service-oriented architectures. Supervisor: Dr. Jinwei Hu.

Tutor, Technische Universität Darmstadt — 2014

Tutor for the lecture "Formale Grundlagen der Informatik I/II" (automata, formal languages, logic for computer science). Supervisor: Prof. Dr. Ulrich Kohlenbach.

Student Assistant, Technische Universität Darmstadt — 2013

Enforcement of security policies in distributed systems. Supervisor: Dr. Richard Gay.

Tutor, Technische Universität Darmstadt — 2012–2014

Tutor for the lecture "Formale Grundlagen der Informatik III": Model checking of parallel systems, automated verification of Java programs.

Freelance Programmer, heimkinomarkt.de GmbH — 2006–2014

Website development (ASP.NET, HTML, CSS, JavaScript, Adobe Flash) and database administration.

Education

Technische Universität Darmstadt, Dr. rer. nat. (Computer Science) 2020

Thesis Abstract Execution: Automatically Proving Infinitely Many Programs [14]

Supervisors Prof. Dr. Reiner Hähnle, Prof. Gilles Barthe, PhD

Grade *Summa cum laude* (with distinction)

The dissertation presents a technique named *Abstract Execution* to prove functional and relational properties of *abstract* programs, especially of program transformation rules. Abstract Execution trades off expressiveness and automation, yielding a general framework allowing for fully automatic proofs in many cases. An application of this technique to well-known equivalence preserving program transformation rules, so-called *refactorings*, provides new preconditions for a safe application of these rules, exceeding the state documented in the literature. Conditional correctness of the rules is proven fully automatically (even for loop transformations). Furthermore, the dissertation contains fundamental contributions to symbolic execution, and proposes a trace-based framework unifying different problems in the area of program verification.

Several follow-up works are based on my Ph.D. thesis [2–5] [11, 12].

Technische Universität Darmstadt, M. Sc. (Computer Science) 2015

Thesis From Trees to Directed Acyclic Graphs: A General Lattice Model for Symbolic Execution [15]

Supervisors Prof. Dr. Reiner Hähnle, Priv.-Doz. Dr. Richard Bubel, Dr. Nathan Wasser

Grades Thesis: 1.0 (very good), Average: 1.35 (very good)

I published the contents of my Master's thesis at an international conference [10].

Technische Universität Darmstadt, B. Sc. (Computer Science) 2013

Thesis Enforcing Datalog Policies with Service Automata on Distributed Version Control Systems [16]

Supervisors Prof. Dr. Heiko Mantel, Dr. Richard Grewe (né Gay)

Grades Thesis: 1.7 (good), Average: 1.4 (very good)

Publications

Google Scholar `https://scholar.google.com/citations?user=s3VQvR8AAAAJ`
DBLP `https://dblp.uni-trier.de/pers/hd/s/Steinh=ouml=fel:Dominic`

Citations: 102, h-index: 6, i10-index: 3

Below are lists of my theses and peer-reviewed publications.

Please note that I authored my Bachelor's and Master's theses [15, 16] and one conference presentation [10] under my former family name Scheurer.

Peer-Reviewed Conference Publications

1. **Dominic Steinhöfel** and Andreas Zeller. Input Invariants. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE '22), November 14–18, 2022, Singapore*. ACM, 2022. To appear.
2. Elvira Albert, Reiner Hähnle, Alicia Merayo, and **Dominic Steinhöfel**. Certified Abstract Cost Analysis. In Esther Guerra and Mariëlle Stoelinga, editors, *24th International Conference on Fundamental Approaches to Software Engineering (FASE), Held as Part of the European Joint Conferences on Theory and Practice of Software (ETAPS) 2021*, volume 12649 of *LNCS*, pages 24–45. Springer, 2021. doi: 10.1007/978-3-030-71500-7_2.

3. Marco Scaletta, Reiner Hähnle, **Dominic Steinhöfel**, and Richard Bubel. Delta-Based Verification of Software Product Families. In Eli Tilevich and Coen De Roover, editors, *GPCE '21: Concepts and Experiences, Chicago, IL, USA, October 17 - 18, 2021*, pages 69–82. ACM, 2021. doi: 10.1145/3486609.3487200.

4. Reiner Hähnle, Asmae Heydari Tabar, Arya Mazaheri, Mohammad Norouzi, **Dominic Steinhöfel**, and Felix Wolf. Safer Parallelization. In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation: Engineering Principles - 9th International Symposium on Leveraging Applications of Formal Methods, ISoLA 2020, Rhodes, Greece, October 20-30, 2020, Proceedings, Part II*, volume 12477 of *Lecture Notes in Computer Science*, pages 117–137. Springer, 2020. doi: 10.1007/978-3-030-61470-6_8.

5. **Dominic Steinhöfel**. REFINITY to Model and Prove Program Transformation Rules. In Bruno C. d. S. Oliveira, editor, *Programming Languages and Systems - 18th Asian Symposium, APLAS 2020, Fukuoka, Japan, November 30 - December 2, 2020, Proceedings*, volume 12470 of *LNCS*, pages 311–319. Springer, 2020. doi: 10.1007/978-3-030-64437-6_16.

6. **Dominic Steinhöfel** and Reiner Hähnle. Abstract Execution. In Maurice H. ter Beek, Annabelle McIver, and José N. Oliveira, editors, *Formal Methods - The Next 30 Years - Third World Congress, FM 2019, Porto, Portugal, October 7-11, 2019, Proceedings*, volume 11800 of *Lecture Notes in Computer Science*, pages 319–336. Springer, 2019. doi: 10.1007/978-3-030-30942-8_20.

7. **Dominic Steinhöfel** and Reiner Hähnle. The Trace Modality. In Luís Soares Barbosa and Alexandru Baltag, editors, *Dynamic Logic. New Trends and Applications - Second International Workshop, DaLí 2019, Porto, Portugal, October 7-11, 2019, Proceedings*, volume 12005 of *LNCS*, pages 124–140. Springer, 2019. doi: 10.1007/978-3-030-38808-9_8.

8. **Dominic Steinhöfel** and Reiner Hähnle. Modular, Correct Compilation with Automatic Soundness Proofs. In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation. Modeling - 8th International Symposium, ISoLA 2018, Limassol, Cyprus, November 5-9, 2018, Proceedings, Part I*, volume 11244 of *Lecture Notes in Computer Science*, pages 424–447. Springer, 2018. doi: 10.1007/978-3-030-03418-4_25.

9. **Dominic Steinhöfel** and Nathan Wasser. A New Invariant Rule for the Analysis of Loops with Non-standard Control Flows. In Nadia Polikarpova and Steve A. Schneider, editors, *Integrated Formal Methods - 13th International Conference, IFM 2017, Turin, Italy, September 20-22, 2017, Proceedings*, volume 10510 of *Lecture Notes in Computer Science*, pages 279–294. Springer, 2017. doi: 10.1007/978-3-319-66845-1_18.

10. **Dominic Scheurer**, Reiner Hähnle, and Richard Bubel. A General Lattice Model for Merging Symbolic Execution Branches. In Kazuhiro Ogata, Mark Lawford, and Shaoying Liu, editors, *Formal Methods and Software Engineering - 18th International Conference on Formal Engineering Methods, ICFEM 2016, Tokyo, Japan, November 14-18, 2016, Proceedings*, volume 10009 of *Lecture Notes in Computer Science*, pages 57–73, 2016. doi: 10.1007/978-3-319-47846-3_5.

Peer-Reviewed Journal Publications and Book Chapters

11. **Dominic Steinhöfel**. Symbolic Execution: Foundations, Techniques, Applications, and Future Perspectives. In Wolfgang Ahrendt, Bernhard Beckert, Richard Bubel, and Einar Broch Johnsen, editors, *The Logic of Software. A Tasting Menu of Formal Methods: Essays Dedicated to Reiner Hähnle on the Occasion of His 60th Birthday*, volume 10009 of *LNCS*, pages 446–480. Springer, 2021. doi: 10.1007/978-3-031-08166-8_22.

12. **Dominic Steinhöfel**. *Ever Change a Running System:* Structured Software Reengineering Using Automatically Proven-Correct Transformation Rules. In Michael Felderer, Wilhelm Hasselbring, Heiko Koziolek, Florian Matthes, Lutz Prechelt, Ralf H. Reussner, Bernhard Rumpe, and Ina Schaefer, editors, *Ernst Denert Award for Software Engineering 2020: Practice Meets Foundations*, pages 197–226. Springer, 2020. doi: 10.1007/978-3-030-83128-8_10.

13. Stijn de Gouw, Frank S. de Boer, Richard Bubel, Reiner Hähnle, Jurriaan Rot, and **Dominic Steinhöfel**. Verifying OpenJDK's Sort Method for Generic Collections. *J. Autom. Reasoning*, 62(1):93–126, 2019. doi: 10.1007/s10817-017-9426-4.

Theses

14. **Dominic Steinhöfel**. *Abstract Execution: Automatically Proving Infinitely Many Programs*. PhD thesis, Technical University of Darmstadt, Department of Computer Science, Darmstadt, Germany, 2020. doi: 10.25534/tuprints-00008540. URL `http://tuprints.ulb.tu-darmstadt.de/8540/`.

15. **Dominic Scheurer**. From Trees to DAGs: A General Lattice Model for Symbolic Execution. Master's thesis, Technical University of Darmstadt, Department of Computer Science, Darmstadt, Germany, 2015. URL `https://download.hrz.tu-darmstadt.de/media/FB20/Dekanat/Publikationen/SE/Scheurer_-_2015_-_From_Trees_to_DAGs.pdf`.

16. **Dominic Scheurer**. Enforcing Datalog Policies with Service Automata on Distributed Version Control Systems. Bachelor's thesis, Technical University of Darmstadt, Department of Computer Science, Darmstadt, Germany, 2013.

External Funding

I was associated to the LOEWE project "Software Factory 4.0"[1] funded by the German state of Hesse and supported the application process as non-PI. Our work applying Abstract Execution to program parallelization [4] resulted from this project. Currently, I am preparing a DFG proposal about using ISLa constraints to connect program verification at the system and unit levels.

Presentations at International Meetings

Invited Talks

Tutorial Session @ 15th Intern. Conf. on integrated Formal Methods, Bergen, Norway 2019

Title: *"How to Prove the Correctness of Refactoring Rules"*
Link: `https://ifm2019.hvl.no/refa/#pcrr`

In this tutorial session, I gave a 30-minutes talk presenting Abstract Execution and its application to proving program transformation rules. The participants used REFINITY in the second 30-minutes slot to prove the conditional correctness of two refactoring rules on their own.

International Conferences with Publication

18th Asian Symposium on Programming Languages and Systems, Japan/Held Online 2020

Title: *"REFINITY to Model and Prove Program Transformation Rules"*
Link: `https://conf.researchr.org/track/aplas-2020/aplas-2020-papers#event-overview`

Third World Congress on Formal Methods, Porto, Portugal 2019

Title: *"Abstract Execution"*
Link: `https://easychair.org/smart-program/FM2019/`

Second Intern. Workshop on Dynamic Logic, Porto, Portugal 2019

Title: *"The Trace Modality"*
Link: `https://workshop.dali.di.uminho.pt/`

[1] `https://www.software-factory-4-0.de/en/`

8th ISoLA, Limassol, Cyprus 2018

Title: *"Modular, Correct Compilation with Automatic Soundness Proofs"*
Link: `http://www.isola-conference.org/isola2018/programme.html`

The website lists Reiner Hähnle (coauthor) as speaker; the talk was prepared and given by me, however.

13th Intern. Conf. on integrated Formal Methods, Torino, Italy 2017

Title: *"A New Invariant Rule for the Analysis of Loops with Non-standard Control Flows"*
Link: `http://ifm2017.di.unito.it/program.php`

18th Intern. Conf. on Formal Engineering Methods, Tokyo, Japan 2016

Title: *"A General Lattice Model for Merging Symbolic Execution Branches"*

The website of this event is no longer accessible.

Workshops without Publication

18th Intern. KeY Symposium, Manigod, France 2019

Title: *"Abstract Execution"*
Link: `https://www.key-project.org/key-symposium-2019/`

17th Intern. KeY Symposium, Gothenburg, Sweden 2018

Title: *"Correct Compilation with Automatic Soundness Proofs"*
Link: `https://www.key-project.org/key-symposium-2018/`

PhD Symposium @ 13th Intern. Conf. on integrated Formal Methods, Torino, Italy 2017

Title: *"Assessing the Coverage of Formal Specifications"*
Link: `http://ifm2017.di.unito.it/callForPhDSymposium.php`

16th Intern. KeY Symposium, Rastatt, Germany 2017

First Talk: *"A New Invariant Rule for the Analysis of Loops with Non-standard Control Flows"*
Second Talk: *"Assessing the Coverage of Formal Specifications"*
Link: `https://www.key-project.org/key-symposium-2017/`

30th "Deduktionstreffen" of the German Informatics Society, Klagenfurt, Austria 2016

Title: *"A General Lattics Model for Merging Symbolic Execution Branches"*

The website of this event is no longer accessible.

14th Intern. KeY Symposium, Gothenburg, Sweden 2015

Title: *"A General Lattics Model for Merging Symbolic Execution Branches"*
Link: `https://www.key-project.org/keysymposium15/`

Other Events

Symposium on the Occasion of Reiner Hähnles 60th Birthday, Darmstadt, Germany 2022

Title: *"All you need to know about Symbolic Execution"*

Finals of the Ernst Denert Award for Software Engineering, Held Online 2021

Title: *"Ever Touch a Running System"*
Link: `https://se-2021.tu-bs.de/programm/`
Link: `https://www.youtube.com/watch?v=JMOVhAt0aqI`

Teaching Experience

Ongoing Thesis Supervisions

I am currently (co-)supervising four final theses.

Supervised Theses

So far, I (co-)supervised six final theses.

Courses

Multimodal Seminar on Symbolic Execution 2022

This seminar, designed and organized by myself in 2022 at CISPA/UdS, addresses automatic software testing techniques based on symbolic execution. In contrast to standard seminars, it consists of lecture, presentation, and lab sessions. The students implement fundamental aspects from the most recently presented paper in the lab sessions. To facilitate this, I created a symbolic execution framework in the form of a Jupyter book such that the students can focus on the most critical aspects without much boilerplate work. I plan to publicly release the Jupyter book as an interactive "Symbolic Execution Book" and to expand this seminar to a lecture on "Rigorous Software Engineering."

Software Engineering Exercises 2016–2017

The lecture "Software Engineering" is a mandatory course for Bachelor students of computer science at Technische Universität Darmstadt. It consists of a weekly, 90-minutes lecture and a 45-minutes exercise session held directly after the lecture in the same room. In the winter terms 2016 and 2017, I was responsible for the exercises, which included preparing the exercise sheets, chairing the weekly exercise sessions, and administrating and coaching student assistants who offered weekly consultation hours and corrected exercise sheets with bonus assignments. In 2017, 500 students registered for the exam of the course.

Bachelorpraktikum 2015–2017

The Bachelorpraktikum (english "bachelor lab") is a mandatory course for 5-th semester Bachelor students of computer science at Technische Universität Darmstadt. Teams of four to five students realize, during a whole 6-month term, software projects offered by members of the university (mostly, but not exclusively, of the computer science department). I assisted in the organization of the Bachelorpraktikum in the summer and winter term 2015/2016 and in the winter term 2017, with responsibilities such as the assignments of students to teams or the communication with student assistants ("team leaders") in case of problems with the teamwork. I additionally contributed to the course by developing a grading rubric for quality assurance documents produced by the students, and by creating a complex online administration software "BP Admin" especially for the course. As of 2020, BP Admin is still being used by other groups organizing this course.

Seminars 2018–2022

I supervised several participants of the seminars *"Software Project Failures"* (2017), *"Symbolic Execution"* (2018/19) and *"Actor-Based Languages"* (2019/20), all at TU Darmstadt.

I was the main organizer for the seminars *"Selected Topics in Specification and Testing"* in 2021 and *"Automated Testing and Debugging"* in 2021/22 (at CISPA).

Reviewing and Service

Organizer

First Intern. HacKeYthon, Karlsruhe, Germany 2018

The "HacKeYthon" is an event with the goal to bring forward the development of the KeY program verification framework, and, at the same time, unite experienced developers, new project members and associates as well as interested students. Participants work in small groups, which are mixed in terms of experience to enable knowledge transfer, on chosen programming projects related to KeY. The idea of the HacKeYthon was originally conceived by me in 2017, and I was the main organizer of the first such event in 2018.

Link: `https://www.key-project.org/1st-hackeython-2018/`

15th Intern. KeY Symposium, Manigod, France 2016

I organized the 15th KeY Symposium in Manigod, France. This included communication with participants (emails, website), assembling the program, and organizational issues regarding the venue (rooms, catering).

Link: `http://i12www.ira.uka.de/key/keysymposium16/`

Reviewer

I have been reviewing articles for the journals FAOC (2017), JLAMP (2020), SCICO (2021), TOSEM (2021), and STTT (2022) and the conferences TAP (2016, 2018, and 2022), SAS (2017), CPP (2017 and 2018), ISoLA (2018), FASE (2018 and 2020), DaLí and TABLEAUX (2019), and CONCUR and SEFM (2020).

PhD Thesis Committee

I served in two Ph.D. examination committees as "Akademischer Beisitzer:" Jesko Hecking-Harbusch (2021, supervisor: Bernd Finkbeiner) and Nikolas Havrikov (2022, supervisor: Andreas Zeller).

Languages

German (native), English (fully proficient), Spanish (advanced), French (intermediate), Italian (beginner), Latin (*Latinum*).