Progress Report of the Max Planck Institute for Software Systems (MPI-SWS)

August 2015 – January 2018

MAX PLANCK INSTITUTE
FOR SOFTWARE SYSTEMS

Contents

1 State of the Institute

This progress report of the Max Planck Institute for Software Systems (MPI-SWS) covers the period August 2015 – January 2018. We begin with an overview of:

- the mission, goals, and general structure of the institute (Section 1.1), and
- the current state of the institute and our recent accomplishments (Section 1.2).

Currently, the institute's faculty consists of 4 scientific directors, 5 tenured, and 4 tenure-track faculty members, each with an independent research group. During the reporting period, Björn Brandenburg, Deepak Garg, and Viktor Vafeiadis received tenure. Joel Ouaknine joined as a Scientific Director, and Eva Darulova, Maria Christakis, and Adish Singla joined as tenure-track faculty members. In addition, there are 2 research group leaders on non-tenure track positions, Daniel Neider and Damien Zufferey, both of whom joined in the reporting period. With these hires, we have consolidated our strong presence in verification, programming languages, and software engineering, and in social information systems.

The subsequent sections of the document provide individual progress reports by each of the institute's 13 independent research groups that were active during this review period. Finally, Section 16 provides summary information and details about the institute and its activities.

1.1 General overview of the institute

This section presents a general overview of the goals, structure, and organization of the institute, not specific to the present review period.

1.1.1 Mission and strategic goals

Computer systems permeate our daily life. They support the operation of our financial, medical, educational, and administrative institutions; they facilitate science, manufacturing, transportation, and trade; and they enable new forms of entertainment and social exchange. The Max Planck Institute for Software Systems, located in Kaiserslautern and Saarbrücken, studies the principles of efficient, dependable, secure, and usable computing systems, as well as their interaction with the physical and social context in which they

operate. Areas of particular interest include dependable software, cyber-physical systems, social computing, and privacy. The Institute conducts foundational research in relevant areas of computer science and beyond, covering theory, empirical analysis, and data-driven investigation.

As an academic institution dedicated to high-risk, long-term research, the primary goal is to have impact primarily through publications, artifacts, and people. We seek to attract outstanding talent from all over the world, thus broadening the pool of talent in Germany and Europe. At the same time, we expect our graduates to be competitive for academic and research positions at top universities and laboratories worldwide. In the process, we aim to contribute to a stronger and broader base of software systems research in Germany and Europe.

1.1.2 Situation

MPI-SWS, one of 84 institutes comprising the Max Planck Society (MPS), was founded in November 2004 and opened its doors in August 2005. The institute has two sites, one located on the campus of Saarland University (UdS), the other on the campus of the Technical University (TU) Kaiserslautern. The sites are 45 minutes apart by car, door-to-door.

Kaiserslautern and Saarbrücken are cities with about 100,000 and 180,000 inhabitants, respectively. The cities offer attractive surroundings and a low cost of living. Access to major metropolitan areas is easy via high-speed rail (two hours to Paris) and low-cost flights from the local airports (Saarbrücken and Luxembourg). Frankfurt airport, the closest international hub, is a 60 minute drive from Kaiserslautern and a 90 minute drive from Saarbrücken.

Several research organizations in related areas are located at the two sites. The computer science department at Saarland University ranks among the top five in Germany. The Max Planck Institute for Informatics (MPI-INF) in Saarbrücken focuses on algorithms, vision and graphics, bioinformatics, databases and information systems, and networking. The German Research Center for Artificial Intelligence (DFKI), an applied research lab on artificial intelligence, has locations in both Saarbrücken and Kaiserslautern. Recently, the Helmholtz Foundation established the Helmholtz Center for IT-Security as a continuation of the Center for IT Security, Privacy, and Accountability (CISPA) at Saarland University. MPI-SWS is part of the Cluster of Excellence on "Multimodal Computing and Interaction," and the Center for IT Security, Privacy and Accountability (CISPA) at Saarland University.

The computer science department at the TU Kaiserslautern ranks in the

top quartile of departments in Germany. Kaiserslautern hosts two applied research institutes, the Fraunhofer Institute for Experimental Software Engineering and the Fraunhofer Institute for Industrial Mathematics, in addition to the German Research Center for Artificial Intelligence (DFKI). There are also a number of information technology startups and a few mid-sized companies at both sites.

MPI-SWS faculty participate in the Cluster of Excellence for Multimedia Computing and Communication (MMCI) at Saarland University, the Saarbrücken Graduate School for Computer Science, the Center for IT Security, Privacy and Accountability (CISPA) at Saarland University, and the Kaiserslautern Science Alliance.

The MPI-SWS has a total budget of about EUR 10M per year and 18 faculty positions (13 of which are currently filled). The institute buildings at the two sites jointly offer space for over 200 researchers and staff. Additional growth is expected through external funding. In this reporting period, the institute received over €2.15M in external funding.

1.1.3 Research directions

Software systems is the part of computer science that lays the foundation for the practical use of information science and technology. We interpret the term broadly to include all areas of computer science and related disciplines that contribute to the design, analysis, implementation, and evaluation of software-based systems. Thus, we include research in the design and implementation of dependable systems, information retrieval and data science, distributed systems and networks, embedded and cyber-physical systems, programming languages and programming systems, security and privacy, software engineering and verification, social computing systems and human-computer interaction, and theoretical foundations in logic and algorithms. Across these areas, we emphasize collaboration and combine theory, empirical, and data-driven methodologies to address fundamental challenges in software systems.

Much of the institute's work is currently focused on the following broad research objectives; they are each being addressed by multiple groups within the institute and involve external collaborations:

- **Dependable software systems:** Software systems are central to most pillars or our modern society, including industry, business, finance, government, democracy, education, personal productivity and entertainment. Software is increasingly complex, subject to ever-shorter

design and release cycles, composed from many different components written in different languages at different levels of abstraction, must execute on distributed, heterogeneous, multi-core hardware with weak memory models, and must resist security attacks. Developing design and programming methodologies, analysis, verification, and testing technologies that enable cost-effective design and verification of dependable software systems remains a key challenge addressed by the institute.

- **Dependable Cyber-Physical Systems (CPS):** A cyber-physical system combines computation and communication with physical processes. Cyber-physical systems are ubiquitous, and range from large-scale infrastructure (energy or resource distribution networks, civil infrastructure) to healthcare management to (semi-)autonomous control systems such as automobiles, airplanes, or robots. Such systems are subject to stringent timing and resource constraints as well as uncertainties in the operating environment. Nevertheless, they must operate safely and reliably. The institute seeks a comprehensive foundational understanding of design, implementation, and analysis of cyber-physical systems that ensures end-to-end behaviors. Recent research in CPS has focused on fundamental problems in linear dynamical systems, abstraction-based control design, real-time scheduling on multicore architectures, analysis and verification of numerical software, programming models for robotics, and computational fabrication.

- **Social information systems:** Societal-scale systems like Facebook, Google, Amazon, Uber, Coursera etc., are rapidly transforming the media landscape, trade, education, personal and corporate communication, as well as political discourse. In these systems, algorithms trained on users' past behavior increasingly determine what news and information users get to see, who they meet, and what goods and services they are offered at what price. The capability to capture, predict and influence users' behavior, awareness, and opportunities, in the hands of large corporations and governments, raises fundamental questions about freedom, transparency, fairness, bias, and potential discrimination. Social computing research at MPI-SWS focuses on developing computational methods for processing and analyzing large scale social data, aiming to uncover complex social behavior, and to inform the design of human-centered systems. Recent research has focussed on understanding the dissemination of information, ideas, and

influence over social networks, enhancing fairness and transparency of machine learning-based decision making systems, and modeling and steering the learning behaviors of humans towards social good.

- **Privacy-preserving systems:** The advent of Cloud-scale computing and storage, social media, mobile computing and sensing, combined with advances in algorithms and statistical learning, have enabled the capture, transmission, aggregation, search, and mining of vast amounts of digital information, and have placed this information at the fingertips of corporations, governments, and individuals. This technology has ushered in the era of Big Data with its fantastic new opportunities for knowledge extraction, optimization, and personalization, but has also created unprecedented new threats to citizens' privacy and freedom. Understanding these threats and devising practical technologies to effectively mitigate them is a key challenge addressed by the institute. Recent research has focussed on private analytics, digital capture privacy, and privacy in mobile systems.

These and other challenges are of fundamental importance to society, and are inadequately addressed by either industrial research (which tends to be focused on new functionality and near-term solutions to emerging challenges) or university research (where it is more difficult to quickly build up significant strengths in emerging areas, especially when cutting across traditional academic silos).

As a leading research institute in software systems, we emphasize a research environment conducive to long-term, fundamental research on these and other challenges. In particular, we continue to hire faculty who are, individually and as a group, well positioned to address broad challenges in software systems.

1.2 The state of the institute, and recent accomplishments

In this section, we briefly summarize the state of the institute, as well as some key statistics concerning our research output, notable accolades, etc.

Personnel. During the reporting period, the institute has hired a new scientific director: Joel Ouaknine (algorithms, theory, and logic), three new tenure-track faculty: Eva Darulova (program analysis and cyber-physical systems), Maria Christakis (software engineering: testing and verification), and Adish Singla (applied machine learning and social computing). Three faculty members, Björn Brandenburg, Deepak Garg, and Viktor Vafeiadis,

received tenure. The institute has also recruited 9 new postdoctoral researchers and 18 new doctoral students.

Publications and talks. MPI-SWS has produced 292 peer-reviewed publications during the reporting period. We have significant and consistent presence in top-tier venues in multiple sub-areas in computer science (see individual group sections for details). MPI-SWS faculty gave invited talks at 18 conferences and 31 workshops. We collectively served as program chairs or co-chairs for 13 conferences and 6 workshops. MPI-SWS faculty and postdocs have served on the PCs of 135 conferences and workshops. The details are given in the individual group sections.

Awards and honors. Institute researchers have won numerous awards in the reporting period. We highlight a selection of awards below.

- Mislove, Marcon, Gummadi, Druschel, and Bhattacharjee received the SIGCOMM Test of Time Award in 2017.
- Dreyer received the 2017 ACM SIGPLAN Robin Milner Young Researcher Award, the highest international accolade granted to mid-career researchers in the area of programming languages. He gave the associated Milner lecture at POPL 2018.
- Druschel received the Microsoft Research Outstanding Collaborator Award in 2016 and the EuroSys Lifetime Achievement Award in 2017.
- Pouly, postdoc of Ouaknine, received the Ackermann Award in 2017.
- Nasri won a post-doctoral Humboldt Fellowship.
- Dreyer and Gummadi were appointed Honorary Professors of Computer Science at Saarland University in 2017.
- Christakis has been presented with a Facebook Faculty Research Award for her research on combining static and dynamic program analysis, which also received other awards, including the EAPLS Best PhD Dissertation Award.
- Jourdan, former postdoc of Dreyer, received the 2016 Thesis Prize of the GDR GPL (French research group on programming and software engineering) for his PhD thesis, “Verasco: A Formally Verified C Static Analyzer”.

In addition, institute researchers have obtained 15 "best paper" or "distinguished paper" awards and multiple nominations.

External grant funding. Securing external funding is one way to demonstrate the scope and timeliness of our research projects. In particular, the ERC research awards are increasingly used as an yardstick for scientific excellence both for individuals and for institutions across Europe.

Although the institute provides its faculty members with base funding to run their research groups, we actively encourage all faculty to seek external funding. Securing such funding is important not only in terms of bringing additional resources to the institute, but also in providing junior faculty with grant-writing experience that will be essential for their future careers.

Two faculty members received ERC Consolidator Grants in the reporting period: Derek Dreyer received the award for the project "RustBelt: Logical Foundations for the Future of Safe Systems Programming" (€1.950.000) and Joel Ouaknine for the project "Analysis, Verification, and Synthesis of Infinite-State Systems" (€1.835.000). Further, we have recently learnt that Krishna Gummadi has received an ERC Advanced Grant for 2018. In addition, Druschel and Majumdar have an ongoing ERC Synergy award.

Björn Brandenburg received funding from DFG (as part of the bilateral ANR-DFG program) for "RT-Proofs: Formal Proofs for Real-Time Systems". Maria Christakis received funding as part of the Facebook Faculty Research Award. Eva Darulova obtained a DFG grant titled "Automated Rigorous Verification and Synthesis of Approximations" in October 2017. David Swasey was funded by a Microsoft Research PhD Fellowship. Garg's research has been funded by a grant from the DFG. Druschel, Francis, Garg, Gomez-Rodriguez, and Gummadi serve as co-PIs in Saarland University's Collaborative Research Center on Methods and Tools for Understanding and Controlling Privacy, a DFG research center. Gummadi's research has been partially funded by industry grants from Data Transparency Lab and AT&T research. Viktor Vafeiadis group's research has been partially funded by the European Commission's FP7 FET young explorers grant ADVENT (April 2013 – April 2016). Rupak Majumdar has been the recipient of a Toyota Research Contract (2013–2018). Druschel also serves as PI on the Saarbruecken Graduate School and the MMCI Cluster of Excellence at Saarland University.

Teaching. While teaching is not a formal requirement for institute faculty, we strongly encourage faculty to regularly teach courses regardless. We view

teaching as an important contribution to the local university ecosystem and an important endeavor both in terms of training our doctoral students and in ensuring that our faculty are well-prepared for any future positions they may hold at other academic institutions. During this review period, institute faculty taught 16 courses, 8 of them core courses. (For further details, see Section 15.8.)

Part I
Current Research Groups

2 The Real-Time Systems Group

2.1 Overview

The report covers the period August 2015 – January 2018. The Real-Time Systems Group's efforts are centered on the theoretical foundations and practical challenges of building temporally predictable computing systems.

Personnel. The group is led by **Björn Brandenburg** and currently consists of one postdoctoral fellow (Mitra Nasri), who joined the group in July 2016, and three graduate students (Arpan Gujarati, Manohar Vanga, and Felipe Cerqueira). A fourth doctoral student (Alexander Wieder) defended his dissertation in December 2017 and is now employed by Huawei Technologies. Three (non-doctoral) graduate research assistants joined the group while working towards their Master's theses: Mahircan Gül (TU KL, December 2014–February 2016), Cosmin Marin (TU KL, November 2016–), and Elena Lucherini (Scuola Superiore Sant'Anna, Pisa, Italy, December 2016–May 2017). Malte Appel (UdS, May 2017–) joined the group as an undergraduate research assistant, working primarily with Arpan Gujarati, and is expected to start work on his Bachelor's thesis in early 2018. Alessandro Biondi (Scuola Superiore Sant'Anna, Pisa, Italy) visited the group from December 2015 until May 2016. Finally, four undergraduate research interns visited the group in the reporting period.

Collaborations. The group has collaborated with researchers in both industry and academia. A collaboration with SYSGO AG on the problem of integrating support for latency-sensitive, low-criticality workloads into existing certified real-time operating systems for high-criticality applications resulted in a paper presented at RTNS'17 [309]. Further, there is an ongoing collaboration with Bosch Corporate Research (the group of Arne Hamann) centered on OS support for consolidated automotive workloads. A joint project with Microsoft Research on the problem of horizontal on-demand scaling of compute infrastructure for *machine learning as a service* (MLaaS) workloads, which resulted from Arpan Gujarati's internship at their location in Redmond, WA, has resulted in a publication accepted at Middleware'17 [161]. A joint paper with Vincenzo Bonifaci (IASI–CNR, Italy), Gianlorenzo D'Angelo (Gran Sasso Science Institute, Italy), and Alberto Marchetti-Spaccamela (Sapienza Universita' di Roma, Italy) on the problem of scheduling real-time workloads with arbitrary processor affinity restrictions on multicore platforms was published at ECRTS'16 [57]. A collabo-

ration with Jian-Jia Chen (TU Dortmund) resulted in a journal paper [76] and a tech report [77] (currently in submission). Most recently, collaborations with, respectively, Rob Davis (U. of York) and Gabriel Parmer (George Washington University) have resulted in two papers at RTAS'18 [239, 271].

Publications. In the reporting period, group members published in total 15 conferences papers at RTSS'15 [160, 316], ECRTS'16 [52, 57, 69], RTNS'16 [230, 241], RTSS'16 [53, 62], RTAS'17 [238, 256], RTNS'17 [309], ICIP'17 [149], RTSS'17 [237], and Middleware'17 [161]. Two additional papers have been accepted at RTAS'18 (but have not yet appeared) [239, 271]. Additionally, group members contributed six papers to the following workshops: RTSOPS'16 [242], JRWRTC'16 [172], CRTS'16 [236], RTSOPS'17 [240], and CERTS'17 [19, 162]. Finally, the group published one journal paper [76], and a collaborative review of self-suspensions in real-time systems is currently under review [77].

Awards and fellowships. During the reporting periods, members of the Real-Time Systems group received the following awards. Mitra Nasri won a post-doctoral Humboldt Fellowship. In July 2016, Cerqueira et al.'s work [69] on mechanized proofs for real-time systems was recognized with the ECRTS'16 Best Paper award. In September 2016, Mitra Nasri won the RTNS'16 Best Paper award for her work in collaboration with Mohaqeqi et al. [230] on the problem of finding optimal harmonic periods for real-time control tasks. In December 2016, Brandenburg and Gül's work [62] on practical, empirically near-optimal multiprocessor real-time scheduling was recognized with the RTSS'16 Best Paper award. In April 2017, Nasri and Brandenburg's work on space-, overhead-, and schedulability-efficient non-preemptive scheduling [238] was recognized with an outstanding-paper award at RTAS'17. In April 2017, Patel et al.'s paper on a mechanism for avoiding timer interference in real-time operating systems [256] was recognized with the RTAS'17 Best Paper award. (The first author of the paper, Pratyush Patel, was an undergraduate research intern in the group from May until August 2016.) In December 2017, Arpan Gujarati received the Best Student Paper award at Middleware'17 for his paper on resource-efficient, distributed autoscaling for "machine learning as a service" providers [161].

Software, tools, and data. The group maintains three primary open-source projects: **(i)** LITMUS$^{\mathrm{RT}}$ (`http://www.litmus-rt.org/`, since 2006), a multiprocessor real-time extension of the Linux kernel; **(ii)** SchedCAT

(`https://people.mpi-sws.org/~bbb/projects/schedcat`, since 2011), a schedulability analysis toolkit; and **(iii)** Prosa (`http://prosa.mpi-sws.org`, since 2016), a Coq library for mechanized schedulability analysis. Each of these projects has seen substantial improvements and new releases in the reporting period, and has played a major role in a number of the group's publications: LITMUS$^{\mathrm{RT}}$ in [57, 62, 309], SchedCAT in [52, 53, 62, 160, 316], and Prosa in [69]. Additionally, the group has released code, data, and workloads for a number of publications [316, 53, 62, 52, 57, 256, 237, 309, 238, 161].

Teaching. Björn Brandenburg co-taught (with Peter Druschel) the graduate operating systems course at University of Saarland in both the Winter semester 2015/2016 and the Winter semester 2017/2018.

External funding. Mitra Nasri is funded by a post-doctoral Humboldt Fellowship (July 2016–July 2018). Björn Brandenburg, together with co-PIs Prof. Rolf Ernst (TU Braunschweig), Dr. Sophie Quinton (INRIA Rhone-Alpes, Grenoble), Dr. Jean-Francois Monin (Verimag, Grenoble), Dr. Pierre Roux (ONERA, Toulouse), and Dr. Marc Boyer (ONERA, Toulouse), has been awarded a bilateral French-German grant, funded by ANR in France and DFG in Germany. The funding rate of the call in the area of computer science was less than 13% (6 out of 47). Of the total funds of approximately 770,000 EUR (over three years), about 190,000 EUR will support work at MPI-SWS. The project officially starts in early 2018 (see future work).

Invited talks. In the reporting period, Björn Brandenburg gave an invited keynote talk at the 5$^{\mathrm{th}}$ Brazilian Symposium on Computing Systems Engineering (SBESC'15) as well as minor invited talks and contributions at Lockheed Martin's "Embedded Computing Community of Practice" (August 2015), ECRTS'16, CAIRES'16, and TuToR'16.

Service. Björn Brandenburg was PC co-chair of EMSOFT'17, is currently serving as PC chair of EMSOFT'18, and is an associate editor of ACM TECS. Members of the group have further served as publication chair of ECRTS'17, publicity chairs of RTSS'15–'17, as reviewers for various journals, and served on the PCs of RTSS'16, ECRTS'16–'17, RTAS'16–'18, EuroSys'16, EMSOFT'16, SYSTOR'16, RTNS'16, and various work-in-progress tracks and minor events. At MPI-SWS, group members have been involved in PhD, post-doc, and faculty recruiting, MPI-SWS's program for

KL Science Night, the 2017 summer school, and various reading groups and seminars. Björn Brandenburg is the institute's current CPTS representative.

2.2 Research agenda

The Real-Time System Group's research activities in the reporting timeframe can be categorized into two topic areas, which will be discussed in sequence: **(i)** *the design and implementation of real-time operating systems* and **(ii)** *static analyses and algorithmic foundations for real-time systems.*

Real-time operating systems. The central research theme of the group continues to be analytically sound real-time operating systems, and the reporting period saw the completion of a number of projects in this area.

The *TimerShield* project [256] explored the impact of *timer interference* in real-time operating systems with support for high-resolution timers (such as the LAPIC timer on x86 platforms or various core-local timers on ARM SoCs). OS kernels must multiplex (many) *software* timers onto a small number of *hardware* timers (usually one), and to this end typically program the available hardware timer simply to fire at the expiration time of the next-earliest software timer. As a result, the execution of a high-priority real-time task may be briefly delayed while the kernel handles the hardware timer interrupt corresponding to an event of interest only to a lower-priority real-time or best-effort background task—a form of priority inversion that implies a lack of strong temporal isolation. Using Linux with the PREEMPT_RT real-time patch as a case study, Patel et al. [256] showed that this type of interference can accumulate to significant delays in the presence of only a modest number of periodic real-time tasks that use POSIX's `clock_nanosleep()` interface to control activations (which is the normal way of achieving periodic activations). As a solution, they proposed *TimerShield*, a new priority-aware high-resolution timer subsystem that eliminates all timer interference. The key technique in TimerShield is to reprogram the hardware timer as part of each context switch while *masking* all software timers belonging to lower-priority tasks (or threads); the key challenge is to make this fast as the context-switch path is extremely performance-sensitive and because there can be potentially hundreds of software timers spread across 100 priority levels (in Linux, even more in other RTOSs). To this end, TimerShield adopted segment trees to quickly execute range queries across the pending timers, which was shown to incur only a small increase in overheads in return for the complete elimination of timer interference. This work was recognized with the RTAS'17 best-paper award.

In work targeting *mixed-criticality systems* (i.e., systems hosting both highly critical tasks and tasks implementing less essential functionality), as part of a collaboration with researchers at SYSGO AG (the company behind *PikeOS*, a certified RTOS for safety-critical applications), Vanga et al. [309] explored the question of how to incorporate support for *latency-sensitive low-criticality* workloads into a time-triggered RTOS traditionally aimed at highly critical applications (such as PikeOS). Whereas most prior academic work on mixed-criticality systems has focused on guarantees for high-criticality tasks (under what has been criticized as somewhat unrealistic assumptions) while considering low-criticality tasks as largely "expendable," in practice, the challenge is not so much to guarantee that high-criticality tasks are isolated from interference from other components (which anyway has been required for critical avionics workloads for many years), but rather to provide acceptable performance for low-criticality tasks despite the employed isolation techniques. While a detailed description of the proposed solution is beyond the scope of this report, it is worth pointing out that the work [309] places great emphasis on practicality and real-world constraints (such as support for legacy applications and integration into existing workflows), in contrast to the dominant theme in the mixed-criticality literature, which was possible only thanks to the close collaboration with SYSGO AG.

Continuing the group's established line of work on practical multiprocessor real-time scheduling, Brandenburg and Gül [62] showed how to schedule sequential real-time tasks (in particular, periodic or sporadic tasks) on multicore platforms in a way that is efficient both in analytical terms (i.e., schedulability) and practical terms (i.e., implementation complexity and runtime overheads). In theory, it is desirable for the scheduler to make *optimal* allocation decisions, in the sense that no real-time task misses a deadline unless the workload is infeasible. Practical real-time scheduling, however, is a tradeoff between making good decisions and making decisions *quickly*—RTOS overheads must be low. While several optimal multiprocessor real-time scheduling algorithms are known, they are typically somewhat complicated to implement and require *global coordination* in one way or another, which represents a major efficiency and scalability bottleneck. Real systems thus typically implement non-optimal schedulers. In this work [62], we identified and demonstrated that a certain, simple-to-implement *semi-partitioned* scheduler (a hybrid scheduling approach in which most tasks are statically partitioned and only a few tasks migrate according to a predetermined pattern that does not require global coordination) is empirically near-optimal when combined with novel allocation heuristics. A prototype implementation in LITMUS$^{\mathrm{RT}}$ further demonstrated that this ap-

proach incurs runtime overheads essentially as low as a purely core-local partitioned scheduler. That is, the proposed approach achieves the "best of both worlds": empirically near-optimal schedulability in theory, and a simple, easy to understand, low-overhead implementation in practice. This result was recognized with the RTSS'16 "best paper" award.

In additional projects pertaining to real-time operating systems for multicore platforms, the group has collaborated on projects exploring, respectively, scheduler support for hierarchical processor affinities [57] and scalable memory reclamation techniques [271].

Targeting platforms at the other end of the spectrum, namely deeply embedded single-core micro-controllers with just a few dozen kilobytes of flash memory and a few kilobytes of RAM (e.g., *Internet-of-Things*-class devices), Nasri et al. developed two *memory-friendly* scheduling techniques [238, 239]. These techniques, called *offline equivalence* [238] and *FIFO with offsets* [239], combine the advantages of *table-driven* (or *time-triggered*) scheduling, which is highly deterministic, but requires potentially large, offline-generated scheduling tables to be stored, with simple non-preemptive online schedulers, which are efficient with regard to memory needs and runtime overheads, but suffer from poor schedulability (i.e., they are not very good at meeting deadlines). As the name suggests, the offline-equivalence technique tweaks a simple online policy (such as rate-monotonic scheduling) so that it re-creates any given offline schedule at runtime. It does this, however, without requiring access to the full table. Rather, the system stores only a (much smaller) *table of differences* that encodes only those times where the decision of the base online policy differs from that encoded in the reference table, that is, the times at which the base policy must be overridden in order to not diverge from the reference table. The scheme was implemented and evaluated in an Arduino-based prototype, and shown to perform well in terms of both runtime overheads and memory consumption. The work [238] was recognized with an RTAS'17 "outstanding paper" award.

The FIFO-with-offsets approach, subject of an upcoming RTAS'18 paper [239], extends the offline equivalence idea with an alternative memory-friendly scheduling technique—which is based, as the name suggests, on a FIFO scheduler and the careful selection of activation offsets—that achieves even lower memory needs for most (but not all) workloads.

Static analyses and real-time foundations. In the reporting period, the group made several contributions to the analytical foundations needed for provably predictable real-time systems.

In work on the analysis of non-preemptive uniprocessor sytems, Nasri and Brandenburg presented the first exact (i.e., necessary and sufficient) schedulability test for work-conserving and non-work-conserving policies that can cope with release jitter and execution time variations. As it is based on a state-space exploration approach, it is fundamentally of exponential complexity; however, an empirical evaluation showed to scale to realistic workload sizes (up to 30 tasks with up to 100,000 jobs per hyperperiod).

A collaboration with Jian-Jia Chen of TU Dortmund led to two papers on the analysis of self-suspending real-time tasks [76, 77].

Continuing the group's established line of work on multiprocessor real-time synchronization, Yang et al. [316] published the most accurate analysis available to date for a number of semaphore (i.e., suspension-based) protocols for global schedulers, Biondi and Brandenburg [52] presented an analysis of spin locks and lock-free data structures under partitioned earliest-deadline first (EDF) scheduling, and Biondi et al. [53] published the first non-trivial worst-case blocking analysis of nested spin locks.

In work targeting *networked real-time control systems* (NCSs), Gujarati and Brandenburg [160] presented a static reliability analysis of replicated message streams on a shared CAN bus, taking into account memory corruptions, host crashes, and retransmissions, on both the temporal and logical correctness of each message. Continuing this line of work, Gujarati et al. [162] recently developed a method for accounting for robustness properties expressed as (m, k)-firm constraints during reliability analysis.

Machine-checked schedulability proofs. Last but not least, Cerqueira et al. [69] presented *Prosa*, a library and framework for mechanized schedulability proofs using the Coq proof assistant. Motivated by a number of high-profile errata in recent years, the Prosa project seeks to raise the rigor of proofs in the real-time systems community, while simultaneously ensuring a high degree of readability for researchers who are not experts in formal methods. This work was recognized with the ECRTS'16 "best paper" award.

Future work. Prosa is a new major initiative and forms the basis for the just-starting *RT-Proofs* project (see funding). The goal of the funded project is the development of a formal, verified foundation for the analysis of uni- and multiprocessor real-time systems corresponding to what one might find in an introductory textbook. This undertaking will drive a large part of the agenda of the upcoming reporting period.

In systems-oriented work, the group is currently building a *timed* key-

value store with semantics inspired by the *logical execution time (LET)* paradigm. This middleware is intended to serve as a basis for *Byzantine* fault-tolerant NCSs. The main challenge is to enable "effortless" (i.e., transparent to the control engineer) replication at as-high-as-possible control frequencies (e.g., as they arise in active vibration dampening systems).

Furthermore, there is an ongoing, multi-year effort (partially) in collaboration with Bosch targeting the design, implementation, and evaluation of a flexible processor reservation and slack-management framework in LITMUS$^{\mathrm{RT}}$. While there exists a rich literature on various processor reservation schemes, there is little prior work investigating at the kernel level how to support multiple such schemes in a *flexible*, *robust*, and *maintainable* way, and few evaluations considering actual applications.

In work on analytical foundations of predictable systems, there are plans for a *sound* (i.e., provably not optimistic) end-to-end reliability analysis of replicated NCSs taking into account both logical and temporal failures. Additionally, the group is developing generalized schedulability analyses scheduled job sets with precedence constraints under non- and limited-preemptive scheduling on both uni- and multiprocessors. Finally, in joint work with Geoffrey Nelissen (Instituto Politécnico do Porto, Portugal), the group is working on a new schedulability analysis of self-suspending tasks.

3 The Practical Formal Methods Group

3.1 Overview

The report covers the period from mid-October 2017 – January 2018. Our research aims to develop theoretical foundations and practical tools for building more reliable and usable software and increasing developer productivity. The research areas on which we focus are software engineering, programming languages, and formal methods. Specifically, we investigate topics in automatic test generation, software verification, program analysis, and empirical software engineering. Our tools and techniques explore novel ways in writing, specifying, verifying, testing, and debugging programs in order to make them more robust while at the same time improving the user experience.

Personnel. The group is led by **Maria Christakis**. During the reporting period, we have hosted a Research Immersion Lab student, and in spring and summer, we will host four interns and an MSc student.

We have offered PhD positions to an MSc student from the Albert Ludwigs University of Freiburg, Germany who has accepted to join our group in the summer, and to a BSc student from the University of California Davis, USA who applied through the Maryland Max Planck PhD Program. Additionally, we have made an offer for a post-doctoral position to a PhD student from the University of Memphis, USA. The outcome of the last two offers is still pending.

Collaborations. Externally, the group collaborates with Prof. Dr. Peter Müller's group at ETH Zurich, Switzerland, Prof. Dr. Isil Dillig's group at the University of Texas at Austin, USA, Prof. Dr. Scott Fleming's group at the University of Memphis, USA, and the Research in Software Engineering group at Microsoft Research Redmond, USA.

We also collaborate with several faculty members from Saarland University, the Technical University of Dresden, the University of Kaiserslautern, MPI for Informatics, and our own institute with the goal of submitting two funding applications to DFG.

Publications. During the reporting period, the group has published a paper at CHI'18 [165], titled "CFar: A Tool to Increase Communication, Productivity, and Review Quality in Collaborative Code Reviews". In this paper, we designed a collaborative code review system, CFar, that introduces an automated code reviewer based on program-analysis technologies. In

particular, our automated reviewer inserts issues detected by the analyses into an otherwise human-human collaborative code review. As a result, we observed that communication and productivity of programmers increased and that the quality of their code improved.

Technology transfer. The software we developed for the CHI'18 [165] paper is currently being used by various product teams at Microsoft.

Teaching. Maria will be co-teaching (with Eva Darulova) an advanced course on Program Analysis at the University of Kaiserslautern and Saarland University during the next winter semester.

External funding. The research of the group has been partially funded by a Facebook Faculty Research Award for our research on combining static and dynamic program analysis. Maria is a principal investigator in two funding applications to DFG, for a Collaborative Research Centre on "Foundations of Perspicuous Systems" (with Saarland University, the Technical University of Dresden, and MPI-INF) and a Research Training Group on "Dealing With Change in Safety-Critical Embedded Systems" (with the University of Kaiserslautern).

Invited talks, awards, and honors. During the reporting period, Maria has been presented with a Facebook Faculty Research Award for her research on combining static and dynamic program analysis, which also received other awards, including the EAPLS Best PhD Dissertation Award. She was also invited to teach at the Cornell, Maryland, Max Planck Predoctoral Research School (CMMRS) 2018, held in Saarbrücken, Germany. The title of her lectures will be "Static Program Analysis Meets Test Case Generation". Additionally, Maria attended the 59th IFIP WG2.4 Meeting on Software Implementation Technology, held in Essex, Vermont, USA, as a first-time observer and was re-invited to the next edition of the meeting.

Service. Maria is chairing the PLDI'18 Student Research Competition and the ECOOP'18 Artifact Evaluation, and has accepted to chair the ECOOP'19 Artifact Evaluation. During the reporting period, she was a PC member for VMCAI'18 and an ERC member for PLDI'18. She has also accepted to serve the Program Committees of OOPSLA'18, iFM'18, ACM Student Research Competition'18, TACAS'19, and ICSE'19.

3.2 Research agenda

Before joining MPI-SWS, Maria mainly worked on narrowing the gap between software verification and systematic test generation. Sound software verification over-approximates the set of possible program executions in order to prove the absence of errors in a program. Systematic testing, however, typically under-approximates the set of possible program executions with the purpose of proving the existence of errors in the program. Maria's work toward bridging this gap had two directions: (1) complementing verification with systematic testing, and (2) pushing systematic testing toward verification [93]. Here, we focus on work in the first direction that led to Maria's Facebook Faculty Research Award.

Modern software projects use a variety of techniques to detect program errors, such as static program analysis, that in practice do not check all possible executions of a program. These techniques often fail to verify certain properties (such as complex assertions), or they verify some program paths under unsound assumptions (such as the absence of arithmetic overflow), which might not hold on all executions of a path. Making such assumptions is customary in static program analysis to increase automation, reduce the annotation overhead for the programmer, reduce the number of false positives, or speed up the analysis. In other words, most practical static analyses sacrifice soundness in favor of other important qualities.

Despite these compromises, static analyzers find real errors in real code. However, as a result of these compromises, it is not clear what guarantees a static analysis actually provides about program correctness. This means that users who are not familiar with an analyzer's implicit compromises do not know how to interpret the absence of warnings. It is also not clear how to use systematic testing to check exactly those properties that are not soundly verified by a static analysis. Consequently, software engineers need to test their programs as if no static analysis were applied, which is inefficient, for one, because it requires large test suites.

Until Maria's work, various approaches had combined verification (in the form of static analysis) and testing, but mainly to determine whether a verification error is spurious (that is, whether a warning emitted by a verification tool is a false positive). However, these approaches do not take into account that unsound static analyses might generate false negatives (that is, they might miss errors), and therefore, do not address compromises of verifiers. In other words, testing aims to target only those program executions for which a verification error has been emitted, thus ignoring executions that have not been previously checked by a static analyzer due to its unsoundness.

To address this problem, Maria developed a technique for combining verification and systematic testing, which guides the latter not only toward those program executions for which a verification error has been emitted, but also toward those executions that unsound verification has missed. In particular, Maria proposed a tool architecture that (1) combines multiple, complementary static analyzers that check different properties and make different unsound assumptions, and (2) complements static analysis with systematic test generation to cover those properties that have not been checked statically.

A key originality of this architecture is that it makes explicit which properties have been checked statically and under which assumptions. Therefore, the correctness guarantees provided by static analyzers are documented precisely, and can guide test generation toward those properties that are not verified yet, leading to smaller and more effective test suites. These test suites will consist of a series of successful test cases that will boost the user's confidence about the correctness of their programs or concrete counterexamples that reproduce an error. Moreover, by automatically generating tests from the explicit results of static analyzers, this technique provides the user with a choice on how much effort to devote to static analysis and how much to testing. That is, the degree of static analysis is configurable and may range from zero to complete. This allows developers to stop the static verification cycle at any time, which is important in practice, where the effort that a developer can devote to static analysis is limited. By developing this architecture, Maria investigated the following scientific topics.

– *How to design an annotation language that supports both verification and systematic testing.* The main virtues of the annotations are that they are (1) simple and easy to support by a wide range of static and dynamic tools, (2) expressive, and (3) well suited for test generation [95].

– *What the compromises of mainstream verifiers are and how to make these compromises explicit.* Maria encoded most soundness compromises in a widely used, commercial static analyzer, and measured the impact of its unsound assumptions on several open source projects, which constituted the first systematic effort to document and evaluate the sources of unsoundness in an analyzer [96]. These results can guide users of static analyzers in using them fruitfully, for instance, in deciding how to complement static analysis with testing, and assist designers of static analyzers in finding good trade-offs for their tools.

– *How to combine verification and systematic testing to maximize code quality and minimize the test effort.* Maria presented a technique for effectively reducing redundancies with static analysis when complementing its verification results by test generation [97]. Her main contribution is a code instrumentation that causes test generation to abort tests that lead to verified executions, prune parts of the search space, and prioritize tests that lead to unverified executions. To increase the usability of the technique, Maria also extended the IDE of a known verifier to seamlessly integrate test generation as well as other approaches for diagnosing verification errors [94]. She investigated how to present the results of all these approaches in the IDE without overwhelming the user with too much information.

Looking ahead, our group is very enthusiastic to continue enabling developers to rely on and benefit from a wide range of tools and techniques that improve their development workflow as well as the quality of their software. This cannot be achieved by simply providing developers with yet another tool. We believe that we can have the biggest impact by leveraging both novel and existing techniques, such that they complement each other symbiotically, to streamline the software development process. Some areas for future research include the following.

Verification and testing. We will continue working at the intersection of verification and testing. In particular, we will explore how to compute the minimum set of unsound assumptions in a static analysis that compromise the sound verification of each program property. As a result, any subsequent verification or bug-finding methodology, such as test generation, would reach its maximum effectiveness. For instance, test suites would be smallest, testing times shortest, and redundancies with prior static analysis would be brought down to zero.

We also plan to reverse the integration of verification with testing. We will design a scalable program-analysis technique, which efficiently combines test case generation with online static analysis, that is, static analysis that runs at the same time as the system under test. This combination will aim to predict, at runtime, whether the remainder of the current execution of the system is robust. The novelty of the technique is that the static analysis will take as input the concrete state that is observed at runtime, and that it will be constrained to explore all possible execution paths up to a limited depth. Consequently, any abstraction (or over-approximation) of the static analysis will be bounded by these two factors, resulting in fewer false positives and more detailed warning messages.

Smart contracts. In the last few years, there have emerged several general-use, blockchain-based, distributed-computing platforms, the most popular of which is Ethereum. A key feature of Ethereum is its support for contract accounts in addition to user accounts. Like normal bank accounts, contract and user accounts store a balance and are owned by a user. A contract account, however, is not directly managed by users, but rather through code that is associated with it. Such code expresses contractual agreements between users, for instance, to implement and enforce an auction protocol. Contract accounts with their associated code and state are called smart contracts.

We are working on understanding smart contracts and explaining them to a user. Through sampling-based techniques, we are building an automaton that describes numerous transactions from the blockchain involving a target smart contract. We ensure that our automata are simple, general, but also precise enough to convey to the user the functionality of the corresponding contracts. Our ultimate goal is to determine the level of expressiveness that smart contracts require, and therefore, the types of bugs that become possible, the program-analysis techniques that detect them, and the programming languages that are suited best for writing these contracts.

Learning-based program analysis. As discussed earlier, when a static analyzer is unsound, users are unsure about how to interpret the absence of warnings, and when an analyzer is incomplete, users find investigating spurious warnings cumbersome and very time consuming.

To address these issues, we intend to automatically adapt the soundness and precision of static analyzers based on the specific runtime environment of the analyzed system. In other words, we plan to leverage data collected at runtime during execution of the system in order to tune the number of missed bugs and false positives of its static analysis. As a result, static analyses become tailored to their target software, and every source of unsoundness or imprecision is perspicuously computed from concrete runs of the system. Moreover, the program-correctness results of the analyses explicitly state under which unsound assumptions a particular piece of code is found correct, and for which generated warnings the relevant code was too opaque to automatically reason about.

We will also explore how to intelligently guide program analysis and test case generation toward optimal executions, that is, toward program executions that optimize a particular metric, such as branch coverage.

Code reviewing. In our recent CHI'18 paper, we designed a service that turns program analyzers into code-review bots, whose analysis warnings are presented to users as comments in code reviews. We will now explore whether time-management techniques, applied at code-review time, help developers in correctly classifying analysis warnings as false positives. We will investigate how to divide reviewing of analysis warnings in time-bounded tasks and how to minimize distractions in the code such that developers become more productive in addressing the detected issues.

We would also like to employ lightweight verification techniques to determine whether the changes in a code review are provably correct. Code reviews at large companies typically take several hours to complete. Often, the changes are small or irrelevant to program correctness, for instance, when introducing source code comments or additional logging. In such cases, a code-review bot could automatically sign off when it proves a changeset correct, saving authors and reviewers a significant amount of time.

4 The Automated Verification and Approximation Group

4.1 Overview

This report covers the period from August 2015 – January 2018. The research of this group focuses on the verification and optimization of numerical programs, and in particular those running on resource-constraint platforms where the tradeoff between the accuracy of a computation and its efficiency is important. In this reporting period, the group has primarily focused on building up a framework (Daisy) for this task and started several projects towards the goal of automated and sound numerical approximations.

Personnel. The group is led by **Eva Darulova** (joined MPI-SWS in Sept. 2015) and currently has two graduate students, Heiko Becker and Debasmita Lohar who joined in May 2016 and May 2017, respectively. Another graduate student, Anastasiia Izycheva, started in the group in mid 2016, but left the group in the summer of 2017 due to personal reasons with an MSc degree. She is continuing her PhD studies at the Technical University München under Prof. Helmut Seidl, with the expectation that our collaboration will continue. The group hosted seven interns during the reporting period: Saksham Sharma (May - July 2016), Einar Horn (May - August 2016), Debasmita Lohar (July - Sept. 2016), Yehia Abd Alrahman (Sept - Oct 2016), Ezequiel Postan (Sept - Dec 2016), Raphael Monat (Feb - June 2017), Robert Bastian (Sept - Nov 2017). The group has also hosted several research immersion lab students: Heiko Becker (June - Dec 2016), Anastasiia Izycheva (June - Dec 2016), Fabian Ritter (Sept - Nov 2016) and Hizbullah Abdul Aziz (Oct - Dec 2017). Of these, Heiko and Anastasiia joined the group as PhD students.

Collaborations. Internally, the group is collaborating with the Rigorous Software Engineering Group.

Externally, the group has collaborations with Magnus Myreen (Chalmers), Anthony Fox (Cambridge), Sylvie Putot and Eric Goubault (École Polytechnique), Zachary Tatlock and Pavel Panchekha (University of Washington), Anastasiia Izycheva and Helmut Seidl (TU Munich).

Publications. During the reporting period, group members have coauthored papers in TOPLAS [104] and FMCAD [173]. Additionally, papers have been accepted to appear in ICCPS 2018 and TACAS 2018.

Software, tools, and data. The software developed as part of the research of this group is all open-source and publicly available: Rosa (`github.com/malyzajko/rosa`), Daisy (`github.com/malyzajko/daisy`), FloVer (`https://gitlab.mpi-sws.org/AVA/FloVer`).

Teaching. Eva Darulova offered a seminar on "Approximate Computing: Promise or Hype?" in the summer 2016 and an advanced lecture course on "Static Program Analysis" in the summer 2017 at Saarland University.

External funding. Eva obtained a DFG grant titled "Automated Rigorous Verification and Synthesis of Approximations" in October 2017 (over 252 530 Euro).

Service. Internally, Eva Darulova served on the graduate admissions committee in 2016 and on the faculty hiring committees in 2016 and 2018. Eva has also organized or helped to organize the Kaiserslautern science night ("Nacht, die Wissen schafft") in April 2016, the Schülerinnentag (school girls day) in Sept. 2016 and 2017, the Girl's Day in April 2017 and an event for the Unicamp at Saarland University (for school girls) in August 2017. Heiko Becker also significantly helped with the Girl's Day organisation.

Eva Darulova has served as the equal opportunity officer since November 2016. In this capacity, she has helped to develop the equal opportunity website and plan.

Externally, Eva was CAV'17 workshop chair, co-organiser of Dagstuhl seminar 17352 (Aug'17), and co-organiser of PLMW at POPL'17. Eva served on the program committees of Scala'16, VMCAI'16, CC'17, WAX'17, NSV'17, Onward!'17, Scala'17, CGO'18 and PLDI'18 and on the external review committees of PLDI'16 and PLDI'17. She has also reviewed for the journals ACM Transactions on Mathematical Software, ACM TOPLAS, IEEE Transactions on Computers, Software Testing and Verification and Reliability.

4.2 Research agenda

4.2.1 Verification and Optimization of Numerical Programs

Overview Computing resources are fundamentally limited and sometimes an exact solution may not even exist. Thus, when implementing real-world systems, approximations are inevitable, as are the errors introduced by them. The magnitude of errors is problem-dependent but higher accuracy generally

comes at a cost in terms of memory, energy or runtime, effectively creating an accuracy-efficiency tradeoff. Unfortunately, the current way of programming with approximations is mostly manual, and consequently costly, error prone and often produces suboptimal results.

The current main goal of the group's research is to develop an end-to-end system which approximates numerical programs in an automated and trustworthy fashion. The programmer should write exact high-level code and our 'approximating compiler' will generate an efficient implementation satisfying a given accuracy specification. Towards this vision, the group has focused on building a framework which provides a unified basis for developing and combining different techniques, as well as several verification and optimization approaches. In this reporting period, these efforts have been mostly focused on finite-precision arithmetic, but several projects have been already planned for the future for the automated synthesis of numerical approximations which go beyond finite precision.

Verification of Accuracy (finished) One of the main challenges when dealing with numerical programs is automated, sound, and yet accurate-enough numerical error estimation. This is difficult for finite-precision arithmetic which introduces unavoidable roundoff errors and which is unintuitive due to its discrepancy with continuous real arithmetic. Prior work has developed methods which can compute upper bounds on roundoff errors for straight-line expressions. In the reporting period, the group has continued this work to handle nonlinear arithmetic accurately, determine closed-form symbolic invariants for unbounded loops, and quantify the effects of discontinuities on numerical errors, for both floating-point and fixed-point arithmetic [104]. This work has been done within a previously developed tool, but is being integrated into Daisy.

Daisy - Framework for Analysis and Optimization of Numerical Programs (ongoing) While finite-precision computations have recently garnered significant interest, most of the techniques and tools were developed independently. As a consequence, reuse and combination of the techniques is challenging and much of the underlying building blocks have been re-implemented several times. In this reporting period, the group has built a new open-source framework, named Daisy, which provides in a single tool the main building blocks for accuracy analysis of floating-point and fixed-point computations which have emerged from recent related work. Together with its modular structure and optimization methods, Daisy allows developers to

easily recombine, explore and develop new techniques.

Daisy has been successfully used internally in our group to quickly develop new techniques, demonstrating its re-usability and modularity. It has also been used as a verification backend for the unsound optimization tool Herbie; a paper about this project is currently in submission.

Accurate Computation of Relative Errors (finished) State-of-the-art static analysis tools for verifying finite-precision code compute worst-case absolute error bounds on numerical errors. These are, however, often not a good estimate of accuracy as they do not take into account the magnitude of the computed values. Relative errors, which compute errors relative to the value's magnitude, are thus preferable. Prior tools have reported relative error bounds, however, these were merely computed via absolute errors and thus not necessarily tight or more informative. Furthermore, whenever the computed value is close to zero on part of the domain, the tools do not report any relative error estimate at all.

In this project [173], we have performed the first systematic study of the quality of relative error bounds computed by today's tools. Building on this, we have extended existing techniques for the *direct* computation of relative errors. Our experiments have shown that computing relative errors directly, as opposed to via absolute errors, is often beneficial and can provide error estimates up to six orders of magnitude tighter, i.e. more accurate. We have also shown that interval subdivision, another commonly used technique to reduce over-approximations, has less benefit when computing relative errors directly, but it can help to alleviate the effects of the inherent issue of relative error estimates close to zero. This analysis is now available in Daisy.

Optimizing Finite-precision (ongoing) Finite-precision arithmetic faces an inherent tradeoff between accuracy and efficiency. The points in this tradeoff space are determined, among other factors, by different data types but also evaluation orders. I.e. the shorter a precision's bit-length, the larger the roundoff error will be, but the faster the program will run. Similarly, the fewer arithmetic operations the program performs, the faster it will run; however, the effect on the roundoff error is less clear-cut. Manually optimizing the efficiency of finite-precision programs while ensuring that results remain accurate enough is challenging, in part because the space of possible data types and evaluation orders is prohibitively large. The group has developed a fully automated and sound technique for optimizing the performance of floating-point and fixed-point arithmetic kernels. Our technique combines

rewriting and mixed-precision tuning. Rewriting searches through different evaluation orders to find one which minimizes the roundoff error at no additional runtime cost. Mixed-precision tuning assigns different finite precisions to different variables and operations and thus provides finer-grained control than uniform precision. Our experiments have shown that when these two techniques are designed and applied together, they can provide higher performance improvements than each alone.

This analysis has also been developed in Daisy and the group is continuing this work with a focus on soundly optimizing loops.

Formal Certificates of Correctness (ongoing) While automated static analysis tools are highly valuable for bounding finite-precision roundoff errors, their results are only as correct as the implementations of the static analysis tools. Formal verification by a theorem prover provides more confidence, however, this is all but infeasible for a complex tool like Daisy. Furthermore, many parts such as a heuristic search need not be formally verified; it suffices if the final result is shown to be correct.

In order to provide more confidence and thus make the results applicable in safety-critical applications, the group has developed a modular framework for *formally checking the results* of roundoff error dataflow analyses automatically. An untrusted analysis tool encodes its results in a certificate, which is then verified by our checker functions. The increased confidence results from the fact that the checker functions have been shown to be correct with respect to floating-point semantics in the theorem provers Coq and HOL4. These checkers can be run directly inside the theorem provers, however this in-logic evaluation is usually fairly slow. We have thus leveraged verified binary code generation provided by the CakeML project to obtain fast, and yet fully verified, certificate checkers.

While this project is technically separate from Daisy and can be used to verify the results of any dataflow analysis, we intend to use it as a formal backend for Daisy. This project is currently work in progress, with the main modular design and a first analysis being finished. It has also been successful in uncovering a bug in Daisy's analysis.

Probabilistic Analysis (ongoing) Approximating real arithmetic with finite-precision arithmetic becomes even more complex in the presence of discrete choice: due to round-off errors, a program may make a different decision than a real-valued ideal one would. In this case it is, however, not sufficient to consider worst-case absolute or relative errors. As errors

are virtually always present, a program would, according to a worst-case analysis, always make the wrong the decision.

In this project, we thus combine Daisy's roundoff error analysis with a probabilistic analysis of a program's values, in order to compute the *probability* of the program making an incorrect decision. This project is currently work in progress.

Future Planned Projects The previously described projects focused on verification and optimization of finite-precision arithmetic and build the foundation of our vision of an 'approximating compiler'. Having this required basis, the group is planning to extend the approximations considered to e.g. approximations of elementary functions. These functions are mostly implemented in libraries, and thus have a fixed accuracy that they provide. For many applications such high accuracy is not needed and thus presents inefficient resource usage. We plan to develop a synthesis approach which will provide numerical approximations with as much accuracy as required, but not more. Furthermore, each program usually contains several such expensive functions, another part of this project will be to distribute the 'error budget' efficiently among different call sites.

This project will be covered by the obtained DFG grant.

4.2.2 Survey Paper on Approximate Computing

The previously described project is part of a larger area of approximate computing, which aims to take advantage of the accuracy - efficiency tradeoff by various hardware and software techniques. Eva is current part of a working group whose goal is to provide a survey of the different techniques across the stack and thus help bridge the gap between the otherwise largely isolated efforts.

4.2.3 Programming Robots for Non-Experts

The group is furthermore involved in a project with the Rigorous Software Engineering Group, whose goal is to provide non-expert users programming support for robotic applications. The idea is to let users define their own domain-specific language gradually. This language is close to natural language and an NLP semantic parser is used as a sort of synthesis engine.

5 The Foundations of Programming Group

5.1 Overview

The report covers the period from August 2015 to January 2018. The research of this group focuses on the design, semantics, verification and implementation of modern programming languages and systems, with a particular emphasis on the importance of modularity in designing and reasoning about programs. During the review period, the group's research has centered primarily around the ERC-funded RustBelt project, the aim of which is to build the first formal foundations for the Rust programming language. In support of this goal, we have also made significant advances in developing foundations and applications of the Iris framework for higher-order concurrent separation logic (a joint project with collaborators from Aarhus and Delft, which we had begun in the previous review period).

Personnel. The group is led by **Derek Dreyer**, who joined the institute in January 2008 and received tenure in 2013. It currently includes five doctoral students (Hoang-Hai Dang, Ralf Jung, Jan-Oliver Kaiser, David Swasey, and Joshua Yanovski) and two postdocs (Pierre-Marie Pédrot and Azalea Raad). During the review period, the group also included doctoral students Georg Neis (now at Google Munich) and Scott Kilpatrick (now at Two Sigma), as well as postdocs Jacques-Henri Jourdan (now at CNRS) and Ori Lahav (now at Tel Aviv University), and interns Jeehoon Kang and Zhen Zhang. Raad and Lahav have been funded by Dreyer's RustBelt project but working in KL under the primary supervision of Viktor Vafeiadis. Swasey is co-advised by Deepak Garg.

Collaborations. The group has joint publications with the Software Analysis and Verification Group (Vafeiadis) and the Foundations of Computer Security Group (Garg). Externally, the group has engaged in successful collaborations with leading researchers in Europe and Asia, including: Lars Birkedal and Aleš Bizjak (Aarhus University), Chung-Kil Hur and Jeehoon Kang (Seoul National University), and Robbert Krebbers (TU Delft).

Publications. The group publishes regularly in the top conferences and journals in the field of programming languages. During the review period, group members have co-authored one article in the Journal of Functional Programming (JFP) [328], and ten papers in top conferences, including three in POPL [181, 180, 175], one in PLDI [203], two in ICFP [246, 176], two in

ESOP [191, 299], one in OOPSLA [296], and one in ECOOP [179]. (The above list omits papers that are already covered by Vafeiadis's section and do not include Dreyer as a co-author.) We also have forthcoming papers accepted to appear in ESOP [259] and JFP [177] in 2018.

Software, tools, and data. For nearly all papers published by our group, we produce mechanized proof developments using the Coq proof assistant. These mechanized proof developments help to promote reusability and maintainability of our technical results.

In addition, our group is one of the lead development teams behind the Iris framework for higher-order concurrent separation logic in Coq. We are developing Iris together with Krebbers' team at TU Delft and Birkedal's team at Aarhus University. Iris has been fundamental in enabling a number of our research efforts during this review period [176, 191, 299, 179, 296, 175], particularly our RustBelt verification [175]. See discussion of Iris below.

Teaching. Dreyer has co-taught the Semantics core course with Prof. Gert Smolka at UdS, once in Winter 2015-16 and once in Winter 2017-18.

External funding. Dreyer was awarded a 2015 ERC Consolidator Grant of 1.95 million euros, for the project "RustBelt: Logical Foundations for the Future of Safe Systems Programming". The project runs from April 2016 to March 2021.

In addition, Swasey was funded by a Microsoft Research PhD Fellowship from January 2014 to December 2016.

Invited talks, awards, and honors. Dreyer received the 2017 ACM SIGPLAN Robin Milner Young Researcher Award, the highest international accolade granted to mid-career researchers in the area of programming languages.

Dreyer gave a Milner Award keynote lecture at POPL'18, with the title "The Type Soundness Theorem That You Really Want to Prove (and Now You Can)".

Dreyer also gave a Distinguished Lecture at the University of Chicago in May 2016, and a Colloquium Lecture at Cornell University in November 2017, both on the RustBelt project. He gave invited talks at MFPS'17, FTfJP'17, and S-REPLS'16, and will give the keynote lecture at ESOP'18.

Dreyer gave a series of lectures on separation logic at the Cornell, Maryland, Max Planck Pre-doctoral Research School (CMMRS) 2017.

Dreyer has been invited repeatedly to give lectures on writing and speaking skills at several editions of PLMW (the ACM SIGPLAN Programming Languages Mentoring Workshop), both in their incarnations at POPL (2016, 2017, 2018) and ICFP (2016, 2017).

Dreyer was granted the title of Honorarprofessor of Computer Science at Saarland University in September 2017.

Jourdan received the 2016 thesis prize of the GDR GPL (French research group on programming and software engineering) for his PhD thesis, "Verasco: A Formally Verified C Static Analyzer".

Our PLDI'17 [203], ECOOP'17 [179], and OOPSLA'17 [296] papers all received distinguished paper (a.k.a. "best paper") awards.

Service. Dreyer is serving as General Chair of ICFP'19 in Berlin. He is also serving as Steering Committee Chair for PLMW, and as a member of the Steering Committee for ICFP.

In July 2017, Dreyer was appointed as an Associate Editor of ACM Transactions on Programming Languages and Systems (TOPLAS). He also continues to serve on the editorial board of the Journal of Functional Programming (JFP), and served as guest co-editor of a special 2016 issue of JFP [120] devoted to selected papers from ICFP'14.

Dreyer has served on the program committees of POPL'17 and FSCD'16, and will serve on the program committee of OOPSLA'18.

Jung served on the Artifact Evaluation Committee for CAV'17. Jourdan will serve on the PC of ITP'18. Swasey will serve on the PC of OCAP'18.

Internally within MPI-SWS, Dreyer has been serving since January 2017 as the chair of the graduate admissions committee. He is also serving as lead organizer of the Cornell, Maryland, Max Planck Pre-doctoral Research School (CMMRS) 2018.

5.2 Research agenda

Over the past decade, the research of the Foundations of Programming group has focused, in a somewhat bottom-up fashion, on developing a toolbox of reusable verification technologies, whose ultimate goal was to enable the development of semantic foundations for realistic programming languages. In particular, we made groundbreaking contributions to the study of *Kripke logical relations*—for building semantic models of higher-order stateful programs—and *concurrent separation logic*—for reasoning modularly about fine-grained concurrent programs.

In this latest review period, our toolbox has finally come of age, and we have made fundamental use of it in order to tackle a major open problem in programming languages, namely: building the first formal foundations for the safety of the Rust programming language. We will now give a high-level overview of these efforts, beginning with the work on our "toolbox", Iris.

Iris

In 2004, O'Hearn [248] and Brookes [64] pioneered *concurrent separation logic* (CSL), a simple foundation for reasoning modularly about concurrent shared-memory programs. CSL demonstrated that the core concepts of the original separation logic [272]—ownership and separation of resources—were just as useful for reasoning about concurrent programs as for sequential programs if not more so. This seminal development in program verification (which eventually received the 2016 Gödel Prize [65]), led to an outpouring of work on extensions and variants of concurrent separation logic to account for more interesting and challenging kinds of concurrent programs. Unfortunately, it also led to a situation in which, as Parkinson memorably put it [255], "there is a disturbing trend for each new library or concurrency primitive to require a new separation logic."

In 2014, in the hopes of simplifying and consolidating the fractured field of concurrent separation logic(s), we initiated the Iris project. In our POPL'15 paper [178] introducing Iris, we showed that a number of advanced CSLs could be *derived* within a more general and much simpler framework, in which the only primitive logical notions were *monoids* (for describing proof-specific protocols on shared state) and *invariants* (for enforcing those protocols). However, much work was left to do to make Iris into a fully general and practical framework for concurrent program verification.

In the present review period, we have made good on the promise of Iris, along several axes: *theory*, *implementation*, and *applications*. First of all, in our ICFP'16 [176] and ESOP'17 [191] papers, we showed how to simplify and clarify the *theoretical* foundations of Iris by reducing the built-in primitives of Iris down to a very simple core. This core consists of (1) a *base logic* comprised of higher-order BI (the assertion language of separation logic) extended with a handful of simple modalities, and (2) a generalization of Iris's "monoids" to something called *higher-order ghost state*. The latter feature enables one to define more complex protocols on shared state, wherein the very structure of the protocol depends recursively on the language of propositions. Though this mechanism may seem somewhat obscure at first glance, it was already present in more limited forms in several existing log-

ics [18, 116], and in its general form it turns out to be extremely powerful. Specifically, we showed that *the entire program specification layer* of separation logic (*i.e.,* Hoare triples)—along with Iris's notion of *invariants* on shared state—can be *encoded* directly in terms of its modal base logic + higher-order ghost state. This kind of encoding is useful because it means that one can develop soundness proofs for Iris and other separation logics at a much higher level of abstraction than was previously possible. This approach is described in depth in our forthcoming JFP article on "Iris from the ground up" [177].

At the same time as we were evolving the foundations of Iris, our collaborator Robbert Krebbers spearheaded essential advances to the *implementation* of Iris as a verification tool. The original version of Iris [178] was mechanized and proven sound in Coq, but the Coq implementation did not provide any convenient tactical support for *using* Iris to verify programs. In their POPL'17 paper [192], Krebbers *et al.* developed the "Iris proof mode", which enables one to perform interactive tactic-based proofs *within Iris* in Coq with much the same ease as one performs normal tactic-based proofs in Coq's base logic.

With the Iris proof mode in hand, we have been able to really "use Iris in anger", investigating *applications* of it to a variety of significant verification problems. (Two of the resulting papers received best-paper awards, from ECOOP'17 [179] and OOPSLA'17 [296], respectively.)

In our ECOOP'17 paper [179], we showed that, although Iris seems superficially to be only applicable to reasoning about concurrent languages with a sequentially consistent semantics, it is in fact readily applicable to programs with weak-memory (aka relaxed-memory) semantics as well! More concretely: Iris is parameterized over an operational semantics for the language under consideration, which is assumed to be an interleaving semantics (allowing threads to each take turns interacting with the shared machine state). Many would assume that "interleaving" equals "sequentially consistent", but what we showed in this paper was that the dependence on an interleaving semantics is no obstacle to reasoning about weak memory models, so long as those memory models can be formalized operationally. In particular, we instantiated Iris with an operational semantics for release-acquire (RA) consistency based on recent work by Lahav *et al.* [199] (see Vafeiadis's section of the report), and then showed how to derive within Iris higher-order variants of two separation logics for RA consistency (RSL [305] and GPS [303]), which we called iRSL and iGPS. And within those derived logics, we gave mechanized verifications for a number of challenging case studies, including the first mechanized verification of the RCU synchroniza-

tion mechanism used in Linux. (See also Vafeiadis's section of the report.)

In our OOPSLA'17 paper [296], we showed how Iris could be used to reason about object capability patterns (OCPs). OCPs are widely used in web programming to enforce data abstraction when one is linking with untrusted code, but no prior work has offered a general explanation for what they achieve. In this paper, we showed how the higher-order and abstraction facilities of Iris provided a natural fit for compositionally specifying and verifying OCPs and their clients. (See also Garg's section of the report.)

Last but not least, our POPL'18 paper [175] on RustBelt made critical use of Iris, as we describe below.

Future work on Iris. Presently, the Iris proof mode in Coq is tied to the Iris logic. This is problematic because it limits the applicability of the proof mode, even though the tactical support provided by the proof mode ought in principle to work for a variety of different separation logics. In particular, when one derives a logic (like GPS) within Iris, one cannot presently conduct interactive proofs *within* that derived logic with the same ease with which one conducts proofs within Iris itself.

In ongoing work, we are working on a generalized version of the Iris proof mode called MoSeL, in which the tactics of the proof mode are parameterized by an interface describing a modal extension of BI (the assertion language of separation logic). We have thus far instantiated MoSeL with six very different separation logics, which implement this interface in different ways. MoSeL thus promises to extend Iris's support for interactive and semi-automated proof to a much wider range of separation logics.

We are also using the Iris proof mode as a testbed for exploring a new typed tactic language for Coq called Mtac2. Mtac2 is an evolution of our previous work on Mtac [328], a monadic extension to Coq that allowed a functional style of tactic programming to be implemented directly in the typed language of Coq itself. However, despite its name, Mtac was really more of a metaprogramming language than a full-blown tactic language: compared to Coq's built-in tactic languages (OCaml and Ltac), it was missing an essential feature of of tactic programming, namely the ability to directly manipulate Coq's proof state and perform backward reasoning on it. Mtac2 combines Mtac's original support for typed metaprogramming with additional support for programming of backward-reasoning tactics in the style of Ltac. We are porting significant pieces of the Iris proof mode (IPM) to use Mtac2 instead of Ltac, both as a case study for Mtac2 and in order to improve the robustness of the IPM code.

RustBelt

A longstanding question in the design of programming languages is how to balance *safety* and *control*. C-like languages give programmers low-level control over resource management at the expense of safety, whereas Java-like languages give programmers safe high-level abstractions at the expense of control.

Rust is a new language developed at Mozilla Research that marries together the low-level flexibility of modern C++ with a strong "ownership-based" (or "substructural") type system guaranteeing type safety, memory safety, and data race freedom. As such, Rust has the potential to revolutionize systems programming, making it possible to build software systems that are *safe by construction*, without having to give up low-level control over performance. Since the 1.0 release of Rust in 2015, the language has soared in popularity, with over 100 companies having adopted the use of Rust in production code.

Unfortunately, none of Rust's safety claims (which are essential to its popularity) have been formally investigated, and it is not at all clear that they hold. To rule out data races and other common programming errors, Rust's core type system prohibits the aliasing of mutable state, but this is too restrictive for implementing many low-level data structures and synchronization mechanisms. Consequently, Rust's standard libraries make widespread internal use of `unsafe` features, which enable them to circumvent the type system when necessary. The hope is that such `unsafe` code is "safely encapsulated", so that clients of these libraries will never be able to observe any unsafe/undefined behaviors. But verifying this (or even formalizing what "safely encapsulated" means) poses a fundamental PL challenge, because the standard technology for proving safety of programming languages—the syntactic "progress and preservation" method of Wright and Felleisen [315]—is not applicable to programs that use unsafe features.

The aim of the ERC-funded RustBelt project is to build the first formal foundations for the Rust language and its safety guarantees. Our approach avoids the limitations of the Wright-Felleisen method by employing the technique of *semantic type soundness*: Using Kripke logical relations, we build a semantic model of the Rust type system which, given the interface of a Rust library, says what verification condition its implementation must satisfy in order to be deemed a safe extension to the language (even if the implementation uses `unsafe` features of Rust). The basic idea of using Kripke logical relations to prove semantic type soundness in this way is not new, but it has never been applied before to a type system or libraries remotely

as sophisticated as those of Rust.

In our "flagship" POPL'18 paper on RustBelt [175], we developed a Kripke logical-relations model for λ_{Rust}, a λ-calculus representing a significant subset of the Rust programming language, and we used this model to verify semantic soundness of both λ_{Rust} and some of the most important libraries that are used throughout the Rust ecosystem. As we describe below, much remains to be done to put Rust on a sound formal footing, but this first paper was a major step forward, two years in the making.

The key technical challenge in developing our RustBelt model was determining the right logic in which to formalize it. In most prior work, Kripke logical-relations models are formalized in ordinary higher-order logic or set theory, but those are relatively low-level formalisms, and the resulting models are known to become extremely complex and tedious to work with. For RustBelt, we instead chose a much higher-level logical framework for encoding the RustBelt model, namely Iris! Since Iris is specifically geared toward reasoning about ownership and separation in concurrent programs, and ownership and separation are fundamental concepts in the Rust type system, Iris is a perfect fit for modeling Rust.

That said, Iris alone is not a complete solution: Iris merely provides a powerful base framework, a useful set of primitives with which to derive appropriate domain-specific separation logics. In the case of RustBelt, we used Iris to derive a novel *lifetime logic*, whose primary feature is a notion of *borrow propositions* that mirrors the "borrowing" mechanism for tracking aliasing in Rust. This lifetime logic has made it possible for us to give fairly direct interpretations of a number of Rust's most semantically complex types, and to verify their soundness at a high level of abstraction.

Future work on RustBelt. Our initial work on RustBelt made a number of simplifying assumptions, which we hope to overcome over the next few years. Most obviously, we would like to extend λ_{Rust} and its semantic model to account for important and subtle Rust features we have thus far omitted, such as trait objects, panics, and automatic invocation of destructors.

More interesting is the question of the Rust memory model. In our POPL'18 paper, we gave λ_{Rust} a memory model featuring both non-atomic accesses and sequentially consistent (SC) atomic accesses, which is common in the verification literature. However, the truth is that Rust's memory model is not only more complex than that—it does not have one! To a first approximation, Rust's memory model is similar that of C/C++, in that it provides support for C++-style relaxed-memory atomic operations which

are used by some critical concurrency libraries. Unfortunately, (1) the C++ memory model is known to be flawed, (2) there is not a canonical fix, and (3) Rust's (as yet undefined) memory model is not exactly that of C++ anyway. We have made progress on this question nonetheless by pushing on several relevant problems.

First of all, in our PLDI'17 paper [203], we uncovered and repaired a previously unknown flaw in the semantics of SC accesses in the C++ memory model; and in our POPL'17 paper [180], we developed a new "promising semantics" for C++-style concurrency that avoids the major known flaw in the C++ memory model, the so-called "out of thin air" problem. Our fix for SC accesses has already been adopted by the C++ standards committee, and we are hopeful that our "promising semantics" will gain traction in the future. (See Vafeiadis's section of the report for further details.)

Secondly, in ongoing work, we are working on adapting our RustBelt verification to target the promising semantics. This involves (1) generalizing RustBelt's lifetime logic (and the proofs that rely on it) to account for the different per-thread "views" of the machine state that show up in the promising semantics, and (2) porting the RustBelt verification to use a logic that is sound for the promising semantics, based on iGPS [179].

Lastly, we are investigating challenging orthogonal questions that are relevant to defining the Rust memory model. In particular, there is an open question as to how Rust should define undefined behavior so that useful compiler optimizations based on non-aliasing assumptions remain sound even in the presence of `unsafe` code. Ralf Jung, lead author of the RustBelt paper, spent the summer of 2017 at Mozilla exploring these issues with the Rust team, and hopes to continue doing so for the last piece of his dissertation.

6 The Distributed Systems Group

6.1 Overview

This report covers the period from Aug 2015–Dec 2017. The group's research during this period has focused on the following areas: 1) Policy compliance in distributed data processing systems; 2) OS support for strong and efficient in-process isolation; and 3) privacy in mobile systems.

Personnel. The group is led by **Peter Druschel** and currently has seven graduate students: Eslam Elnikety, Viktor Erdèlyi, James Litton, Aastha Mehta, Richard Roberts, Roberta de Viti, and Anjo Vahldiek. Eslam, Aastha, and Anjo are co-advised by Deepak Garg; James and Richard are co-advised by Bobby Bhattacharjee and Dave Levin, respectively, as part of the Maryland-Max Planck Ph.D. program. Eslam, Viktor, and Anjo are expected to graduate in the first half of 2018. Anjo and Eslam have accepted offers for positions at Intel (Beaverton) and Amazon (Dresden), respectively.

Paarijaat Aditya completed his dissertation research and joined Nokia Bell Labs in Stuttgart in October 2017. Rijurekha Sen completed her post-doctoral work as a Humboldt Fellow at the institute and took a post as Assistant Professor at IIT Delhi in January 2018. Stevens Le Blond, a post-doc and then research scientist in the group, took a position as research scientist at EPFL in January 2017.

Lily Tsai graduated with an A.B./S.M. in CS from Harvard University in 2017 and is spending a year doing research in the group, supported by a Fulbright Scholarship. Nuno Duarte finished his BS and MS degrees in CS at the Instituto Superior Tècnico (IST) Lisbon and spends 6-months doing research in the group.

Prof. Lorenzo Alvisi (University of Texas at Austin and Cornell University) has visited the group (and the institute) again in the summers of 2016 and 2017, supported by a Humboldt Research Award. Prof. Bobby Bhattacharjee (University of Maryland, College Park) visited the group and institute during his sabbatical in 2016.

Collaborations. Internally, the group has collaborated with the groups of Deepak Garg, Björn Brandenburg, Manuel Gomez-Rodriguez, and Krishna Gummadi. Externally, we have worked with colleagues at the University of Maryland, Northeastern University, Technical University of Dresden, University of Edinburgh, UCSD, Berkeley, George Mason University, New York University, Purdue University, and the Max Planck Institute for Informatics.

Publications. Group members have co-authored papers that appear at OSDI [211], Mobisys [11, 166, 210, 129], Usenix Security [126, 227, 249], NDSS [54], Oakland [138], and Eurosys [189]. Several paper submissions are under review [301, 306, 125].

Teaching. Peter Druschel taught the core course on Operating Systems in 2015/16 and 2017/18 (jointly with Björn Brandenburg) and a core course on Distributed Systems in 2016/2017 (jointly with Krishna Gummadi). Both courses were offered at Saarland University and the TU Kaiserslautern.

External funding. Druschel is a co-PI in Saarland University's MMCI Cluster of Excellence and the Saarbrücken Graduate School in Computer Science, funded by the German National Science Foundation (DFG, €45M, 2013–2019). He is also a co-PI in Saarland University's Collaborative Research Center on Methods and Tools for Understanding and Controlling Privacy, funded by the DFG (€8.5M, 2016–2020). From 2011-2017, he was co-PI and assistant director of the Center for Information Security, Privacy and Trust, funded by the German ministry of science (BMBF, €21M). Jointly with Rupak Majumdar, Michael Backes (CISPA), and Gerhard Weikum (MPI for Informatics), Druschel is a co-PIs on an ERC Synergy Grant on Privacy, Accountability, Compliance, and Trust in the Internet (€9.25M, 2015–2021).

Awards and invited talks. Druschel received the Microsoft Research Outstanding Collaborator Award in 2016 and the EuroSys Lifetime Achievement Award in 2017. A paper he co-authored with Krishna Gummadi, Bobby Bhattacharjee and students received the SIGCOMM Test of Time Award in 2017. Druschel had the honor of delivering a distinguished lecture at the University of Texas, Austin, in 2016 and the Gerard Salton Memorial Lecture at Cornell University in 2017. He lectured at the Portugal/UT Austin Summer School in Distributed Computing in September 2015. Aastha Mehta was invited to the 2016 Heidelberg Laureates Forum.

Research community service. Peter serves on the editorial boards of the Communications of the ACM (CACM) and the Royal Society Open Science Journal through 2017. He served on the Technical Advisory Board (TAB) of Microsoft Research, Cambridge, through 2016 and he continues to serve on the TAB of Microsoft Research, India. He also serves on the scien-

tific committee of the Laboratory on Information, Networking and Communication Sciences (LINCS), Paris.

He was a member of the selection committee for the EuroSys Jochen Liedtke Young Researcher Award in 2016 and chaired that committee in 2017. He was a member of the ACM SIGOPS Mark Weiser Award Committee in 2016 and 2017, and will chair that committee in 2018. He was a member of the SIGCOMM Lifetime Award Committee in 2016. Peter also served on the program committees of OSDI in 2016, SOSP and HotMobile in 2017.

Institute service. Peter served as the institute's managing director during July 2014–June 2016. During the reporting period, he initiated the Maryland-Max Planck Ph.D. program in Computer Science with Bobby Bhattacharjee and he serves as the program's co-director. Jointly with Lorenzo Alvisi and Bobby Bhattacharjee, he initiated the Cornell, Maryland, Max Planck Pre-doctoral Research School in Computer Science in 2017 and continues to serve as the school's co-director.

Jointly with Derek Dreyer and Kurt Mehlhorn at the MPI for Informatics, Peter led an effort to develop a proposal for a Max Planck Graduate Center in Computer and Information Science, which combines the strengths of all MPIs and selected faculty at German universities in CIS to attract top Ph.D. students in a combined graduate program. The proposal was funded by the MPS and will accept the first batch of students in 2019.

Max Planck Society. Peter served on the strategy committee (Perspektivenkommission) of the Chemistry, Physics, and Technology Section (CPTS) of the MPS through June 2016. He was elected to serve as the deputy chair of the CPTS starting in June 2018 and serve as the chair for a 3-year term starting in 2020.

During the reporting period, Druschel also served on two presidential committees of the MPS: The committee on the Support of Junior Scientists and the committee on IT Security. He continues to serve on the selection panel of the joint Fraunhofer/Max Planck research program.

Lastly, Druschel co-organized a Symposium on Foundations of Security and Privacy for the CPTS in July 2015, led a task force to develop a proposal for a new MPI for Cybersecurity and Privacy, and currently serves on committees to identity the location and founding directors for the institute, which is expected to start in 2018.

6.2 Research agenda

The group's research takes an empirical approach towards realizing the potential of emerging distributed and mobile systems while ensuring security and privacy. During the reporting period, the group's work has focused on policy compliance in data processing systems, on OS support for light-weight isolation, and privacy in mobile systems.

6.2.1 Compliance

In this project, we have been stuying methods to enforce declarative data usage policies in data processing systems efficiently, while relying on a small trusted computing base. This work is done in collaboration with Deepak Garg's group. Previously, we had worked on enforcing policies at the storage layer [307] and in a distributed data retrieval system [126]. During this reporting period, we were able to reduce the overhead of enforcing rich per-user policies by an order of magnitude, by using a combination of static analysis and dynamic enforcement using OS capabilities [125]. We have also shown how to efficiently enforce rich policies in database-backed data processing systems by rewriting SQL queries in the database adapter [227]. A detailed description of that work can be found in Deepak Garg's section of this report.

6.2.2 OS support for light-weight isolation

One of the technical challenges we encountered in the context of our work on compliance is the need to isolate per-session state. To effectively control the flow of information, the state of different per-user sessions must be isolated from each other. However, strong memory isolation and privilege separation have traditionally been tied to the process abstraction. As a result, isolation has been burdened with the overhead of context switching and inter-process communication. Applications that require fine-grained isolation and frequent domain switching had to resort to weaker isolation methods like language-based isolaton, software-based isolation (SFI), or address-space layout randomization (ASLR). In this project, which we conduct jointly with Deepak Garg's group as well as researchers at the University of Maryland, we aim to provide strong, hardware-based isolation with much lower cost than previously possible.

We developed *light-weight contexts (lwCs)*, the state-of-the-art in OS-based in-process isolation. Most recently, we developed *ERIM*, a system that leverages support for memory protections keys (MPK), which appeared

in recent Intel CPUs, to provide hardware-based isolation with even lower swtiching cost, because a domain switch does not require kernel intervention. These novel techniques have increased the efficiency of our compliance work, but can also benefit a much larger class of applications.

Light-weight contexts (lwCs): An OS abstraction for safety and performance We introduce a new OS abstraction—light-weight contexts (lwCs)—that provides independent units of protection, privilege, and execution state within a process [211]. A process may include several lwCs, each with possibly different views of memory, file descriptors, and access capabilities. lwCs can be used to efficiently implement roll-back (process can return to a prior recorded state), isolated address spaces (lwCs within the process may have different views of memory, e.g., isolating sensitive data from network-facing components or isolating different user sessions), and privilege separation (in-process reference monitors can arbitrate and control access).

lwCs can be implemented efficiently: the overhead of a lwC is proportional to the amount of memory exclusive to the lwC; switching lwCs is quicker than switching kernel threads within the same process. We describe the lwC abstraction and API, and an implementation of lwCs within the FreeBSD 11.0 kernel. Finally, we present an evaluation of common usage patterns, including fast roll-back, session isolation, sensitive data isolation, and in-process reference monitoring, using Apache, nginx, PHP, and OpenSSL.

ERIM: Secure and Efficient In-process Isolation with Memory Protection Keys Many applications can benefit from isolating sensitive data in a secure library. Examples include protecting cryptographic keys behind a narrow crypto API to defend against vulnerabilities like OpenSSL's Heartbleed bug. When such a library is called relatively infrequently, page-based hardware isolation can be used, because the cost of kernel-mediated or hypervisor-mediated domain switching is tolerable. However, some applications, such as isolating session keys in a web server or isolating the safe region with code pointers in code-pointer integrity (CPI), require very frequent switching. In such applications, the overheads of kernel-based or hypervisor-mediated domain switching are prohibitively high.

We developed ERIM, a novel technique that provides the security of hardware-enforced isolation with low overhead, even at high switching rates (ERIM supports up to 100,000 switches per CPU core a second with an

overhead less than 0.5%). The key idea is to combine memory protection keys (MPKs), a feature recently added to Intel CPUs that allows isolation purely in userspace, with kernel binary inspection to prevent circumvention. We show how to apply ERIM to isolate frequently accessed session keys (not just long-term keys) in nginx, a high performance web server, and how to isolate sensitive data in CPI. Our measurements indicate only a small degradation in performance, even with very high rates of switching between the untrusted application and the secure library. A paper on ERIM is currently under submission [306].

6.2.3 Privacy in mobile systems

In a third research project, we have been studying practical methods to reconcile rich functionality with user privacy in mobile systems. This work is done in collaboration with colleagues at the University of Maryland, the University of Rochester, Purdue University, and the MPI for Informatics.

iPic: Digital capture privacy The ubiquity of personal devices with built-in cameras have led to a transformation in how and when digital images are captured, shared, and archived. Photographs and videos from social gatherings, public events, and even crime scenes are commonplace online. While the spontaneity afforded by these devices have led to new personal and creative outlets, privacy concerns of bystanders (and indeed, in some cases, unwilling subjects) have remained largely unaddressed.

We have designed iPic [11], a trusted software platform that integrates digital capture with user-defined privacy. In iPic, users choose a level of privacy (e.g., image capture allowed or not) based upon social context (e.g., out in public vs. with friends vs. at work). The privacy choices of nearby users are advertised via short-range radio, and iPic-compliant capture platforms generate media *edited* to conform to the privacy choices of image subjects.

iPic relies on state-of-the-art face detection and recognition based on convolutional neural networks to associate a subject captured in an image with a privacy policy broadcast by a nearby device. We adapted these techniques for use on mobile platforms and for images of people captured incidentally. iPic also uses secure multiparty computation to ensure that users' visual features and privacy choices are not revealed publicly, regardless of whether they are the subjects of an image capture. Just as importanty, iPic preserves the ease-of-use and spontaneous nature of capture and sharing between trusted users. An experimental evaluation of iPic shows that a

practical, energy-efficient system that conforms to the privacy choices of image subjects can be built and deployed using current hardware [11].

Privacy capsules Preventing the leakage of user information via untrusted third-party apps is a key challenge in mobile privacy. We have designed and evaluated *privacy capsules (PCs)*, a platform execution model for mobile apps that prevents the flow of private information to untrusted parties by design. With PCs, apps execute in two sequential phases. In the *unsealed* phase, the app has no access to sensitive input but full access to untrusted network resources. In the *sealed* state, the untrusted app has access to sensitive input, but can no longer communicate with untrusted resources. Privacy capsules are implemented by the mobile platform, are language independent, and require few changes to apps. Using a prototype PC implementation in Android, we show that PCs have low performance and energy overhead, and are suitable for a large class of apps [166].

SeCloak: ARM TrustZone-based Mobile Peripheral Control Reliable on-off control of peripherals on smart devices is a key to security and privacy in many scenarios. Journalists want to reliably turn off radios to protect their sources during investigative reporting. Users wish to ensure cameras and microphones are reliably off during private meetings. In this paper, we present SeCloak, an ARM TrustZone-based solution that ensures reliable on-off control of peripherals even when the platform software is compromised. We design a secure kernel that co-exists with software running on mobile devices (e.g., Android and Linux) without requiring any code modifications. An Android prototype demonstrates that mobile peripherals like radios, cameras, and microphones can be controlled reliably with a very small trusted computing base and with minimal performance overhead. A paper on SeCloak appears at Mobisys 2018 [210]. In ongoing work, we are generalizing the system to provide secure I/O paths, which provide integrity and confidentiality of I/O on smart devices in the presence of a compromised platform.

Scalable positioning of commodity mobile devices We developed Sonoloc, a mobile app and system that allows a set of co-located commodity smart devices to determine their relative positions without local infrastructure. Sonoloc enables users to address each other based on their relative positions at events like meetings, talks, or conferences. This capability can, for instance, aid spontaneous and private communication among users based

on their relative position (e.g., in a given section of a room, at the same table, or in a given seat), facilitate interaction between speaker and audience in a lecture hall, and enable the distribution of materials and the collection of feedback based on users' location. Sonoloc can position any number of devices within acoustic range with a constant number of chirps emitted by a self-organized subset of devices. Our experimental evaluation shows that the system can locate up to hundreds of devices with an accuracy of tens of centimeters using up to 15 audio chirps emitted by dynamically selected devices, in actual rooms and despite substantial background noise. A paper on the design and experimental evaluation of Sonoloc appears at Mobisys 2018 [129].

Encounter-based Communication (EbC) In previous work, we had developed SDDR, an energy-efficient protocol that allows devices to form a shared secret and associated public identifier with any device in Bluetooth range [209]. We had also shown how to use these *secure encounters* to bootstrap named communication abstractions called *events* for groups of (previously) co-located users, thus enabling communication and sharing among the participants of an event, without requiring the exchange of personal information or long-term device identifiers and while leaving users in control of their privacy and the confidentiality of the information they share.

In recent work, we have built on this earlier work to develop the vision of *encounter-based communication*, which enables spontaneous, privacy-preserving, secure communication among personal smart devices and IoT devices, both during and after an encounter. EbC enables both direct communication between two devices that encountered each other, and a powerful form of group communication among devices connected by chains of encounters, subject to spatial, temporal, and causality constraints. A paper that explores the opportunities and challenges associated with EbC, and evaluates a prototype EbC system for Android that relies on the Microsoft Embedded Social platform as the Cloud back-end is currently under submission [301].

Current work: In ongoing work, we are working with researchers at the University or Maryland and Microsoft Research, Redmond, on developing the EbC vision. Specifically, we are working on developing a scalable and privacy-preserving Cloud-based implementation of EbC multi-hop messaging, and on effective means of dealing with unwanted communication. We are also planning to release the Android EbC library to mobile app develop-

ers that use the Microsoft Embedded Social platform to encourage adoption.

7 The Large Scale Internet Systems Group

7.1 Overview

The report covers the period from August 2015 - Januray 2018. The research of this (now mis-named) group is focused on a single goal: to substantially solve the 45-year old open problem of data anonymization. In order to do so, we are abandoning the formal approach that has dominated anonymization research within computer science for the last 15 years. Instead, we are taking an empirical, even entrepreneurial approach that aims to fully encompass all aspects of the problem: technical, business, regulatory, and legal. To that end, the work is being carried out in close research partnership with the startup Aircloak, for which Francis is a co-founder and MPG is an investor and share-holder.

During this period, we developed a new anonymization technique which we call Diffix. In a nutshell, Diffix parses SQL queries and adds noise and does other filtering based on the individual conditions in the query. The development has proceeded to a point where in November 2017 we launched a bounty program to break Diffix' anonymity—the first program of its kind. As part of this, we developed a way to measure the systems anonymity, tying this measure to the payout amount. The measure is based on the GDPR (the new European data protection legislation) criteria for anonymity.

Personnel. The group is led by **Paul Francis** and currently has one graduate student, Reinhard Munz. Reinhard is expected to defend in May. Ekin Akkus graduate shortly after the reporting period started.

Collaborations. The majority of our research is in research collaboration with Aircloak. The collaboration is very close, with interaction taking place on a daily basis. Besides contributing development resources and requirements (through customer interaction), Aircloak participates in all stages of design, from brainstorming to specification. There are no publishing restrictions associated with this collaboration.

The group also has a joint submission with a set of authors including another MPI-SWS group (Deepak Garg), a student at University Saarland (Fabienne Eigner), and a professor at TU Wien (Matteo Maffei, formerly at Saarland).

Publications. During the reporting period we published a single paper, in the Annual Privacy Forum (APF) 2017 [143]. The paper was joint with

Aircloak authors and student Reinhard Munz.

Software, tools, and data. The primary output of the group is the product developed by Aircloak that implements the jointly-developed Diffix anonymization approach, and the associated documentation. The product is in use commercially, and the software is proprietary. The documentation is openly available, including the details of Diffix (the system is completely transparent). As of this moment, there is no academic (low cost) license for the product, but only because no university has asked to use it: Aircloak is in support of such a license and will implement it when needed.

Through the bounty program, the implementation is openly available for testing (and attacking), and well-documented (see challenge.aircloak.com). Attackers may place their own data behind the product if they wish. The bounty program started at the end of Nov. 2017, and among the academic organizations participating are EPFL, MIT, USC (California), UCL (London), University of Quebec, INRIA, and Aarhus University.

Patents and technology transfer. During the reporting period we applied for one new patent (for Diffix). Three others that were started before the reporting period and are for the most part have either been awarded or are still under evaluation.

Press. MPI-SWS and Diffix frequently appear in marketing and press materials produced by Aircloak. It is to Aircloak's advantage to market its relationship with MPI-SWS, and to portray Diffix as the output of research done by MPI-SWS.

Invited talks

- SPMED Workshop, "A breakthrough in (anonymity X utility) for anonymized analytics", July 2016
- SciDataCon Workshop, "The Data Transparency Lab:Experiences in data sharing", Sep. 2016
- Distinguished Lecture, EPFL, " "A breakthrough in (anonymity X utility) for anonymized analytics", Dec. 2016
- Keynote, IEEE S&P Workshop on Privacy Engineering, "The Diffix Framework: Noise Revisited, Again", May 2017

- IPEN Workshop, "Diffix: High Utility Database Anonymization", June 2017
- MyData Workshop, "Diffix: GDPR-level Anonymity with High Utility Analytics", Aug. 2017
- FCG Workshop, TU Munich, "Diffix: Strong Anonymity, High Utility Analytics", Sep. 2017
- CPDP Conference, Brussles, "Anonymization: Benefits, Challenges, Innovations", Jan. 2018

Service.

- Internal
  - Managing Director (since July 2016)
  - MPG director evaluation committee for Anja Feldmann
  - Initiated MPG review of software IP licensing and participated on review panel
- External
  - NSDI 2016 PC member
  - Evaluation for NWO grant application (Netherlands Organisation for Scientific Research)
  - Evaluation for ERC SAP (Assessment of Completed Projects)
  - WWW 2017 Security Track PC member

7.2 Research agenda

For the last four years our research has been almost exclusively focused on one problem; that of designing a database query system that is on one hand strongly anonymous and on the other hand provides good analytic utility. This problem, in the modern database context, is at least 45 years old, with the earliest citation we know of being from 1972 regarding census data release (a citation that proposes adding random noise to answers).

Approach. As researchers, we are taking an unconventional approach to the problem, in two respects. First, in contradiction to the vast majority of research in this space, we are not taking a formal approach to the problem. Second, we are not carrying out this research "in the lab", so to speak. Rather we are taking an entrepreneurial approach, using a startup as a means of truly understanding the utility/anonymity requirements, and to measure the effectiveness of our solutions: whether companies are willing to pay money for the technology is one measure of success.

This approach is risky and over the short term has led to an apparent loss of productivity as traditionally measured (publications, students, grants). A few words justifying the approach are therefore in order.

Regarding not taking a formal approach (to a problem with practical goals), one needs to recognize that a formal approach to a problem necessarily confines the solution space. This may be ok when the practical requirements are clear and simple, as for instance can be the case with crypto algorithms. When the requirements are complex or varying, as is the case with data anonymization, then the gap between the simplified formal problem and the real problem can render the formal solution almost useless in practice.

This is exactly what is happening in the computer science community. The tacit assumption is that a formal approach must be taken, and among formal approaches, differential privacy is the dominant model. While differential privacy has been stunningly successful as a publication factory and PR device for Apple and a few other companies, it has produced almost nothing of practical value.

Regarding not taking a more traditional systems approach (i.e. define some requirements, design, build, and measure a system), it would be hard in this case to know where to start. There really has been nothing in place from which to draw requirements. Industry shares data all the time, but generally only with minimal effort to mask personally identifying information, and with substantial technical and contractual effort to limit distribution and use of the data. We could of course make up some reasonable-sounding requirements, but it is too easy for this to devolve to a focus on the requirements of the program committee rather than the users. Even GDPR has been unable to define clear requirements for anonymity.

Research At the time of the last SAB meeting, we had built a system and were testing it against several real applications. The experiments were largely negative: the system was hard for analysts to use and performed

poorly.

In December of 2015 we completely trashed the anonymization design and went back to the drawing board. We also decided to simplify the deployment model, focusing now on in-house deployments instead of being able to deploy on an untrusted cloud. A key enabler was the decision to restrict the kinds of queries an analyst could write. Up to that point we had assumed that the analyst would want the flexibility of writing queries in procedural code. In the end it was too hard to protect against the attacks that were subsequently enabled, and so we settled on a subset of SQL as the query language. The design has evolved over these two years as we 1) discover weaknesses in the anonymity, 2) relax the restrictions we place on the SQL in response to customer demand, and 3) fix performance issues. Although this process of evolution continues, the core design principles have so far shown themselves to be fairly robust. The resulting design (moving target though it is) is called Diffix.

A key concept in Diffix is that of understanding the *filter conditions* of the queries, and using the conditions as the basis for anonymization [143]. By way of example, the following query, which produces a histogram of ages, has two conditions, one on gender and one on age.

```
SELECT age, count(*) FROM table
WHERE gender = 'M' GROUP BY age
```

We add the sum of multiple noise samples to the outputs. Each condition contributes two kinds of noise samples. One is a *static* noise value based purely on the semantics of the condition itself (i.e. *gender = 'M'* produces the same noise sample wherever it appears). Another is a *dynamic* noise value that depends on the condition as well as the set of distinct users that comprise the reported bucket. In each case, the noise is produced by seeding a random number generator with the conditions semantics / distinct users as appropriate. An important aspect is that this noise is *sticky*. The same query produces the same noise; the noise cannot be averaged away with repeated queries.

Diffix makes a variety of decisions based on *noisy thresholds* (threshold values with added noise). These include for instance decisions to remove outliers before summing, and to suppress reporting of buckets that have too few users. These noisy thresholds also use sticky noise layers.

Of course, a key weakness of an empirical approach to secure system design is that there is always uncertainty as to whether holes remain. A commonly used way of finding holes in secure systems that are too complex

for formal analysis is a *bug bounty* program. Towards the end of 2017 we felt that Diffix was mature enough that we could launch such a program. To our knowledge, ours is the first bounty program targeted to flaws in anonymity (see `challenge.aircloak.com`).

One of the interesting challenges for the bounty program was to come up with a measure of the effectiveness of attacks that could be mapped into payout amounts. We defined attacks based on the three criteria for anonymity defined by the GDPR (singling-out, linkability, and inference). We then developed a measure that takes into account how much prior knowledge of the database contents that the attacker needs to carry out the attack, and on the confidence the attacker has in the correctness of what is learned. The former effectively captures the effect of external knowledge an attacker may have (and is generally where anonymization schemes fail). Attackers obtain a measure through attacks on real databases, and map the corresponding measure to the payout.

Our measure is empirical: the measure is based on the outcome of actual attacks. This has the clear disadvantage (compared to some formal measures) that the measure is only as good as the imagination of the attacker. On the other hand, it has the advantage of generality: it can be used to measure any anonymization scheme. As such the measure could potentially be used as the basis for certifying anonymization schemes for GDPR by European Data Protection Authorities (DPA). We have been in touch with the French National Data Protection Authority CNIL, who plan to start work on a certification program for anonymity later this year.

Moving ahead, we expect to focus on the following:

- Continue to engage the research community, both in attacking Diffix and in finding applications
- Expand the set of SQL and analytics features, especially statistical tests and machine learning algorithms
- Continue to measure the anonymity properties of Diffix
- Continue to improve performance

As a final thought, I believe that our entrepreneurial research approach to this problem will prove to be, after all, very successful. After a few years of dead ends, I believe that we now have a promising and seriously interesting technique. I am convinced that we would not have arrived here with a conventional research approach.

I think a key reason for this is the constant "reality check" that the startup imposes. There is a relentless demand for better utility: more SQL, more statistical tests, easier configuration, less distortion, better performance, and even things like integration with Tableau (which impacts what SQL is needed and therefore what attacks are possible). Without this demand, it is way to easy to soften up on utility and focus only on anonymity.

In fact we observed this first hand in the one student anonymization research project not associated with the startup (led by Reinhard Munz). The intent of the project was to design an anonymization system that is practical in that it exhibits low noise and has no query budget, but based on simple query semantics. Reinhard discovered a number of unlikely but possible attacks in the resulting design. Given a choice between trying to empirically determine the likelihood of the attacks, versus reducing utility in order to get a formally analyzable system, the latter was regarded as more likely to be publishable and therefore the more sensible way to go.

In the end, some good theoretical work was done (with Deepak Garg advising). Budgets were adopted into the system, which tracks and honors per-record privacy budgets and is unique in that it allows analysts to learn where in the parameter space budgets have been used. The system was proven to be differentially private, and the resulting paper was recently accepted into POST. In the end, however, a system with essentially no practical value was produced.

The entrepreneurial approach comes at a cost to the institute: we have not been able to incorporate students into the project (partly because of the high risk, but mostly because of a mismatch in goals), we have fewer papers than we might otherwise, and we are not contributing much to the life of the institute. Also we've been lucky that the startup has stayed focused on the research topic and not pivoted to something else. These are all things to keep in mind should anyone else decide to take the same research approach.

8 The Foundations of Computer Security Group

8.1 Overview

This report covers the activities of the FCS group from August 2015 to January 2018. During this period, the research of the group focused on software security (both theoretical foundations and practical systems), language-based security and proof systems for relational program verification.

Personnel. The group is led by Deepak Garg and currently has 8 PhD students (Mohamed Alzayat, Eslam Elnikety, Akram El-Korashy, Katura Harvey, Aastha Mehta, Vineet Rajani, Dave Swasey and Anjo Vahldiek-Oberwagner). The group also has 1 post-doc, Willard Rafnsson.

One PhD student, Ezgi Cicek, graduated during the reporting period and moved to Facebook, London to work on their Infer static analyzer. Another student, Abhishek Bichhawat (co-advised at Saarland University), has submitted his docotoral thesis and is waiting to schedule his defense. He started a post-doc at Carnegie Mellon University in February, 2018. Two other students, Anjo and Eslam, are expected to submit their theses in April 2018 and will start work at Intel and Amazon, respectively.

Marco Patrignani, a post-doc, finished his term successfully and joined the CISPA-Stanford joint program for young faculty. He will move to Stanford University as a visiting assistant professor later this year. A second post-doc, Willard Rafnsson, has just accepted a faculty offer at ITU Copenhagen and will start there at the end of April. Four students finished their Masters' theses in the group, and all subsequently continued into Ph.D. positions: Zoe Paraskevapoulou (now at Princeton University), Iulia Bastys (now at the Chalmers University of Technology), Mohamed Alzayat and Akram El-Korashy (both started their PhDs at MPI-SWS in 2016).

Collaborations. Within MPI-SWS, the group has joint publications and 4 co-advised students with the Distributed Systems group led by Peter Druschel; joint publications and 1 co-advised student with the Foundations of Programming group led by Derek Dreyer, and a forthcoming joint publication with the Large Scale Internet Systems Group led by Paul Francis.

Externally, the group collaborated with researchers at the University of Maryland (Bobby Bhattacharjee, Dave Levin), IMDEA Software Institute (Gilles Barthe), University at Buffalo SUNY (Marco Gaboardi), Chalmers University (Andrei Sabelfeld), INRIA (Tamara Rezk, Catalin Hritcu), University of Edinburgh (Roly Perera, James Cheney, Pramod Bhatotia), CMU

(Jan Hoffmann), TU-Vienna (Matteo Maffei, Florian Zuleger), Microsoft Research (Akash Lal, Aseem Rastogi), Aarhus University (Lars Birkedal), KU-Leuven (Frank Piessens, Dominique Devriese), Northeastern University (Amal Ahmed, Alan Clement) and the University of Potsdam (Christian Hammer).

Publications. During the reporting period (August 2015-January 2018), group members have co-authored papers at POPL [267, 111, 98], USENIX Security [228, 127, 250], OSDI [212], ICFP [13, 99], OOPSLA [296], CSF [258, 270, 257], CCS [92], ESORICS [49], CONCUR [261], POST [269, 56], ES-SoS [268] and LMCS [112]. Additional papers have been accepted at ESOP [12], POST [235] and EuroSys [190], but not yet appeared.

Software, tools, and data. The group's work on information flow control in web browsers has been integrated with the widely used Safari web browser and is available open source.

Teaching. Deepak Garg taught a course titled "Secure information flow control in systems" at Saarland University in the summer of 2016.

External funding. The group's research has been partially funded by two grants from the German Science Foundation, DFG. The first one on information flow control in web browsers provided approximately 240,000 EUR from 2012 to 2016. This covered one student and one intern. The second one, on language support for information flow control, is part of the broader SFB in collaboration with Saarland University, and pays for one graduate student from 2016 to 2020, and two graduate students from 2017 to 2020.

Invited talks and awards. Deepak Garg was an invited lecturer at the Shonan school "Semantics of Effects, Resources and Applications", and an invited speaker at the LORIA/INRIA security seminar, both in 2017. Dave Swasey and Deepak Garg, together with Derek Dreyer, won a distinguished paper award for their OOPSLA 2017 paper [296]. Vineet Rajani, Abhishek Bichhawat, Deepak Garg and Christian Hammer won the 2016 best paper award from the DFG priority program that funded their work on browser security.

Aastha Mehta was selected to participate in the Heidelberg Laureates Forum in 2016, and is an invited student participant at an upcoming Dagstuhl

seminar. Ezgi Cicek was an invited student participant at two Dagstuhl seminars in 2016 and 2017.

Service. Deepak Garg is the chair of the steering committee of the Foundations of Computer Security Workshop (FCS) since 2017. He is also a member of the steering committees of the IEEE Symposium on Security Foundations (CSF) since 2012 and the Conference on Principles of Security and Privacy (POST) since 2016. He chaired FCS for a second time in 2016. He and Marco Patrignani are also founding members of the new POPL-affiliated PriSC workshop on secure compilation (Deepak will chair this workshop in 2019). Deepak is also a co-organizer of a Dagstuhl seminar on secure compilation to be held in May, 2018. During the reporting period, he served on the program committees of CCS '16, EuroS&P '17 '18, POST '16 '17 and PLAS '17. He has also been the publications chair of CSF since 2012.

Willard Rafnsson served on the PC of POST 2016. Marco Patrignani served on the PC of ACM SAC's PL track in 2015, 2016 and 2017. Ezgi Cicek served on the artifact evaluation committee of POPL 2017, and the PCs of the workshop on Incremental Computing (IC) in 2017 and the workshop on Partial Evaluation and Program Manipulation (PEPM) in 2018.

Internally, Deepak has served on the graduate student recruiting committee since 2017, and managed internship applications since 2015. Ezgi was the elected student representative till September 2016. Aastha was part of a graduate student committee for selecting interns in 2015 and 2016.

8.2 Research agenda

The FCS group's research can be divided into three broad, non-exclusive themes: software security, language-based security and proof systems for program verification.

8.2.1 Software security

The group conducts research on software security from both a foundations and a systems-building perspective.

IFC4BC (Information flow control for browser clients) Web browsers routinely handle sensitive user information such as credit card numbers, passwords, browsing history, etc. Unfortunately, the standard web programming model is at odds with securing this information since it encour-

ages the inclusion of fully-privileged *untrusted* JavaScript libraries on web pages. These libraries can (and often do) leak sensitive information, either inadvertently or maliciously. The standard browser security model, which is based on coarse-grained access control, cannot prevent these leaks without denying third-party scripts access to all data, thus forcing developers to make a binary choice between data security and functionality.

The IFC4BC project aims to strike a balance between these extremes by allowing third-party scripts to access sensitive data, but *tracking* how they use the data, and enforcing web page developer-specified data disclosure policies. In the previous two reporting periods, we developed the basic foundations for doing this—a *provably sound* method for fine-grained tracking of information flows in JavaScript that is also several orders of magnitudes more efficient than the previous best solution, and its extension to native APIs of the browser (also called the DOM APIs). During the current reporting period, we built on our existing work to develop a language for specifying information flow and declassification policies on sensitive data, a programmatic API for attaching such policies to source elements on web pages, and an enforcement of these policies over our existing tracking mechanism. Our entire design works in a real web browser, Safari. The overheads of policy enforcement are moderate and not perceivable to end-users in most cases.

IFC4BC has reached its natural conclusion now. Overall, the project resulted in an open-source, usable software artifact (the modifications to the Safari browser's underlying engine), the first and only provably sound technique for information flow tracking in JavaScript that is also reasonably efficient and, most importantly, a very clear understanding of the performance limits of sound fine-grained flow tracking in an interpreted language (JavaScript). The project resulted in a total of 7 published papers (4 within this reporting period [56, 270, 49, 268]), and won the 2016 prize for the best paper within the DFG priority program that funded the project.

Policy compliance in systems This line of work develops practical techniques for enforcing privacy policies in medium- and large-scale systems. It is conducted entirely in collaboration with the group of Peter Druschel and partly in collaboration with researchers at the University of Maryland.

Many systems today handle sensitive, confidential data with different disclosure and use policies. These policies, as well as the code for enforcing them may be spread over many software components and configuration files, increasing the risk of policy violations due to bugs and misconfigurations.

The broad goal of our work is to develop both techniques to specify high-level policies separately from code and dedicated policy enforcement components, thus localizing policy-relevant code. Besides correctness, we lay emphasis on minimizing the overhead on applications—both the changes needed to code and runtime overhead.

In the last report, we described Thoth [127], a system that monitors all data flows between processes in a distributed, parallel pipeline and applies declaratively specified data disclosure and declassification policies at appropriate points. Thoth mediates all inter-process and network I/O in the OS kernel. This induces some overhead, e.g., on a search engine handling about 300 requests/s per node, the overhead on throughput is about 3%. This overhead is quite reasonable for low- and medium-scale systems (and much lower than that of other work using similar techniques), but is still somewhat excessive for large-scale systems.

During the current reporting period, we worked on scaling Thoth's policy enforcement to higher throughput systems. Our redesign, Shai, is based on the observation that the primary source of overhead in Thoth is mediation of I/O. To reduce this cost, we rely on two ideas. First, we offload as many policy checks as possible to an ahead-of-time analysis that requires only a description of the data pipeline (not its code) and a description of policies. Flows that have been checked upfront can be compiled to standard OS permission subsystems, thus eliminating the need for additional runtime mediation. Second, we isolate the reference monitor using new, efficient *in-process* isolation techniques that we have developed. This reduces the cost of the remaining interceptions. Our current prototype uses light-weight contexts [212], a page table-based isolation technique, but we have been working on an even more efficient, purely userspace isolation technique, ERIM, that relies on recent extensions of the x86-64 ISA. Overall, Shai reduces Thoth's runtime overheads by an order of magnitude or more. (Please see Peter Druschel's section of the report for a description of light-weight contexts and ERIM.)

In parallel work, Qapla [228], we enforce fine-grained, declarative policies on *structured data* in relational databases. As opposed to standard database access control mechanisms, which either support policies only at the coarse-granularity of tables and columns or require extensive schema and application changes, Qapla supports fine-grained access rules for individual rows, cells, and even data derived from operations like joins, column transformations and aggregations, with minimal application changes. Qapla policies are specified declaratively in the programmer-friendly SQL syntax along with the database schema, and are enforced by intercepting and rewriting queries

between the application and the database server. Qapla is transparent to compliant applications, has high policy expressiveness (e.g., it can express all relevant policies of the conference management system HotCRP), and has only moderate overhead on query latency.

More recently, we have been working on the prevention of side-channel leaks between co-hosted VMs in data centers. While scheduler- and cache-based side-channels have been studied extensively in prior work, we target leaks due to contention on network resources like NICs and switches. Our approach is to dynamically partition these resources across co-hosted VMs, and to shape the traffic of sensitive sessions in accordance with a policy. The policy may be adaptive: it may change very rapidly with shifts in workloads, but using public information only, which ensures that the policy itself does not leak information covertly. Our prototype implementation is based on the Xen hypervisor, and intercepts network connections just above the physical NIC driver to enforce the policy.

8.2.2 Language-based security

Language-based security refers to the use of language design and verification techniques for security. Within this space, the FCS group worked on two aspects—object capabilities and secure compilation—during the reporting period.

Object capabilities formalized Object capability patterns are programming patterns used in large-scale software projects for best-effort defenses against security vulnerabilities due to bugs. Examples include the caretaker, membrane and sealer-unsealer patterns. While there is a broad, informal understanding of the security properties that these patterns provide, hitherto, there was no precise description of these properties, nor any formal evidence that this is actually the case. In an award-winning paper [296], we develop a program logic to not only specify very general and compositional, formal specifications for these patterns, but also verify reference implementations. Our logic, OCPL, is built on the Iris proof framework (see Derek Dreyer's section for a description of Iris). The key insight is to introduce, as ghost state, a notion of "low" values—data that is safe to share with an adversary. We further show how to derive more complex patterns from simpler ones, and verify stacks of such patterns compositionally. In ongoing work, we are looking at compiling programs verified with OCPL, while preserving their security properties.

An earlier paper [270] provides a formal account of the basic use of

(object) capabilities for security. Specifically, this paper defines formally what confused-deputy attacks (the main motivation for a lot of work on capability-based security) are and, quite surprisingly, that capabilities alone may not be sufficient to prevent such attacks in all cases. Instead, one must rely on information flow tracking to prevent such attacks.

Secure compilation (preliminary work) The FCS group has recently started working quite extensively on secure compilation, an old but recently reinvigorated field of research that combines advances in programming languages, security, verification, and architectures, to devise compilers that guarantee security properties in efficient low-level code. We mention some specific directions we are working on.

First, we are looking at general, formal criteria for compiler security. This is both challenging and important because the adversary in the source language may be very different from that in the target language. Standard literature uses full abstraction (preservation and reflection of contextual equivalence) as the standard criterion, but we show in a recent paper that full abstraction can be a misfit for security since it may not even imply preservation of obvious safety properties [258] and, yet, may unnecessarily force inefficiency in compiled code. Instead, we are looking at security criteria that directly imply the preservation of classes of trace properties and their generalization, hyperproperties, in the presence of co-linked adversarial code. We are interested in both efficient compilation techniques and proof techniques.

Second, we are examining how new hardware architectures can be used to compile source-language isolation primitives efficiently. In particular, we are studying compilation to *capability machines* that support fine-grained memory protection in hardware. We have a formalization of one such RISC-V based machine, Cheri, and are currently verifying a secure compiler to it.

Third, we are looking at compiler-based techniques for attaining specific security properties in adversarial environments. For example, in collaboration with researchers at MSR-India, we have developed a new technique for attaining weak secrecy in low-level code in the presence of very strong adversaries. In yet another line of work, we are examining semantics-preserving cross-compilation between languages that use different static techniques for enforcing information flow control (IFC). Our goal is to establish the relative expressiveness of the many static techniques for IFC proposed in literature.

8.2.3 Proof systems for program verification

Going beyond security, the FCS group works on proof systems for verifying advanced program properties. Our focus lies, in particular, on relational verification—establishing relations between pairs of runs of a program, or runs of pairs of programs.

Relational refinements In collaboration with researchers at the IMDEA Software Institute, the University at Buffalo-SUNY, and Aarhus University, we have been developing highly expressive type systems for verifying both standard (unary) and relational program properties. In a paper published last year [13], we develop UHOL and RHOL, unary and relational *refinement* type systems based on higher-order simple predicate logic (HOL). Even though these type systems are syntax-directed, they have the full expressiveness of HOL. We show how these type systems can be used as meta-frameworks to directly embed existing refinement type systems, as well other kinds of static analyses like dependency tracking. Despite their high expressiveness, UHOL and RHOL have an elementary meta-theory and very simple set-theoretic semantics. In more recent work [12], we describe how the two frameworks can be extended to a different underlying model (the topos of trees) to reason about probabilistic computations over streams.

Relational cost analysis In the last report, we described our initial work on refinement type systems for establishing the complexity of incremental computations (computations that re-use work from prior executions). Early in this reporting period, we generalized that work substantially to handle control flow changes change across runs [99].

Subsequently, we realized that the same style of type system could be used to reason about the *relative cost* of two programs more broadly. The relative cost of two programs is the difference in their execution costs. The analysis of relative cost is useful in many applications, e.g., showing that compiler optimizations actually make programs faster (relative cost is less than zero), proving that a crypto-algorithm's runtime is independent of secret inputs like keys (relative cost of two runs is zero), and establishing the sensitivity of an algorithm's cost to input changes. This led us to coin the term "relational cost analysis" for methods to establish lower and upper bounds on relative cost.

So far, we have written two papers on the subject. Our first paper [98] introduces the problem and builds a refinement type-and-effect system, called RelCost, for establishing relative cost. The type system draws on our work

for the analysis of incremental computations, and ultimately traces lineage to type systems for information flow control. Our second paper [267] combines ideas from UHOL/RHOL described above to develop a cost analysis framework that is cognizant of functional properties and supports full functional verification as well. In particular, we show the expressiveness of our framework by embedding several existing cost analyses in it, and developing newer cost analysis methods that require functional verification. The entire framework has been implemented as a shallow embedding in the Coq proof assistant. In ongoing work, we are examining type theories for cost analysis using *amortization*, wherein cost may be accounted for at a program point different from the one that actually incurs it. There is existing work in this space covering limited language constructs; in contrast, we are building a general type-theory for a full higher-order language.

9 The Networks and Machine Learning Group

9.1 Overview

The report covers the period from August 2015 – January 2018. The group's research interests are in developing machine learning and large-scale data mining methods for the analysis, modeling and control of large online social and information systems. The group is particularly interested in problems at the intersection of networks, information and society, with an emphasis on phenomena arising in the Web and social media.

Personnel. The group is led by **Manuel Gomez Rodriguez** and currently has two graduate students (Utkarsh Upadhyay, from January 2015; Behzad Tabibian, from November 2015) and one postdoctoral scholar (Abir De, from January 2018). Over the reporting period, the group also included a postdoctoral scholar (Isabel Valera, from March 2015 to February 2017) on a Humboldt scholarship, who is now a research group leader under the Minerva Fast Track Fellow program at the Max Planck Institute for Intelligent Systems. Additionally, the group hosted three interns (Joel Castellon, Negar Foroutan and Junaid Ali) and eight visiting graduate students (Ali Zarezade, Abir De, Tomasz Kusmierczyk, Michael Lukasik, Jooyeon Kim, Sandeep Soni, Emaad Manzoor and Bidisha Samanta), each typically staying for 3 months.

Collaborations. The group has collaborated with the social computing group (led by Krishna Gummadi) and the machine teaching group (led by Adish Singla). Externally, the group has collaborated with researchers at Saarland University/CISPA (Michael Backes), Max Planck Institute for Intelligent Systems (Bernhard Schölkopf, Isabel Valera), Georgia Institute of Technology (Le Song, Hongyuan Zha), Cambridge University (Adrian Weller), IIT Kharagpur (Niloy Ganguly), Sharif University (Hamid Rabiee), New York University (Martin Jankowiak) and KAIST (Alice Oh).

Publications. During the reporting period, group members have co-authored three NIPS [105, 136, 323], one AISTATS [322], three WWW [226, 297, 321], three WSDM [21, 304, 326], two KDD [122, 184], one ICDM [308], one SDM [174], one CSF [34], and one NDSS [54] conference papers. Additionally, group members have also co-authored four JMLR journal papers [123, 137, 150, 325] and one TOIS journal paper [151]. According to

Google Scholar (January 2018), Manuel accumulates 2,582 citations, his h-index is 20 and his i10-index is 23.

Software, tools, and data. The group has released companion websites for several of the publications [226, 297, 326], which typically include software packages, data and additional results.

Teaching. Manuel has taught a semester-long graduate seminar on machine learning for social and information networks at TU-KL (Fall 2015) and a 8-hour graduate seminar on "Machine learning for dynamics social network analysis", with an emphasis on temporal point processes, at the University of Sydney (January 2017), Carlos III University (May 2017) and IIT Hyderabad (December 2017).

Invited talks and awards. Manuel taught an invited tutorial at IJCAI 2017, an invited lecture at the machine learning summer school (MLSS 2017), and was an invited speaker at many top academic institutions, including Harvard University, MIT, Yale, Hebrew University of Jerusalem, Tel Aviv University, NYU, KAIST, ETH, UCL, CMU, LÉcole Polytechnique, Stanford University, EPFL, GESIS, and the Max Planck Institute for the Physics of Complex Systems. Manuel and Isabel Valera, together with Krishna Gummadi, Bilal Zafar and Adrian Weller, received a Best Paper Award Honorable Mention at WWW 2017.

Service. Internally, Manuel has served in the faculty recruiting committee in 2017. Moreover, he is managing the publication of monthly highlights on research taking place at MPI-SWS in the institute's website and will organize the Institute retreat, which will take place in September 2019. Externally, Manuel has served as Senior Program Committee (SPC) member at NIPS (2016-2017), AISTATS (2018), WSDM (2018) and SDM (2018) and as Program Committee (PC) of ICML (2016-2017), ICLR (2018), KDD (2015-2017), WSDM (2016-2017), WWW (2016-2018), ICWSM (2016-2018), SDM (2016-2017), AAAI (2016-2018), AISTATS (2016-2017) and IJCAI (2016). Moreover, he has served as a reviewer for the Netherlands Organization for Scientific Research and the Foundation for Polish Science, and as a external examiner for Ph.D. thesis at LÉcole Polytechnique, École Normale Superior Cachan, Hebrew University of Jerusalem and EPFL. Finally, Manuel has co-organized a summer school in machine learning (MLSS 2016), which received over 400 applications from academia and industry.

External funding. Postdoctoral scholar Isabel Valera was funded by a Humboldt postdoctoral scholarship.

9.2 Research agenda

The Networks and Machine Learning group develops machine learning and large-scale data mining methods for the analysis, modeling and control of large social and information online systems. The group is particularly interested in problems at the intersection of networks, information and society, with an emphasis on phenomena arising in the Web and social media.

In the reporting period, the group's research activities spanned the following broad research themes: information acquisition, information reliability, predicting and steering social processes, and fair machine learning.

9.2.1 Information acquisition

The advent of social media and social networking sites is changing dramatically the way in which people acquire information. In the Networks and Machine Learning group, we have focused on several important aspects of the information acquisition process in online settings, motivated by the following three questions:

(i) How efficient are people at curating information in social networking sites?

(ii) Can we understand the interplay between the structure of a social network and the information acquisition process?

(iii) Can we spot knowledge items (*e.g.*, questions and answers) in social media that systematically help people increase their expertise?

To answer the first question, together with researchers from the Networked Systems group led by Krishna Gummadi, we introduced a computational framework to quantify the efficiency of a user as an information curator within a social networking site [21]. The framework is general and applicable to any social networking site with an underlying information network, in which every user *follows* others to receive the information they produce. We find that social media users are sub-optimally efficient at acquiring information and this lack of efficiency is a consequence of the triadic closure mechanism by which users typically follow other users in social networking sites.

To answer the second question, in collaboration with researchers at Georgia Institute of Technology, we developed a temporal point process model for

the joint dynamics of information diffusion and network evolution [136, 137]. This model has shed light on the interplay between the structure of a social network and the information acquisition process and it accurately predicts changes in the structure of a social network triggered by the information acquisition process. Moreover, the model has also served as a showcase of the rich set of possibilities offered by temporal point processes, which had been rarely explored before in large-scale social networking modeling.

Finally, to answer the third question, we first focused on spotting specific knowledge items (*e.g.*, questions and answers) that systematically helped people increase their expertise. To this aim, we developed a probabilistic modeling framework that leverages the crowd to spot information with high (knowledge) value [304]. The key idea behind behind the framework is simple: if a knowledge item has high value, users who learn from the item will become more knowlegeable and thus their subsequent contributions will be assessed more highly by others in terms of *e.g.*, upvotes, likes or shares. Thus, by jointly modeling learning events and contributing events, the framework can identify knowledge that leads to a measurable increase in expertise, as assessed by the crowd.

Then, we focused on spotting ordered sequences of knowledge items with high knowledge value. By uncovering these sequences, one can track users' interests and goals over time, and facilitate the design of automated curriculum building and, more broadly, personalized education. To this aim, we have developed a novel modeling framework for efficiently clustering continuous-time grouped streaming data, the hierarchical Dirichlet Hawkes process (HDHP) [226], which is also one of the first example of models for complex social processes combining temporal point processes and Bayesian nonparametrics.

9.2.2 Information reliability

In the Web and social media, information is often not professionally curated and its high-quality, relevance and reliability is at stake. As a consequence, online knowledge repositories, such as Wikipedia, Stack Overflow and Quora, put in place different evaluation mechanisms to increase the reliability of their content. For example, in Wikipedia, an editor can refute a questionable, false or incomplete statement in an article by removing it. In Stack Overflow, a user can accept or up-vote the answers provided by other users. However, these evaluation mechanisms only provide noise, often biased, measurements of the reliability of information and the trustworthiness of the information sources.

In this context, we have developed a temporal point process modeling framework that leverage the above mentioned noisy evaluations to distill robust, unbiased and interpretable measures of information reliability and source trustworthiness [297]. The key idea is to disentangle to what extent a refutation (or a verification) is due to the intrinsic unreliability of the evaluated information or to the trustworthiness of the source providing the statement.

9.2.3 Predicting and steering social processes

Since I joined MPI-SWS, I started to realize that my doctoral and post-doctoral work on network inference and influence maximization leveraged particular instances of a more general and powerful type of random processes, marked temporal point processes, which could be potentially used to design predictive models and control algorithms for a wide variety of social processes taking place on the Web and social media.

Since then, I have leveraged this realization to lead the design of:

(i) a new generation of data-driven predictive models based on marked temporal point process for a wide range of social processes over social and information networks, from product competition [308] and opinion dynamics [105] to spatiotemporal processes [122, 174]. In all cases, by exploiting the increasing availability of fine-grained user data, the models provide more accurate predictions than the state of the art.

(ii) a series of efficient off-line and online algorithms with provable guarantees to steer social processes both at a user and at a global level [184, 324, 326]. These algorithms exploit an alternative representation of marked temporal point processes using stochastic differential equations (SDEs) with jumps and establish a previously unexplored connection between optimal control of SDEs with jumps and marked temporal point processes, which is of independent interest.

9.2.4 Fair machine learning

There is a growing concern that automated algorithmic decision-making, by now widely spread across a variety of online services, can lead to user discrimination, even in the absence of malicious intent. In this context, the nascent field of fair machine learning aims to develop machine learning methods whose outcomes do not have a disproportionally large adverse impact on particular groups of people sharing certain sensitive traits such a race or sex.

In collaboration with researchers from the Networked Systems group led by Krishna Gummadi, we have tackled the design of margin-based classifiers by introducing a variety of intuitive measures of decision boundary unfairness corresponding to different notions of fairness [322, 323, 321]. This work has received immediate international recognition by means of a best paper award honorable mention at the 26th International World Wide Web Conference (WWW), the flagship conference in Web research.

9.2.5 Other achievements

Together with a variety of collaborators from the Max Planck Institute for Intelligent Systems, Georgia Institute of Technology and Saarland University, Manuel has extended his doctoral work on network inference and influence maximization in a collection of journals papers [123, 150, 151] and revisited this work in the context of privacy research [34].

9.2.6 Plans for future work

In the upcoming years, my research agenda will be strongly influenced by complex social, technological and cognitive phenomena that emerge in an increasingly digital, always-on world and it will decomposed into many conceptual problems. Here, I briefly discuss four of them.

— *Optimizing human learning:* the popularization of online tutoring systems and learning platforms has increased the availability of digital traces of human learning. We will leverage these traces to design effective, personalized teaching algorithms.

— *Detecting and preventing the spread of misinformation:* the amount of misinformation in social media and online social networking sites is rampant and it is often difficult for humans and machine learning algorithms alike to decide whether a piece of news is misinformation. We will design machine learning methods with humans in the loop to effectively detect misinformation.

— *Temporal point processes and graph discovery:* modern generative models such as variational auto encoders (VAEs) and generative adversarial networks (GANs) have proven successful at generating diverse collections of realistic images when parametrized by convolutional neural networks. We will design VAEs and GANs that generate diverse collections of realistic graphs and temporal point processes when parametrized using graph convolutional networks and recurrent marked temporal point processes [122], respectively.

— *Teaching humans to be fair:* there is an exponential growth in fair machine learning, which *teach* algorithms to be fair. Here, we will also use machine learning learning to *teach* a group of decision makers to be fair.

10 The Networked Systems Group

10.1 Overview

This section describes the activities of the Networked Systems group between August 2015 and December 2017. The group's research interests are in measurement, analysis, design, and evaluation of complex Internet-scale systems. In the past, the group's studies have focussed on Internet access networks, trusted cloud computing, peer-to-peer and overlay routing systems. In recent years, the group's projects have focused on *understanding and building social computing systems.* Specifically, they tackle the challenges associated withh understanding, predicting, and controlling the behaviors of their constituent human users and computer systems.

Personnel. The group is led by **Krishna Gummadi**. It is currently comprised of five graduate students (Juhi Kulshretha from April 2011, Muhammad Bilal Zafar from October 2012, Reza Babaei from April 2014, Nina Grgic-Hlaca from November 2016 and Till Speicher from February 2017) and one postdoctoral researcher (Przemyslaw Grabowicz from October 2013). Krishna is also involved in co-advising two graduate students and a masters student (Joanna Asia Biega, a PhD student at the MPI for Informatics, Abhijnan Chakraborty, a PhD student at IIT Kharagpur, and Muhammad Ali, a masters student at University of Saarland). Przemyslaw is involved in co-advising a masters student (David Adelani).

Additionally, the group hosted three long-term visitors supported by Humboldt faculty fellowships and awards: Prof. Hakan Ferhatosmanoglu from Bilkent University, Turkey, Prof. Patrick Loiseau from EURECOM, France, and Prof. Fabricio Benevenuto from UFMG, Brazil.

During the reporting period, (i) Bimal Viswanath graduated with a PhD and joined Bell Labs as a researcher (he has since moved on to University of Chicago as a postdoctoral researcher). (ii) Mainack Mondal graduate with a PhD and joined University of Chicago as a postdoctoral researcher. (iii) Juhi is expected to finish her PhD defense in April 2018. She will be joining Lebnitz Institute for Social Sciences as a postdoctoral researcher in the Computational Social Science department. (iv) Saptarshi Ghosh, a postdoctoral researcher, joined as an Assistant Professor at the IIT Kharagpur, India. (v) Rijurekha Sen (co-advised with Peter Druschel), a postdoctoral researcher, joined as an Assistant Professor at the IIT Delhi, India. (vi) Oana Goga, a postdoctoral researcher, joined as a researcher at CNRS, France (she received a joint first ranking amongst all CS applicants for CNRS positions

in 2016). (vii) Denzil Correa, a postdoctoral researcher, joined as a data scientist at Bayer Business Services. (viii) Przemyslaw Grabowicz, a postdoctoral researcher, is on the job market this year. He has an early offer from University of Nottingham and is considering it.

Collaborations. Internally, the group members have close collaborations with the distributed, machine learning, and machine teaching groups led by Peter Druschel, Manuel Gomez-Rodriguez, and Adish Singla, respectively. Locally, the group members collaborate with the security group at University of Saarland led by Michael Backes.

Within Max Planck Society, the group members collaborate with Gerhard Weikum (MPI for Informatics), Bernhard Schoelkopf and Isabel Valera (MPI for Intelligent Systems), Christoph Engel (MPI for Collective Goods) and Emilio Zagheni (MPI for Demographic Research).

External collaborators include researchers from BU (Mark Crovella), NEU (Alan Mislove), NYU (Kathy Strandberg), CMU (Alexandra Chouldechova), UMD (Michelle Mazurek), AT&T (Balachander Krishnamurthy), INRIA (Renata Teixeira), Eurecat (Nikos Laoutaris), Cambridge (Adrian Weller), ETH Zurich (Hoda Heidari), QCRI (Ingmar Weber), UFMG (Fabricio Benevenuto), KAIST (Meeyoung Cha), IIT Kharagpur (Niloy Ganguly), and IIIT Delhi (Ponnurangam Kumaraguru).

Publications. Due to the inter-disciplinary nature of their work, the group members regularly publish their research in the top conferences, journals, and workshops in different sub-areas of computer science, and occasionally, in social sciences. Specifically, during the reporting period, group members have co-authored papers in:

1. Maching Learning, Datamining, and Information Retrieval: WWW [321, 155], AAAI [158], AISTATS [322], NIPS [323], NIPS ML & Law Symposium [156], FAT-ML [157, 140], FAT* [290], KDD [147], WSDM [21], SIGIR [51], CIKM [50], and ACM Journal of Transactions on the Web [319].

2. Social Computing and Social Media: WWW [71, 264], CSCW [195, 318], ICWSM [72, 153, 279, 70, 154, 320], ASONAM [16], and Journal of Population and Development Review [128].

3. Security and Privacy: NDSS [17], IEEE S&P [310], PETS [234], COSN [312], WWW [311], WPES [23], SOUPS [231], and IEEE Internet Computing (Special Issue on Usable Security and Privacy) [232].

4. Systems and Networking: IMC [278, 148], ICTD [281], and CCR [277].

Software, Technology Transfer, Press. During the review period, we have publicly released Python implementations of our fair (non-discriminatory) learning methods as well as deployed social media apps that allow users to check their secondary footprint (i.e., data about themselves posted by other users) as well as inferring ideological biases of social media accounts behind published news stories. These software releases have few tens to hundreds of users. As follow-up to our IEEE S&P paper [310], we worked with Facebook engineers to plug serious privacy vulnerabilities with Facebook's advertising APIs that allowed any (potentially malicious) advertiser to learn about personally identifiable information such as phone numbers or website visits of any Facebook user. Our different works on Facebook ads related to potential for discrimination [290], lack of transparency [17], and potential for privacy leakages [310], were widely covered in popular press including WIRED, The Verge, The Guardian, and the Telegraph.

Teaching. (i) Krishna Gummadi and Przemyslaw Grabowicz co-taught an advanced lecture on *social media analysis* in Summer 2016. (ii) Krishna Gummadi and Peter Druschel co-taught a core course on *distributed systems* in Winter 2017.

External funding. The research of the group has been partially funded by DFG's Collaborative Research Center grant (250,000 Euros over 4 years) on "Methods and Tools for Understanding and Controlling Privacy" as well as industry grants from Data Transparency Lab (two grants each worth 50,000 Euros) and AT&T research (one grant of 25,000 Euros). Additionally, Przemyslaw Grabowicz won a prestigious Volkswagen Foundation's inter-disciplinary grant (212,000 Euros over 4 years) on "Agenda Setting in the European Union". Oana Goga won a highly selective ANR's (French National Research Foundation's) JCJC grant for young researchers.

Awards. (i) Peter Druschel and Krishna Gummadi received the **ACM SIGCOMM Test-of-Time Award 2017** for their work published at IMC 2007 [229]. (ii) Till Speicher and Krishna Gummadi recieved a **Best Paper Nomination** at the Conference on Fairness, Accountability, and Transparency (FAT*) 2018 [290]. (iii) Bilal Zafar, Manuel Gomez-Rodriguez, and Krishna Gummadi received a **Best Paper Honarable Mention Award** at the World Wide Web Conference (WWW) 2017 [321]. (iv) Oana Goga received the **Best Paper Runner-Up Award** at the IEEE ASONAM 2017 [16]. (v) Nina Grgic-Hlaca, Bilal Zafar, and Krishna Gummadi received a **Notable Paper Award** at the NIPS Symposium on ML and Law

2016 [156]. (vi) Bimal Viswanath, Bilal Zafar, and Krishna Gummadi received the **Best Paper Award** at the Conference on Online Social Networks (COSN) 2015 [312].

Invited Talks. During the reporting period, Krishna gave keynote/invited talks at the following conferences/symposiums (the list is non exhaustive and excludes workshops): IEEE International Conference on Communication Systems and Networks 2018, European Big Data Value Forum on Transparency and Accountability of Algorithmic Systems 2017, NYU School of Law Symposium on Algorithms and Explanations 2017, NIPS Symposium on ML and Law 2016, 3rd GESIS Winter Computational Social Science Symposium 2016, Complex System Society Conference on Complex Systems 2016, Microsoft Research Symposium on Machine Learning, Law, and Policy 2016.

Krishna also gave distinguished lectures at CNRS Grenoble (Laboratorie d'Informatique) and ETH Zurich (Lecture Series on Law and Econmics of Innovation). He also gave invited lectures at the Cornell, Maryland, Max-Planck Pre-Doctoral Research School in Summer 2017 and at the Lorentz Workshop on Intersectionality and Algorithmic Discrimination in Fall 2017.

Service. Internally to MPI-SWS, Krishna lead the effort to redesign MPI-SWS website between 2015 and 2016 and served on the faculty recruiting committee for the years 2015 and 2017.

Externally, Krishna has served as a general co-chair for AAAI's ICWSM 2016 and program co-chair for the first Data Transparency Lab (DTL) Conference and Fairness, Transparency, and Privacy Workshop at DALI 2018. He has also served on the program committees of KDD 2016-2018, SIGIR 2018, ICDE 2018, WSDM 2016-2018, WWW 2016-2018, FAT-ML 2016-2017, FAT* 2018, WebScience 2016, and IMC 2016. Juhi Kulshrestha, Mainack Mondal, Oana Goga, and Przyemysalw Grabowicz have served on the program committees of ICWSM 2017-2018 and WWW 2018 conferences. Krishna also served as an associate editor of ACM Transactions on the Web between 2014 and 2017. He is currently serving as the associate editor for the ACM Transactions on Social Computing and EPJ Data-Science Journal.

Krishna currently serves as a steering committee member of the Measurement-Lab (M-Lab), Conference on Fairness, Accountability, and Transparency (FAT*), and AAAI's International Conference on the Web and Social Media (ICWSM). He also served on the selection committes for WWW 2018 Test-of-Time Award, CNIL-INRIA Privacy Research Award 2017, and Data Transparency Lab (DTL) grants 2015-2017.

10.2 Research Agenda: Foundations of Social Computing

Social computing systems are an emerging class of *societal-scale human-computer systems* that facilitate interactions and knowledge exchange between individuals, organizations, and governments in our society. Examples include social networking sites like Facebook and Google+, blogging sites like Twitter and LiveJournal, content sharing sites like YouTube and Instagram, crowdsourced opinion sites like Yelp and eBay seller ratings, and social peer production sites like Wikipedia and Amazon's Mechanical Turk.

We see social computing systems as representing a societal-scale symbiosis of humans and computer systems, i.e., networking of humans and computers, where human societal behaviors affect computations and vice-versa. Our studies of social computing systems are motivated by the challenges, opportunities, and threats that stem from the ability to understand, predict, and control (influence) the behaviors of their constituent human users and computer systems.

1. User-centric Studies: We conduct empirical studies of user behaviors and interactions in social computing systems using large-scale observational studies of deployed systems as well as smaller-scale studies of users recruited on the Web to participate in experimental systems and surveys.

2. Data-centric Studies: We construct and analyze computational models of user and system behaviors from the data we gather using appropriate data mining (e.g., graph analysis, data clustering and dimensionality reduction), statistical learning (e.g., supervised learning and convex optimization) and NLP (e.g., statistical language and topic models) techniques.

3. Systems-centric Studies: We leverage insights from our empirical and analytical studies above to design, implement and deploy useful systems and services in practice (e.g., fraud detection services, privacy management tools, diversity-preserving recommender systems).

In the past, we have conducted some of the earliest studies of the structure, growth, and evolution of large-scale online social networks and their user interactions. Our studies in the review period focussed on four themes:

1. Fairness, Bias, and Transparency in Data Driven Decision Systems
2. Privacy, Anonymity, and Exposure in Crowdsourcing Systems
3. Trustworthiness, Reputation, and Accountability of Social Identities
4. Information Dissemination and Retrieval in Social Media Systems

Due to space constraints, we will highlight only our results related to the first theme of fairness in data driven decision systems. The works related to non-discriminatory learning have been conducted in collaboration with Manuel Gomez-Rodriguez and his postdoctoral researcher, Isabel Valera.

10.2.1 Discrimination in Algorithmic Decision Making

Decision making processes are increasingly becoming automated and data-driven in both online (e.g., spam filtering, product personalization) as well as offline (e.g., pretrial risk assessment, mortgage approvals) settings. However, as automated data analysis replaces human supervision in decision making, and the scale of the analyzed data becomes "big", concerns have been raised by civil organizations, governments], and researchers about potential loss of transparency, accountability and fairness.

a. Disparate Impact and Disparate Treatment. Anti-discrimination laws in many countries prohibit unfair treatment of people based on certain attributes, also called sensitive attributes, such as gender or race. These laws typically evaluate the fairness of a decision making process by means of two distinct notions: ***disparate treatment*** and ***disparate impact***. A decision making process suffers from disparate treatment if its decisions are (partly) based on the subject's sensitive attribute information, and it has disparate impact if its outcomes disproportionately hurt (or, benefit) people with certain sensitive attribute values (*e.g.,* , females, blacks).

Controlling for both forms of unfairness *simultaneously* in learning tasks such as classification is challenging. One could avoid disparate treatment by not making use of sensitive attribute information. However, ignoring the sensitive attribute information may still lead to disparate impact in outcomes: as classifiers are trained on historical data, if a group with a certain sensitive attribute value was unfairly treated in the past, this unfairness would persist in future predictions through *indirect discrimination*, leading to disparate impact.

In our AISTATS 2017 work [322], we show how to design classifiers that avoid *both* disparate treatment and disparate impact. Our insight is to find a decision boundary that bounds our unfairness measure computed as the covariance between the sensitive attributes and the (signed) distance between the subjects' feature vectors and the boundary. Remarkably, our fair classifier formulation also avoids disparate treatment, as it does not make use of sensitive attribute information while making decisions. Our unfairness measure additionally satisfies several desirable properties: (i) for a wide variety of convex margin-based (linear and non-linear) classifiers, it is convex and can be readily incorporated in their formulations without increasing their complexity; (ii) it allows for clear mechanism to trade-off fairness and accuracy; and, (iii) it can be used to ensure fairness with respect to several sensitive attributes. Experiments with two well-known classifiers, logistic regression and support vector machines, using both synthetic and

real-world data show that our fairness measure allows for a fine-grained control of the level of fairness, often at a small cost in terms of accuracy, and provides more flexibility than the state-of-the-art.

b. Disparate Mistreatment. The notion of disparate impact is useful to tackle unfairness in scenarios where the historical decisions in the training data are biased and there is no ground truth about the correctness of the historical decisions. However, when the ground truth for historical decisions is available, disproportionately beneficial outcomes for certain sensitive attribute value groups can be justified by means of the ground truth.

For such scenarios in our WWW 2017 paper [321], we propose an alternative notion of unfairness, called **disparate mistreatment**. To avoid disparate mistreatment with respect to a given sensitive attribute (e.g., race), a classifier should ensure that its misclassification rates are *similar* for groups of people having different values of that sensitive attribute (e.g., blacks and whites). We introduce intuitive measures of disparate mistreatment for decision boundary-based classifiers and show that, for a wide variety of linear and nonlinear classifiers, these measures can be incorporated into their formulation as convex-concave constraints. The resulting formulations can be solved efficiently using recent advances in convex-concave programming. We experiment with synthetic as well as real world datasets and show that our methodology can be effectively used to avoid disparate mistreatment.

c. Beyond Parity-based Discrimination. Traditional notions of nondiscrimination call for *parity* (equality) in treatment of different sensitive attribute groups. However, the learning mechanisms to achieve parity pay a significant cost in terms of the accuracy (or utility) of their predictions.

In our NIPS 2017 paper [323], we introduce, formalize and evaluate new notions of fairness that are inspired by the concepts of **fair division** and **envy-freeness** in economics and game theory. Our work is motivated by the observation that, in certain decision making scenarios, the existing parity-based fairness notions may be too stringent, precluding more accurate decisions, which may also be desired by every sensitive attribute group. To relax these parity-based notions, we introduce the concept of a user group's **preference** for being assigned one set of decision outcomes over another. Given the choice between various sets of decision outcomes, any group of users would collectively prefer the set that contains the largest fraction (or the greatest number) of beneficial decision outcomes for that group. We design margin-based classifiers that satisfy these preference-based notions of fairness and show that they allow for greater decision accuracy than parity-based fairness with a variety of synthetic and real-world datasets.

10.2.2 Beyond Distributive and Normative Fairness

a. Towards Procedurally Fair Learning. Most prior works, including ours, have focused on only one dimension of fair decision making: **distributive fairness**, i.e., the fairness of the *decision outcomes*. In our AAAI 2018 paper [158], we leverage the rich literature on organizational justice and focus on another dimension of fair decision making: **procedural fairness**, i.e., the fairness of the *decision making process*. We propose measures for procedural fairness that consider the input features in the decision process, and evaluate the moral judgments of humans regarding the use of these features. We operationalize these measures on two real world datasets using human surveys on the Amazon Mechanical Turk (AMT) platform, demonstrating that our measures capture important properties of procedurally fair decision making. We provide fast submodular mechanisms to optimize the tradeoff between procedural fairness and prediction accuracy. On our datasets, we observe empirically that procedural fairness may be achieved with little cost to outcome (distributive) fairness, but that some loss of accuracy is unavoidable.

b. Towards Descriptive Ethics. Most prior works on algorithmic fairness **normatively** prescribe *how fair decisions ought to be made*. In contrast, in our WWW 2018 paper [155], we **descriptively** survey users for *how they perceive and reason about fairness in algorithmic decision making*. A key contribution of this work is the framework we propose to understand why people perceive certain features as fair or unfair to be used in algorithms. Our framework identifies eight properties of features, such as *relevance*, *volitionality* and *reliability*, as latent considerations that inform people's moral judgments about the fairness of feature use in decision-making algorithms. We validate our framework through a series of scenario-based surveys with AMT workers as well as a US population representative sample of respondents selected using SSI. We find that, based on a person's assessment of the eight latent properties of a feature in our exemplar scenario, we can accurately (greater than 85%) predict if the person will judge the use of the feature as fair.

Our findings have important implications. At a high-level, we show that people's unfairness concerns are multi-dimensional and argue that future studies need to address unfairness concerns beyond discrimination. At a low-level, we find considerable disagreements in people's fairness judgments. We identify root causes of the disagreements, and note possible pathways to resolve them.

10.2.3 Future Directions

a. From Group Discrimination to Individual Fairness. Existing works on discrimination capture unfairness at the level of population subgroups, but not at the level of individual users. In ongoing work, we are exploring the pontental for using **inequality indices** that have been extensively studied in economics and social welfare literature to quantify algorithmic unfairness at both the individual and group-levels. Our interest in using these indices is rooted in the *well-justified axiomatic basis* for their designs. Specifically, the property of *subgroup decomposability* allows individual unfairness measured over an entire population to be expressed as the sum of a *between-group* unfairness component and a *within-group* unfairness component. Thus, inequality indices not only offer a unifying approach to quantifying unfairness at the levels of both individuals and groups, but they also reveal previously overlooked tradeoffs between fairness notions.

b. Fairness in Search Rankings. Rankings of people and items are at the heart of selection-making, match-making, and recommender systems, ranging from employment sites to sharing economy platforms. In our CSCW 2017 paper [195], we investigated the sources of bias for political searches in social media that lead to preferential ranking of certain political perspectives over others. In ongoing work, we are exploring new measures and mechanisms to quantify and mitigate unfairness from *position bias* which leads to disproportionately less attention being paid to low-ranked subjects. As no single ranking can achieve individual attention fairness, we are investigating ranking mechanisms that achieve **amortized fairness**, where attention accumulated across a series of rankings is proportional to accumulated relevance. Amortizing individual fairness subject to constraints on ranking quality can be formulated as an integer linear program.

c. Fairness in Selecting a Set of Recommendations. Our ICWSM 2017 paper [72] shows that trending topics on social media sites like Twitter are promoted by crowds whose demographics differ significantly from Twitter's overall user population and that certain demographic groups (e.g. black women) are severely *under-represented* in the process. In ongoing work, we are reimagining crowdsourced recommendations like trending topics as the outcomes of a *multi-winner election* that is periodically repeated. Our insight is that the observed biases in trending topic selection algorithm can be attributed to **unfair representation** in the electoral system. We plan to mitigate the biases by leveraging electoral mechanisms from fair voting literature like *Single Transferable Vote (STV)* that ensure fairness representation criteria such as *proportional representation* and *anti-pluarity*.

11 The Rigorous Software Engineering Group

11.1 Overview

The report covers the period from August 2015 – January 2018. The group's research interests are in the foundational principles of software engineering (models of computation, analysis algorithms) and applications of these principles to programmer productivity tools. Major research topics are in the verification and control of reactive, real-time, and hybrid systems, software verification and program analysis, logic, and automata theory.

Personnel. The group is led by **Rupak Majumdar** and currently has five graduate students (Ivan Gavran, Johannes Kloos, Aman Mathur, Filip Niksic, and Mahmoud Salamati), and three postdoctoral researchers (Marko Horvat, Burcu Ozkan, and Anne-Kathrin Schmuck). Additionally, there are two junior research group leaders (Daniel Neider and Damien Zufferey) who run their own groups with an independent budget.[1] The report of these sub-groups are in Sections 11.3 and 11.4, respectively.

During the review period, Zilong Wang graduated with a PhD degree and joined Huawei. Johannes Kloos has submitted (but not defended) his dissertation. Three MS students (David Deininger, Adrian Leva, Aman Mathur) worked on their MS thesis. Post-doctoral researchers who were associated with the group, as well as their subsequent appointments, are: Dmitry Chistikov (assistant professor at Warwick), Rayna Dimitrova (assistant professor at Leicester), Samira Farahani (researcher at TU Delft), Vinayak Prabhu (assistant professor at Colorado State), and Sadegh Soudjani (assistant professor at Newcastle).

Collaborations. Rupak has a joint ERC Synergy award together with Michael Backes, Peter Druschel, and Gerhard Weikum. Internally, the group has had collaborations with Eva Darulova and Viktor Vafeiadis. Externally, the group has had collaborations with Alessandro Abate (Oxford), Dmitry Chistikov (Warwick), Sylvain Conchon (Paris-Sud), Jyo Deshmukh (Toyota), Javier Esparza (TU Munich), Pierre Ganty (IMDEA Madrid), Tandra Ghose (TU Kaiserslautern), Amit Goel (Apple), Sumit Gulwani (Microsoft Research), Holger Hermanns (Saarland University), Aditya Kanade (IISc Bangalore), James Kapinski (Toyota), Sava Krstic (Intel), K. Narayan Kumar (CMI), Jerome Leroux (LABRI Bordeaux), Anthony Lin (Oxford), Frank McCabe (Google), Thomas Moor (Erlangen), Jochen Hoenicke and

[1] Junior research groups are non-tenure track positions in the Max Planck Society.

Andreas Podelski (University of Freiburg), Shaz Qadeer (Microsoft Research), Philipp Rümmer (Uppsala), Indranil Saha (IIT Kanpur), Sadegh Soudjani (Newcastle), Paulo Tabuada (UCLA), Ufuk Topcu (University of Texas at Austin), James Worrell (Oxford), and Majid Zamani (TU Munich). Many of these collaborations are ongoing.

Publications. The publications of the group have broadly been in two areas: design and verification of cyber-physical systems and theory and practice of formal verification.

In cyber-physical systems, the group has published HSCC (3) [287, 223, 170], EMSOFT (2) [108, 145], Acta Informatica (1) [286], Discrete Event Dynamical Systems (2) [273, 274], Formal Methods in Systems Design (2) [124, 110], ACC (1) [135], CDC (4) [114, 275, 225, 205], QEST (1) [289], WODES (1) [224], IEEE Trans. Autom. Control (1) [75].

In formal verification, the group has published JACM (1) [133], POPL (2) [221, 168], CAV (1) [87], CONCUR (4) [285, 130, 146, 115], Acta Informatica (2) [132, 80], FMCAD (1) [100], HVC (1) [187], TACAS (5) [208, 217, 218, 288, 113], ATVA (1) [106], FSTTCS (1) [131], CCS (1) [102], ICALP (2) [84, 81], Fossacs (2) [88, 79], LICS (1) [20], SODA (1) [85], STACS (1) [83], TCS (1) [82], SIAM J. Applied Algebra and Geometry (1) [86].

In addition, Rupak was the co-editor of [47, 219, 220, 55].

Software, tools, data and technology transfer. We Our work on testing of Simulink models [110, 108] is used by Toyota.

Teaching. Rupak Majumdar taught Complexity Theory (2016, 2017-18) at the University of Kaiserslautern. Rupak Majumdar and Daniel Neider co-taught Advanced Automata Theory (2017). Damien Zufferey taught Concurrency Theory (2017-18). Rayna Dimitrova taught Program Analysis (2016). We also supervised several MS and BS theses.

External funding. Our research is supported in part by an ERC Synergy Award "ImPACT: Privacy, Accountability, Compliance, and Trust in Tomorrow's Internet," with co-PIs Michael Backes, Peter Druschel, and Gerhard Weikum. The project started in February 2015 and is funded for six years. Additionally, our research is supported in part by industrial grants from Toyota ($75K annually for 2015, 2016, 2017) and InStart Logic (2015).

Invited talks. Rupak gave invited talks at ETAPS 2016, SETTA 2017, SYNT 2017, ICFEM 2016.

Major Service. Rupak served on the program committees of several conferences in the last two years, and chaired RV 2015, POPL 2016, and CAV 2017. He organized the Dagstuhl seminars "Formal synthesis of cyber-physical systems," (jointly with Calin Belta, Majid Zamani, and Matthias Rungger) and "Game theory in AI, Logic, and Algorithms" (jointly with Swarat Chaudhuri, Sampath Kannan, and Michael Wooldridge). Rupak serves on the POPL Steering Committee. Rupak is an Associate Editor of TOPLAS and was previously an Associate Editor of TECS.

11.2 Research Agenda

The Rigorous Software Engineering group studies both foundational principles and practical tools for the design and analysis of computer systems. Currently, the research in the group has focused on three different aspects: methodologies and tools for embedded controller design and foundations of infinite-state verification. Our work is often accompanied with tools for software productivity, such as testing and verification tools. The following are some highlights from the last reporting period.

11.2.1 Design and Verification Methodologies for CPS

Abstraction-based Control Design One main focus of our research has been the development of design and verification techniques for autonomous cyber-physical systems. We have considered the problem at various levels: from low level control design for dynamical systems, to efficient testing and verification of hybrid systems models, to programming methodologies for large groups of autonomous robots.

In control design, we have focused on *abstraction-based control design* (ABCD). In this approach, a continuous dynamical system is abstracted to a finite-state system such that a controller designed for the finite-state system can be refined to a controller for the original system while maintaining certain guarantees on the closed-loop dynamics. When the specification is given as ω-regular languages, one can use reactive synthesis algorithms to find a controller for the finite-state abstraction and then refine this controller to ensure that the continuous system satisfies an ω-regular specification.

A key problem in abstraction-based control design is scalability: the abstraction grows exponentially with the dimension of the system. In recent

work, we have focused on different approaches to combat the scalability problem. We have considered a "lazy" version of the abstraction, where multiple, increasingly more abstract, models of the system are constructed, and the synthesis algorithm computes a set of controllers with disjoint domains which together ensure the specification. In this way, the controllers work at the most abstract version of the state space required to ensure the property holds globally.

We have also designed a modeling formalism and a synthesis algorithm that exploits the natural hierarchical structure present in many tasks for more scalable synthesis. We defined *local games* on hierarchical graphs as a modeling formalism that decomposes a large-scale reactive synthesis problem in two dimensions. First, the construction of a hierarchical game graph introduces abstraction layers, where each layer is again a two-player game graph. Second, every such layer is decomposed into multiple local game graphs, each corresponding to a node in the higher level game graph. While local games have the potential to reduce the state space for controller synthesis, they lead to more complex synthesis problems where strategies computed for one local game can impose additional requirements on lower-level local games. We showed how to construct a dynamic controller for local game graphs over hierarchies. The controller computes assume-admissible winning strategies that satisfy local specifications in the presence of environment assumptions, and dynamically updates specifications and strategies due to interactions between games at different abstraction layers at each step of the play. We show that our synthesis procedure is sound: the controller constructs a play that satisfies all local specifications.

Finally, we extended work in abstraction-based control to various continuous-state probablistic models, such as discrete-time stochastic dynamical systems and continuous-time jump Markov processes.

Testing Simulink Models Metrics on hybrid systems quantify the notion of similarity between behaviors and generalize notions of bisimilarity and trace equivalence from discrete systems to hybrid systems. While metrics on hybrid state spaces have been used to give semantics to hybrid systems, so far, the algorithmic computation of metrics as well as the use of metrics in conformance testing had not been studied. We have developed algorithms to compute metrics on timed and hybrid systems.

In [222, 109], we considered the Skorokhod distance on hybrid traces. The Skorokhod distance computes a metric on traces that takes into account timing distortions in addition to differences in the continuous state.

While it has been used before to give semantics to hybrid and probabilistic systems, and to define continuous bisimulation functions, algorithms to compute the distance were not known. In [222], we describe a polynomial-time algorithm to compute the Skorokhod distance between time-sampled traces completed by linear interpolation (called "polyhedral traces"). Computing the Skorokhod distance is non-trivial because the definition of the distance minimizes over an infinite family of continuous retiming functions. Our polynomial-time algorithm uses geometric characterizations of the space of solutions that were discovered in the study of Fréchet distances in computational geometry. In [109], we implemented our algorithm in a tool for conformance testing of Simulink models. In collaboration with Jyo Deshmukh at Toyota, we evaluated our algorithm on a set of industrial control system benchmarks. Our implementation shows that the distance can be computed fast and captures the engineering intuition about "close" behaviors. We also characterize the distance using a timed linear time logic with signals and freeze quantifiers.

Antlab: A Multi-Robot Task Server At the higher level of co-ordinated controller design, we have worked on systems of multiple autonomous robots serving tasks provided declaratively. We have designed and implemented Antlab, an end-to-end system that takes streams of user task requests and executes them using collections of robots. In Antlab, each request is specified declaratively in linear temporal logic extended with quantifiers over robots. The user does not program robots individually, nor know how many robots are available at any time or the precise state of the robots. The Antlab runtime system manages the set of robots, schedules robots to perform tasks, automatically synthesizes robot motion plans from the task specification, and manages the co-ordinated execution of the plan.

We provide a constraint-based formulation for simultaneous task assignment and plan generation for multiple robots working together to satisfy a task specification. In order to scalably handle multiple concurrent tasks, we take a separation of concerns view to plan generation. First, we solve each planning problem in isolation, with an "ideal world" hypothesis that says there are no unspecified dynamic obstacles or adversarial environment actions. Second, to deal with imprecisions of the real world, we implement the plans in receding horizon fashion on top of a standard robot navigation stack. The motion planner dynamically detects environment actions or dynamic obstacles from the environment or from other robots and locally corrects the ideal planned path. It triggers a re-planning step dynamically if

the current path deviates from the planned path or if planner assumptions are violated.

We have implemented Antlab as a C++ and Python library on top of robots running on ROS, using SMT-based and AI planning-based implementations for task and path planning. We evaluated Antlab both in simulation as well as on a set of TurtleBot robots. We demonstrate that it can provide a scalable and robust

Currently, the control problems in Antlab's robots are simple and the focus is on reactive planning and co-ordination. However, a future goal is to combine low-level abstraction-based controllers with Antlab.

11.2.2 Infinite-State Verification

Our second focus is in algorithmic foundations and practical tools for infinite state verification. Some highlights in the last two years are: (1) decidability results and deductive proof rules for parameterized systems; (2) combinatorial constructions to provide theoretical guarantees for random testing.

Parameterized Verification Parameterized verification asks whether all systems in an unbounded family of systems satisfy a given safety or liveness specification. In our research on parameterized verification, we have considered the following problems: decidability of non-atomic networks for safety and liveness, algorithmic analysis of population protocols, and deductive verification of parameterized concurrent systems.

We have characterized the complexity of verification parameterized systems with a designated "leader" process and an arbitrary number of "follower" processes that communicate with a shared, finite-valued register that does not have an atomic test-and-set operation. We show that safety and liveness verification in this setting are both NP-complete when the leader and followers are implemented by finite-state machines [133, 124]. We show that the verification problems get harder (PSPACE for safety, EXPTIME for liveness) when processes are allowed to have unbounded stacks.

Second, we studied *population protocols*, a model of distributed computation introduced by Angluin about 12 years ago. We show natural verification questions for population protocols are decidable (but usually as hard as Petri net reachability) using tools from modern Petri net theory [132, 131, 130].

Finally, we studied extensions to *thread-modular proof rules* for reasoning about concurrent programs [168]. We show a strict hierarchy of proof rules (in terms of expressive power) building up on thread modularity and connect the proof rules to other notions such as Cartesian abstractions. We

also considered a language-theoretic approach to deductive verification of parameterized probabilistic systems. In our approach [208], we search for invariants and ranking functions representable as automatic structures and use algorithms on finite-state automata.

Theoretical Guarantees for Random Testing We have considered the problem of providing formal guarantees on the performance of random testing. Empirically, many bugs in concurrent programs have been observed to have small "bug depth." We show how the informal notion of bug depth can be formalized and, in two situations in concurrent systems testing, how one can provide a theoretical guarantee. First, in testing distributed systems against network partition errors, we demonstrate that many bugs can be found by splitting the nodes in the network in some way. Using the probabilistic method from combinatorics, we show that a popular random testing approach can quickly find a covering family: a set of tests which grows logarithmically with the size of the network but already covers all possible splits [221]. Second, in testing concurrent programs whose executions form a partial order, we define the notion of hitting families of schedules as a set of linearizations of the partial order that are guaranteed to cover all possible orderings of any d elements. Using a mix of probabilistic arguments and explicit combinatorial constructions, we show how to construct hitting families [87].

11.2.3 Exploring programming interfaces

A recent research direction in the group has focused on harnessing novel interfaces for programming. The availability of mass-market immersive head-mounted displays for virtual and augmented reality, together with 3D printers and rapid prototyping systems promises a major change in how humans interact with computers. A fundamental research question is how one can develop new software design and analysis environments that exploit these devices to provide novel interfaces to programmers. Recent projects in this direction include an immersive debugging and software visualization tool for concurrent code, preliminary research in programming smart CAD objects by "direct manipulation" within a virtual environment, as well as natural language interfaces for reactive planning.

A fundamental research question is to understand human visual performance in virtual environments. It turns out that many basic questions in visual perception in artificial 3D environments is unknown, partly because one could not design precise psychophysical experiments in immersive envi-

ronments without incurring major infrastructure cost. Over the last year, we have collaborated with the group of Tandra Ghose, a perceptual psychologist at TU Kaiserslautern, to design perception experiments for virtual reality environments. Initial experiments have focused on performance of human visual search and models for human visual search in 3D. We expect further collaborations in understanding cognitive decision making in 3D virtual environments.

11.3 Formal Methods and Machine Learning

The report covers the period from February 2017, when this group was established, to January 2018. The group's research focus lies in the intersection of formal methods and machine learning with the aim to exploit synergies between both disciplines. During the reporting period, the group has focused mainly on applications of machine learning in verification and synthesis.

Personnel

The group is led by **Daniel Neider**. Daniel is currently supervising a Bachelor project on learning-based reactive synthesis via learning of decision trees.

Collaborations

Internally, the group collaborates with Rupak Majumdar's group as well as Damien Zufferey's group. Externally, the group collaborates with Sergiy Bogomolov (The Australian National University), Deepak D'Souza (Indian Institute of Science, Bangalore), P. Madhusudan (University of Illinois at Urbana-Champaign), Karim Ali (University of Alberta), Paulo Tabuada (University of California, Los Angeles), and Martin Zimmermann (Saarland University).

Publications

During the reporting period, the group has co-authored a TACAS conference paper titled "Invariant Synthesis for Incomplete Verification Engines" [243], which will appear in April 2018. This paper introduces a novel, learning-based framework for synthesizing invariants for programs with specifications in undecidable logics that permit sound-but-incomplete decision procedures. The group has also co-authored an article in Transactions on Computational

Logic titled "Compositional Synthesis of Piece-wise Functions by Learning Classifiers" [244], which proposes classifier learning to synthesize piece-wise functions (i.e., functions that split the domain into regions and apply simpler functions to each region) from logical synthesis specifications.

Two further conference articles are currently under submission. The first article, titled "Synthesizing Optimally Resilient Controllers" [245], deals with synthesis of reactive controllers that are optimally resilient against (unmodeled) disturbances. The second article, titled "Horn-ICE Learning for Synthesizing Invariants and Contracts" [121], presents a learning-based technique for inferring invariants and contracts of programs whose verification conditions are given in form of constrained Horn clauses.

Teaching

During summer term 2017, Daniel has co-taught the graduate course "Advanced Automata Theory" at the University of Kaiserslautern together with Rupak Majumdar.

Service

Daniel has served as program committee member at VMCAI (2018).

Research Agenda

The group's research focus lies in the intersection of formal methods and machine learning. On the one hand, the group develops novel, learning-based methods for the design, construction, and verification of hard- and software. On the other hand, the group explores applications of formal methods in the field of machine learning and aims to devise novel learning algorithms for data with complex dependencies (e.g., expressed as logic formulas). This research is motivated by the observation that combining formal methods and machine learning offers many promising synergies, which have not yet been explored adequately.

Machine Learning in Formal Methods The group explores the use of machine learning techniques in various areas of formal methods, especially in verification and synthesis.

In the area of verification, the group focuses on automated invariant synthesis for a variety of verification settings (such as verification of numeric and heap-manipulating programs, verification of parametric systems, and

regular model checking). Most of these approaches, though not all, build on top of an invariant learning framework called "ICE learning", which Daniel has developed during his doctoral studies. This framework extends the classical machine learning setup with implications (i.e., dependencies of the form $d_1 \Rightarrow d_2$, imposing that the classification of d_1 influences the classification of d_2 but not the other way round), which capture the semantics of the system under analysis. Implications, however, are uncommon in machine learning and, thus, require engineering fundamentally new learning algorithms.

In the area of synthesis, the group mainly focuses on learning-based synthesis of reactive controllers from logical specifications. The group's research so far has focused on "weak" specifications, such as safety and reachability, but will address the problem of learning reactive controllers for the whole class of ω-regular specifications in the future.

Formal Methods in Machine Learning Formal methods have attracted increasing interest for verifying machine learning models. The results, however, are often disappointing as formal methods do not scale to today's large-scale models (e.g., deep neural nets). Therefore, this group considers problems of more manageable size and studies formal methods in the context of an artificial neural net with slightly less than 400 neurons obtained from C. Elegans, a nematode whose nervous system has completely been mapped. The overall goal is to produce sequences of stimuli that make (a simulation of) C. Elegans follow predefined behavior.

Plans for Future Work

In the upcoming years, the group's research will further exploit the synergies between formal methods and machine learning. Here, I briefly discuss three directions of future research.

- *Verification of dynamic and hybrid systems* Based on the ICE learning framework, the group will develop learning-based verification methods for dynamic and hybrid systems. The key challenge here to lift key components of the ICE learning framework, in particular the notion of implications, to continuous domains. The long-term goal is the development of automated, learning-based verification tools for cyber-physical systems.

- *Learning termination proofs* Termination of programs is typically proven by providing appropriate ranking functions. Starting with recent work on compositional synthesis of piece-wise functions [244], the group will

develop learning-based techniques that synthesize ranking functions in an automated manner. In a subsequent step, the group will build methods that learn invariants (i.e., correctness proofs) and ranking functions (i.e., termination proofs) simultaneously.

- *Learning algorithms for data with complex dependencies* When machine learning is applied in a formal context, learning algorithms are often confronted with a combination of data and logical constraints (e.g., in form of logic formulas); one such example is implications as introduced in the ICE learning framework. To improve the availability of machine techniques for verification and synthesis tasks further, the group will develop extensions of classical learning algorithms that can handle both data and logical constraints. The extension of ICE learning with Horn constraints [121] is a first successful step into this direction.

11.4 Program Interactions with the External World

11.4.1 Overview

The report covers the period from December 2016 – January 2018. The research of this group focuses on the programming models and verification techniques for software systems where the software function is not only logical but also depends on the external world. The major research areas we have focused on in the current review period are message-passing concurrency and coordination in cyber-physical systems, programming abstraction for faults in distributed systems, and analyses for the Android framework.

Personnel. The subgroup is led by **Damien Zufferey** and currently has one graduate student (Marcus Pirron), one visiting student (Yunjun Bai from the Chinese Academy of Science). Marcus Pirron started his PhD in June 2017. Yunjun Bai is supported by a grant from the Chinese government. She came in September 2017 and will stay for 18 months. Furthermore, Damien supervised the Master's thesis of Richard Peifer, a master student at Saarland University, in the summer of 2017.

Collaborations. Ongoing external collaborations include Dr. Cezara Dragoi at INRIA (France) and Dr. Josef Widder at TU Wien (Austria), with whom we are looking at verification of distributed systems. With the groups of Prof. Thomas Wies at NYU (USA) and Prof. Ruzica Piskac at Yale University (USA), we are working on extension of GRASShopper, a verifier for

heap manipulating programs. With the groups of Prof. Pavol Cerný and Prof. Bor-Yuh Evan Chang at University of Colorado Boulder and Dr. Arjun Radhakrishna at Microsoft we developed a tool for interface extraction in the Android framework. With Prof. Sicun Gao at UCSD and Dr. Soonho Kong at Toyota research institute we are working on techniques for constraint solving for non-linear equations. We also had early discussions with Dr. Eva Darulova internally and Dr. Philipp Stanley Marbell at University of Cambridge about a potential collaboration related to energy and computation in embedded systems.

Publications. During the reporting period, group members have coauthored a paper to appear in ICSE 18 [266]. In this work we look at the problem of learning the asynchronous communication protocol for Android classes. Android is an event-driven framework and the communication between an application and the framework occurs through callins band callbacks. We develop a testing-based method to learn event-driven interfaces as finite-state machines.

Software, tools, and data. We are continuing development on existing tools as part of our current research. The most notable one are GRASShopper tool [262] for verification of heap manipulating programs, the PSync [119] framework for distributed algorithms, the dReal [144] SMT solver for non-linear arithmetic over the real, and the DroidStar [266] tool inference of typestates in the Android framework. Furthermore, we started building the infrastructure for our ongoing research project related to cyber-physical systems. Once mature enough, we will make these tool available.

Teaching. Damien taught the concurrency theory course at TU-KL during the winter semester 2017. This is a master level course and it planned to happen again in the winter semester 2018.

Service. Damien co-organized the Verification Mentoring Workshop (VMW) 2017 which was collocated with CAV 2018. He was a PC member for CAV 2017, MEMOCODE 2017, SYNT 2017, SMT 2017, TAPAS 2017, TACAS 2018 and CAV 2018, was an ERC member for POPL 17, reviewed for CONCUR 2017, and has accepted to serve the Program Committees of SAS 2018, PLDI Student Research Competition 2018, and the Scala Symposium 2018.

Internally, Damien has organized a series of tutorials about the new hardware laboratories set up in both locations. The tutorials were designed to introduce students and researchers to rapid prototyping tools (3D printing, CNC milling), design and fabrication of printed circuit boards, programming of micro-controllers.

11.4.2 Research agenda

The main research projects in the group focus on the boundaries of programs and their interaction with the world. First, there is work on programming abstraction for fault-tolerant systems and, second, work on cyber-physical systems.

Programming abstraction for fault-tolerant systems. Most programming models are designed with the assumption that the system runs correctly. However, hardware failure are expected, especially in large scale distributed applications. Unfortunately, current programming abstraction gives little support to handle these cases and fault-tolerant systems are notoriously hard to implement. The programmer needs to reason about about faults on top of the existing challenges such as asynchrony.

The PSync [119] domain specific language uses communication-closed rounds as a core abstraction and provides an idealized lockstep semantics which is indistinguishable from the executions over an asynchronous network. This abstraction applies to a wide variety of settings. However, the current implementation focuses on one of them: partially synchronous networks with benign crash-stop faults. We are working on extending PSync to support more types of faults, in particular, Byzantine faults. Byzantine faults can model an attacker who actively tries to compromise the system. Therefore, extending the PSync runtime to Byzantine models will not only help build reliable systems but also secure ones as the system can be made tolerant to attackers.

Communication-closed rounds also help automated verification of fault-tolerant algorithms. For instance, the lockstep model removes the combinatorial blow-up of an interleaving semantics. Dealing with faults tolerant algorithms requires a very expressive logic that mixes set comprehensions, cardinality constraints, and a restricted form of quantification. For example, in a consensus algorithm, the processes try to agree on a common decision value. A strategy to achieve consensus is establishing a majority of processes sharing the same value. This is described by the formula $\exists v.\ \forall p.\ \mathit{decision}(p) = v \ \Rightarrow\ |\{q \,|\, x(q) = v\}| > n/2$ where the parameter

n is the number of processes participating in the consensus, and x a local variable of each process. Our current work has focused on consensus algorithms which have the particularity, among others, of producing an unique decision. We take advantage of such properties when dealing with universal quantifiers. Extending the scope the verifier requires extending the current semi-decision procedure to better deal with quantifiers and functions symbols.

Programming models for robotic and hybrid systems. The availability of affordable rapid prototyping tools like 3D printers, laser cutters, and CNCs, along with cheap system-on-chips, micro-controllers, and sensors, has greatly lowered the barriers to entry for people to design, build, and program robots. However, programming such robots is still a very complex task. On top of the normal software engineering challenges, one also has to consider the interaction with the physical world. To address these challenges, programming languages should go beyond focusing only on the software aspect, and integrate models of the physical world.

The first part of this project, we are looking at combining together components which are themselves self-contained units with their own sensors, actuators, and dynamic controllers. When creating a new assembly from such components, their dynamic gets coupled and they need to coordinate to perform actions. The model we are developing for this level of abstraction requires loosely coupled dynamic so that the controllers can still be used a independent black-boxes. On top of the controller sits message-passing concurrent program that orchestrate the components to make sure the requirements of each controller is met. The coordination is also important to sequence actions and properly sharing the common resource of geometric space, i.e., avoiding collisions. Our approach tries, on one hand, to bring theoretical advances from the programming language community, such as resource logics for sharing the space and session types to structure the communication and, on the other hands, use controller synthesis techniques to deal with the physical coupling.

On the other "end" of this project, we are looking at the procedural generation of 3D models. The advances in rapid prototyping and manufacturing tools enables the production of custom parts more easily and economically. This level of flexibility requires parameterized and reusable designs. Programs are a good candidates as representation for parametric designs. A program describes a family of objects and an execution with specific parameters produces an object. The success of domain specific languages such as

OpenSCAD confirms this intuition. The combination of traditional control-structure (branches, loops, procedures) and dedicated geometric primitives (cube, sphere, intersection, convex hull) offers an unprecedented degree of flexibility. However, the major downside of this approach is that it a one-way process. It is possible to give parameter values to get an object but it is currently not possible to change an object and get the parameter values that correspond to the new object. Using techniques from software synthesis, we aim to change that and develop techniques that enable the direct manipulation of the object. Then an algorithm will automatically find the appropriate parameters and potential changes to the program's structure.

Finally, to connect both parts we are looking at controller synthesis from structural descriptions. Enriching the low-level geometric description with semantics information like joints, actuators, sensors, and a specification, our goal is to directly generate the dynamic controllers orchestrated by our higher-level message-passing layer. This problem has been tackled for serial structures but is still largely open for parallel/coupled structures. Here, we are looking at harnessing the progress in automated reasoning tools and emerging support for non-linear equation in SMT solvers along with techniques from numerical optimizations.

12 The Foundations of Algorithmic Verification Group

12.1 Overview

Joël Ouaknine joined MPI-SWS as Scientific Director in April 2016 (initially part-time, then full-time from 1 August 2016), and accordingly the report covers the period from April 2016 – January 2018. His group's research focuses on a range of fundamental algorithmic problems from verification, synthesis, performance, and control for linear dynamical systems (both discrete and continuous), drawing among others on tools from number theory, Diophantine geometry, and algebraic geometry, with the overarching goal of offering a systematic *exact computational treatment* of various important classes of dynamical systems and other fundamental models used in mathematics, computer science, and the quantitative sciences.

Personnel. The group is led by **Joël Ouaknine** and currently has three postdocs in Saarbrücken (Johar Ashfaque, Ventsi Chonev, and Amaury Pouly). James Worrell, professor of Computer Science at Oxford University, has been a frequent visitor and ongoing collaborator. Joël Ouaknine is also currently co-supervising one PhD student (Mehran Hosseini) as well as a postdoc (Shaull Almagor), both based in Oxford and funded by Ouaknine's ERC grant. One doctoral student (João Sousa Pinto, based in Oxford) graduated during the reporting period.

Collaborations. During the review period, the group has engaged in successful collaborations with a number of leading researchers in Europe, including James Worrell and Ehud Hrushovski (Oxford University), Nathanaël Fijalkow (Alan Turing Institute), Patricia Bouyer (ENS Paris-Saclay), and Nicolas Markey (Université de Rennes).

Publications. During the review period, group members have published papers in the journals J. ACM [90, 60], Information and Computation [67, 59], and ACM Transactions on Computational Logic [206]. In addition, group members have published at leading peer-reviewed conferences including LICS [89, 253], ICALP [91, 66, 58, 15, 61], STACS [141], HSCC [252], CONCUR [207], and CMSB [134].

External funding. The research of the group has been partially funded by ERC Consolidator Grant AVS-ISS (648701), August 2015–August 2020.

Invited talks and awards. Joël Ouaknine was invited speaker at FoSSaCS 2017 (the 20th Int'l Conf. on Foundations of Software Science and Computation Structures, part of the European Joint Conferences on Theory and Practice of Software), as well as the special workshop organised to celebrate the 10th anniversary of the University of Warwick's Centre for Discrete Mathematics and its Applications (DIMAP), in December 2017.

Amaury Pouly received the prestigious Ackermann Award in 2017.

In addition, several papers published by group members received accolades during the review period: *"On the complexity of the Orbit Problem"*, by Chonev, Ouaknine, and Worrell (J. ACM 2016) [90] was listed as a Notable Article by ACM Computing Reviews 21st Annual Best of Computing for 2016.[2] The authors of the paper *"Semialgebraic Invariant Synthesis for the Kannan-Lipton Orbit Problem"*, by Fijalkow, Ohlmann, Ouaknine, Pouly, and Worrell (STACS 2017) [141] were invited to submit an extended version of their work to a special issue of the journal Theory of Computing Systems, now under review. Finally, the papers *"Polynomial Time Corresponds to Solutions of Polynomial Ordinary Differential Equations of Polynomial Length: The General Purpose Analog Computer and Computable Analysis Are Two Efficiently Equivalent Models of Computations"* (ICALP 2016) [58] and *"Strong Turing Completeness of Continuous Chemical Reaction Networks and Compilation of Mixed Analog-Digital Programs"* (CMSB 2017), both co-authored by Pouly, each won Best Paper Award at the conference at which they appeared.

Service. Joël Ouaknine served on the PC of FoSSaCS 2018 and as PC Chair of LICS 2017. Since 2017, he also serves on the Steering Committee of LICS. He is Associate Editor for the Journal of Computer and System Sciences, Elsevier. He organised a successful international workshop on Algorithmic Aspects of Dynamical Systems at the Bellairs Research Centre in March 2017.

He is presently chairing the Faculty Recruiting Committee of MPI-SWS for the 2017-2018 season.

12.2 Research agenda

The Foundations of Algorithmic Verification Group focuses on theoretical problems arising out of automated-verification research, broadly construed, with a particular emphasis on algorithmic questions.

[2] `http://www.computingreviews.com/recommend/bestof/notableitems.cfm?bestYear=2016`

During the reporting period, the group's efforts have revolved around the analysis of discrete and continuous linear dynamical systems. Such systems are widely used as abstractions of components of computer programs and embedded systems, including cyber-physical systems. The problems considered include reachability, invariant synthesis, performance, and controllability questions; somewhat surprisingly, several of these questions turn out to have intimate connections to deep problems in mathematics, particularly in number theory, Diophantine geometry, and algebraic geometry.

Consider, for example, the following *simple linear loop*:

```
while (2x-3y>0) do { x:=3x-y ; y:=x-5y }
```

Here the loop guard is a linear inequality, and the loop body is a sequence of linear updates. Remarkably, it is not known how to decide whether a given simple linear loop terminates on a given initial value, even if we restrict to loops with at most 6 variables.[3] The mathematician Terrence Tao has remarked on the subject that "it is faintly outrageous that this problem is still open; it is saying that we do not know how to decide the halting problem even for *linear* automata" [298, Sec. 3.9].

There are similar, seemingly simple, open problems in the theory of continuous linear dynamical systems. For example, consider the solution $\boldsymbol{x}(t) \in \mathbb{R}^d$ of a linear differential equation

$$\frac{d\boldsymbol{x}}{dt} = A\boldsymbol{x}\,,$$

where A is a $d \times d$ matrix of rational numbers, for a given rational initial condition $\boldsymbol{x}(0)$. Given a rational hyperplane $\mathcal{H}$, the continuous Skolem-Pisot Problem asks whether the trajectory $\boldsymbol{x}(t)$ ever reaches $\mathcal{H}$ at some time t. Again, this problem is not known to be decidable [48].

In the face of such limits, researchers in automated verification have focused on incomplete methods and on models with restricted linear dynamics. For example, the main approach to proving termination of linear loops involves synthesising linear and lexicographic-linear ranking functions. This method underlies Microsoft Research's TERMINATOR tool [101]. However there are terminating linear loops that have no linear ranking function, so this approach does not yield a general decision procedure. Likewise, theoretical work on verifying hybrid systems has focused on models with very simple continuous dynamics, such as timed automata, rectangular hybrid

[3]Note that unlike the infamous Collatz Problem, here there is no conditional within the body of the loop.

automata, and o-minimal hybrid automata. Such frameworks can be very restrictive, e.g., to guarantee o-minimality of the solution of a differential equation $\frac{d\boldsymbol{x}}{dt} = A\boldsymbol{x}$, one requires strong assumptions on the spectrum of the matrix A [196].

Our research vision is to attack foundational problems in the verification of software and cyber-physical systems in terms of decision problems on linear dynamical systems and extensions thereof, such as affine programs and linear hybrid automata. Our high-level goal is to comprehensively map the algorithmic landscape of verification problems for both discrete and continuous linear dynamical systems, and attendant extensions.

In what follows, we describe select recent achievements and ongoing research work.

Some recent results and research directions. The *Orbit Problem* (closely related to the termination of linear loops) was introduced by Harrison in 1969 as a formulation of the reachability problem for linear sequential machines. It is stated as follows:

> Given a square matrix $A \in \mathbb{Q}^{d\times d}$ and vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{Q}^d$, decide whether there exists a non-negative integer n such that $A^n\boldsymbol{x} = \boldsymbol{y}$.

The decidability of this problem remained open for over ten years, until it was shown to be decidable in polynomial time by Kannan and Lipton [182]. In the conclusion of the journal version of their work [183], the authors discussed a higher-dimensional extension of the Orbit Problem, as follows:

> Given a square matrix $A \in \mathbb{Q}^{d\times d}$, a vector $\boldsymbol{x} \in \mathbb{Q}^d$, and a subspace $V \subseteq \mathbb{Q}^d$, decide whether there exists a non-negative integer n such that $A^n\boldsymbol{x} \in V$.

Kannan and Lipton speculated that for target spaces V of dimension one the Orbit Problem might be solvable, "hopefully with a polynomial-time bound". They moreover observed that the cases in which the target space V has dimension two or three seem "harder", and proposed this line of research as an approach towards the Skolem Problem, a famous related question that has been open for many decades [298]. In spite of this, to the best of our knowledge, no progress was recorded on the higher-dimensional Orbit Problem in the intervening two-and-a-half decades, until we recently showed, making use of sophisticated tools from analytic number theory and Diophantine geometry, that one has decidability in PTIME when the target space has dimension one, and in $\mathrm{NP}^{\mathrm{RP}}$ when the target space has dimension

two or three [90]. (In dimension four, the problem is equivalent to an known and longstanding open case of the Skolem Problem.)

We have since substantially widened the scope of discrete-time reachability questions that we examine, for instance by considering polyhedral sources and targets, as well as various controllability problems. Paper [15], published during the reporting period, falls squarely within this line of work, and several further results and open questions are presently in the pipeline.

On the continuous side, the Skolem-Pisot Problem (hyperplane reachability, or equivalently the Zero Problem for single-variable linear ordinary differential equations with constant coefficients) is only known to be decidable in dimension at most two (or order two in the ODE formulation) [48]. Decidability remains open even if one seeks a zero in a *bounded* interval. In recent work, we showed decidability (in arbitrary dimension) of the latter problem subject to Schanuel's Conjecture, a unifying conjecture in transcendental number theory [91]. Moreover, we provide unconditional effective reductions of the unbounded problem to the bounded problem in dimension at most 7, and complete the picture by showing that decidability of the unbounded problem in dimension 9 (or higher) would entail major breakthroughs in the field of Diophantine approximation [89].

Related results in this line of work include the decidability of the *structural liveness* problem for linear hybrid automata, as an immediate corollary of the decidability of the Polytope Escape Problem for continuous linear dynamical problem [252], as well as controllability of certain kinds of switched linear systems [253].

A major thrust of our ongoing research agenda is in the area of *automated invariant synthesis*. Invariants are one of the most fundamental and useful notions in the quantitative sciences, and within computer science play a central rôle in areas such as program analysis and verification, abstract interpretation, static analysis, and theorem proving. To this day, automated invariant synthesis remains a topic of active research; see, e.g., [139], and particularly Sec. 8 therein. In program analysis, invariants play a central role in methods and tools seeking to establish correctness properties of computer programs, including—but not limited to—termination analysis.

Last year, we obtained important preliminary results on the synthesis of semialgebraic invariants for simple linear loops [141], and have since substantially improved our analysis techniques, with several results currently under preparation. We also harbour a strong interest in the automated synthesis of *polynomial* invariants for affine programs. (Affine programs are a simple kind of nondeterministic imperative programs, which may contain arbitrarily nested loops, and in which the only instructions are assignments whose right-

hand sides are affine expressions, such as $x_3 := x_1 - 3x_2 + 7$. Conventional imperative programs can be abstracted to affine programs by replacing conditionals with nondeterminism and conservatively over-approximating non-affine assignments; affine programs thus enable one to reason about more complex programs.)

A polynomial invariant for an affine program with n variables assigns to each program location an algebraic subset[4] of $\mathbb{R}^n$ such that the resulting family of subsets is preserved under the transition relation of the program. Such an invariant is specified by giving a finite set of polynomial equalities at each location. The problem of computing polynomial invariants for affine programs and related formalisms has been extensively studied over the past 40-odd years, ever since Michael Karr gave an algorithm to compute *affine* (or linear) invariants for affine programs in a seminal 1976 paper [185]. As of today, however, no method is known to compute the strongest polynomial invariant, i.e., (a basis for) the set of all polynomial relations holding at each location of a given affine program. Existing methods are either heuristic in nature, or only known to be complete relative to restricted classes of invariants or programs. We are presently attacking this challenging longstanding open problem using techniques from algebraic geometry, group theory, and algebraic number theory.

[4]An algebraic set (or variety) is the set of common zeros of a finite collection of polynomials.

13 The Machine Teaching Group

13.1 Overview

The report covers the period from October 2017 – January 2018. The group's research interests are in the algorithmic foundations of "machine teaching", and applying these algorithms in the applications domains of computational education and trustworthy AI. Formally, machine teaching is an inverse problem of machine learning: it involves a teacher with a desired goal, and the teacher's objective is to find an optimal training sequence to steer a student/learner towards this goal. For instance, in an educational setting, the teacher (e.g., a tutoring system) has an educational goal that she wants to communicate to a student via a set of demonstrations; in adversarial attacks known as training-set poisoning, the teacher (e.g., a hacking algorithm) manipulates the behavior of a machine learning system by maliciously modifying the training data. Machine teaching is an emerging sub-field of AI, and the group plans to advance the research on machine teaching via (i) developing new models, algorithms, and theory of machine teaching; (ii) applying these algorithms to real-world applications by building and deploying new services; and (iii) shaping the field by organizing workshops and giving tutorials to highlight important directions of research[5][327].

Personnel. The group is led by **Adish Singla**. Over the reporting period, the group is hosting two students: Anette Hunziker (University of Zurich) for a master thesis project, and Arpit Merchant (IIIT Hyderabad) for a research fellowship.

Collaborations. Internally at MPI-SWS, the group has collaborated with the social computing group (led by Krishna Gummadi), and the networks and machine learning group (led by Manuel Gomez Rodriguez). Externally, the group has started new collaborations in the area of machine teaching with researchers at Saarland University (Verena Wolf), EPFL (Volkan Cevher), ETH Zurich (Joachim Buchmann, Andreas Krause), Caltech (Yisong Yue), Georgia Institute of Technology (Le Song), and University of Wisconsin (Xiaojin Zhu).

Publications. During the reporting period of four months, group members have published a paper to provide an overview of machine teaching,

[5] `http://teaching-machines.cc/nips2017/`

primarily with a goal of shaping the field and highlighting important directions of future research [327]. In collaboration with the social computing group, and the networks and machine learning group, group members have published two workshop papers [291, 302]. Also, based on prior work done by group members before joining, the group would be presenting three papers at AAAI'18 conference [265, 167, 284].

Service. Internally, Adish served on the qualifying exam committee for Nina Grgic-Hlaca, a graduate student in the social computing group. Externally, Adish has served as a Program Committee (PC) member of NIPS conference (2017). Adish co-organized a workshop at NIPS conference (2017) on "Teaching Machines, Robots, and Humans", and also gave an opening tutorial at the workshop.

13.2 Research agenda

The group's research focus is on developing the algorithmic foundations of machine teaching by grounding our work in two important application domains of computational education and trustworthy AI.

Teacher as a "friend" or a "foe". These two application domains highlight the contrasting role played by the teacher as a "friend" or a "foe", and is further explained below.

- *teacher as a "friend":* In educational applications, we have a human learner as a student, and the focus is on developing algorithms for a teacher in the form of personalized tutoring systems and training simulators.

- *teacher as a "foe":* When studying adversarial attacks, we have a machine learning system as a student (a victim of attacks), and the teacher is an attacker who wants to manipulate the behavior of the student by maliciously modifying the training data. Here, the focus is on understanding the behavior of the attacker which in turn enables us to design optimal defense strategies against future attacks.

Before describing some of the research problems we are working in the group, it is important to discuss the key distinctive ideas that differentiate the concept of "machine teaching" from the standard "machine learning" as further explained below.

Key ideas of machine teaching framework. Intuitively speaking, machine teaching framework studies the problem of developing effective teachers, whereas machine learning framework is geared towards developing effective students. Formally, machine teaching studies the interaction between a teacher and a student, and can be seen as an inverse problem of machine learning with the following key ideas:

- the teacher has a desired goal, and the teacher's objective is to find an optimal training sequence to steer a student/learner towards this goal;
- the interaction between the teacher and the student is limited (e.g., via a set of demonstrations); furthermore, this interaction is expensive, and the objective is to minimize the total cost of interaction required to achieve the goal;
- and importantly, the teacher does not engineer the student's learning process/algorithm—in fact, the student could be a complete black-box.

Below, we describe some of the problems that group members are currently pursuing—this includes ongoing research activities and work planned for the upcoming year.

13.2.1 Algorithmic Foundations of Machine Teaching

Most of the existing algorithmic work on machine teaching has focused primarily on understanding the theoretical connections between the information complexity of teaching vs. learning (or simply put, understanding when and by how much a helpful teacher can speed up the learning process of a student). As a downside, these existing algorithms and theoretical results for machine teaching only consider very simple, idealistic settings—these settings are not rich enough to model any of the above-mentioned real-world motivating applications. The group is working on several problems to build solid algorithmic foundations of machine teaching more suitable for real-world applications; we discuss a few of them below:

Learning process of the student. Most of the theoretical results on machine teaching have focused on students implementing a very specific class of learning algorithm known as "consistent version-space" algorithm. We are working on extending these theoretical results by considering more realistic learning processes of a student, then developing novel teaching algorithms

for these students, and validating our results via extensive user studies. In particular, one of the key focus is to model students with limited memory and computational power, and students implementing sequential learning algorithms such as reinforcement learning algorithms.

Power of the teacher. Most of the existing machine teaching settings assume a very powerful teacher who knows the exact learning process of the student and can observe the current "state" of the student at any time—clearly, this is unrealistic for most of the real-world applications. We are working on characterizing the complexity of teaching when a teacher has limited power or incomplete knowledge of student's model, or when there is a mismatch in reward or feature representations. One of the key ideas that we are pursuing here is that a teacher, via interacting with a student over time and observing student's performance, can do probabilistic inference about the student's learning process. This idea is a key to personalize teaching when teacher's power is limited.

Teaching signals. Existing machine teaching settings usually consider a simple protocol of interaction/communication where a teacher can only provide labeled examples as input to the student. We are looking at more powerful settings that could allow a teacher to use richer teaching signals suitable for the application. For instance, when teaching an inverse reinforcement learning agent (e.g., training medical students via a simulator), the teacher could provide rich demonstrations, visualizations, and relative pairwise comparison between a pair of actions.

Teaching a classroom. Machine teaching setting usually considers the interaction between a teacher and a single student. There are many real-world settings where a teacher has to teach a group of students with a constraint that same teaching sequence should be used for all students. In collaboration with researchers at EPFL, we are working on extending teaching models to this classroom setting and studying the following questions: (i) what is the effect on teaching progress as the diversity of the classroom increases, and (ii) what is the best grouping of students if we are allowed to create a few partitions of the classroom.

13.2.2 Computational Education

One of the important application domains for machine teaching is computational educational; here, machine teaching formulation could enable further

development of rigorous algorithms for intelligent tutoring systems and provide new mathematical models for developing personalized simulator-based training systems. Recall that in an educational setting, we have a human learner as a student, and the focus is on developing algorithms for a teacher in the form of personalized tutoring systems and training simulators. Below, we highlight some of the key application scenarios and research problems that we are working on.

Simulator-based training systems for healthcare. Recently, we have started collaborating with researchers at ETH Zurich and a Swiss company Virtamed in the healthcare domain. Virtamed manufactures surgical simulators whose aim is to teach surgical tasks to medical students using simulator-based training. Technically speaking, here we model a student as an "inverse reinforcement learning" agent, and we seek to develop machine teaching algorithms that can effectively provide a set of demonstrations to be able to communicate an optimal policy to the student. There are several fundamental challenges in tackling this problem, for instance, (i) breaking the complex teaching task into sub-tasks; (ii) dealing with a mismatch in how students and the teacher represent the underlying world (i.e., the state space, features, or the perceived reward values); and (iii) personalization of teaching demonstrations by inferring student's policy over time.

Multi-task teaching in citizen science projects. We are studying the problem of teaching participants of citizen science projects in order to improve their image annotation accuracy. In particular, we consider eBird project, a popular citizen science project for monitoring the bio-diversity of birds species, where participants take pictures of birds with their smartphone apps and annotate them with a name of the species [283, 282]. Our goal is to teach classification rules to participants so that they can distinguish different bird species—this is a very challenging task as we have over 1000 species of birds, many of which are difficult to distinguish visually. From a more technical level, we are studying a novel machine teaching formulation for a multi-task problem setting where the teacher's goal is to teach multiple tasks simultaneously. In order to effectively teach large number of tasks, it requires fundamentally new ideas and models of teaching, for instance, (i) finding an optimal curriculum of tasks (i.e., which tasks should be taught first or how to interleave the teaching process of different tasks); (ii) knowledge transfer across tasks and exploiting the inherent structure among tasks to speed-up teaching (e.g., coarse-to-fine grained teaching when there

is a hierarchical structure); and (iii) incorporating the aspects of student's forgetting behavior when teacher switches from one task to another.

13.2.3 Trustworthy AI

Another important application domain for machine teaching is towards building trustworthy AI and ensuring safety of machine learning systems by deploying optimal defense strategies against adversarial attacks. Recall that, in adversarial attacks known as training-set poisoning, we have a machine learning system as a student (a victim of attacks), and the teacher is an attacker who wants to manipulate the behavior of the student by maliciously modifying the training data.

Optimal defense strategies against training-set poisoning attacks. From an attacker's point of view (the teacher), the optimal training set poisoning attack can be cast as a solution to an optimization problem based on machine teaching framework. Understanding the optimal training-set poisoning attacks can, in turn, help us design optimal defenses against attackers. For instance, we can develop an automated defense system that could flag the parts of training data which are likely to be attacked (based on our model of the teacher) and focus human analysts' attention on those parts. This could drastically increase the chance of detecting such attacks by analysts once they know where to look. Technically, this requires us to develop (i) realistic models of an attacker based on historic data, and (ii) fast approximation algorithms for solving combinatorial optimization problem of machine teaching, for example, by utilizing the submodularity properties of the underlying objective.

Optimal defense strategies in iterative attacks. Training-set poisoning attacks only model a "batch" setting where the attacker first manipulates the training data, and then a machine learning system uses this data. However, an increasing number of real-world machine learning systems (e.g., web search, recommendation systems, financial systems) are iterative algorithms and are being updated over time. An important line of research problem here is to model interactions between an attacker (the teacher) and a learning algorithm (the student) as a repeated game. Here, we would like to design a robust defense system that can anticipate the teacher's actions and develop an optimal defense policy against future attacks.

14 The Software Analysis and Verification Group

14.1 Overview

The report covers the period from August 2015 to January 2018.

The SAV group works on the verification of complex software systems and their components, such as compilers and concurrent algorithms. It does so by developing theories and tools for rigorously applying formal reasoning principles to build correct software systems.

The group's focus has been on weak memory consistency—both on developing suitable semantics for weak memory consistency and on reasoning about the correctness of concurrent programs under such semantics.

Personnel. The group is led by **Viktor Vafeiadis** and consists of one postdoctoral researcher, Azalea Raad, who joined MPI-SWS in July 2017, and two PhD students: Marko Doko and Soham Chakraborty. Ori Lahav was also a postdoctoral researcher in the group until September 2017, when he joined the faculty of Tel Aviv University.

During the reporting period, the group had six interns, each typically staying 2-3 months: Mengda He, Michalis Kokologiannakis, Orestis Melkonian, Anton Podkopaev, Nandini Singhal, Mohit Vyas, Haoze Wu.

Collaborations. The group collaborates with Derek Dreyer's and Rupak Majumdar's groups. We also have collaborated with researchers at Cambridge (Sewell), Uppsala (Sagonas), SNU (Hur), TAU (Lahav, Rinetzky), and Teesside (Qin).

Publications. The group publishes regularly in the top conferences and journals of its field. During the reporting period, group members have coauthored 23 papers: four journal articles [45, 201, 197, 198], four POPL [199, 181, 180, 188], two CGO [73, 74], two ECOOP [179, 263], one AiML [200], one CONCUR [146], one DISC [164], one ESOP [118], one FM [202], one ICFP [246], one NSDI [254], one PLDI [203], one PDP [163], one TABLEAUX [204], and one VMCAI [117] conference papers.

Three among these papers [203, 179, 204] received best paper awards at their respective conferences (PLDI, ECOOP, TABLEAUX). Many of the publications come with machine-checked proof developments in Coq, which are available online.

External funding. The group's research has been partially funded by the European Commission's FP7 FET young explorers grant ADVENT (April 2013 – April 2016). The grant partially funded Ori Lahav's postdoc.

Impact. In a PLDI'17 paper [203], we found flaws in the C++ concurrency model that invalidated the deployed compilation schemes to the Power architecture. We proposed a correction to the C++ standard, which was adopted by the C++ committee.

In a CGO'17 paper [73], we found and reported some miscompilation bugs in LLVM that were fixed.

In SepCompCert [181], we found two compilation bugs in CompCert 2.4 that were then fixed, and later our approach for proving the correctness of separate compilation was adopted by CompCert 2.7.

Invited talks and summer schools. Viktor delivered invited keynotes at CAV 2017 and EuroLLVM 2017, and an invited tutorial at CONCUR 2017. Viktor and Ori gave a summer school on weak memory consistency at Saint Petersburg University in 2017.

Service. Viktor was co-chair of CPP 2017. He served on the program committees of POPL 2018, CPP 2018, ESOP 2017, ITP 2016, NETYS 2016, TASE 2016, PDP/4PAD 2016, and on the external review committees of POPL 2017 and CAV 2016.

14.2 Research agenda

Our research focus has been on reasoning under *weak memory consistency* (WMC), which formalizes the behaviors that may be observed in multi-threaded programs subjected to compiler optimizations and running on modern hardware. These behaviors are hard to specify because they include outcomes that cannot be observed simply by interleaving the memory accesses of the various threads of a program. Moreover, they are potentially hard to reason about because they also render impotent the sophisticated formal methods that have been developed to tame concurrency, which almost universally assume a strong (i.e., sequentially consistent) memory model.

We have worked both on the semantics of WMC for programming languages and on verifying programs under such semantics. On the semantic side, we have worked on the C/C++ concurrency model, finding and correcting a significant flaw in its treatment of SC accesses, on the LLVM model, and on a new "promising" model that solves the "out-of-thin-air" problem

of concurrency semantics. On the verification side, we have developed new program logics (FSL and iGPS) and an efficient model checker for concurrent C/C++ programs.

In addition, we have worked on *compositional compiler correctness*—that is, on proving the correctness of compilers that support separate compilation and linking. In collaboration with Derek Dreyer's group and Hur's group at Korea, we have developed two approaches: a heavier-weight one [246] that even allows linking of code produced by different verified compilers, and a much lighter-weight one (SepCompCert [181]) that supports only linking of code produced by the same compiler but which can readily be applied to CompCert with only minor adaptations to its codebase.

14.2.1 WMC Semantics

C/C++ concurrency model [199, 203] In 2011, the C and C++ standards introduced the language's concurrency model defining the semantics of concurrent memory accesses in C/C++. This model supports racy "atomic" accesses at a range of different consistency levels, from very weak consistency ("relaxed") to strong, sequential consistency ("SC").

In a series of papers, we studied formally various aspects of the C/C++ semantics, which led to a few unexpected outcomes. We found that the semantics of SC atomic accesses is flawed, in that (contrary to previously published results) both suggested compilation schemes to the Power architecture are unsound. Futher, the semantics of SC fences is overly weak and does not guarantee sequential consistency even when placed between every two atomic accesses of a program. Finally, the semantics of release/acquire atomics can be strengthened without any performance degradation.

Based on these observations, we proposed a model, called RC11 (for Repaired C11), with a better semantics for SC accesses that restores the soundness of the compilation schemes to Power, maintains the DRF-SC guarantee, and provides stronger, more useful, guarantees to SC fences. In addition, we formally proved, for the first time, the correctness of the proposed stronger compilation schemes to Power that preserve load-to-store ordering and avoid "out-of-thin-air" reads. The C++ standards committee largely accepted our proposed changes to the model and incorporated them in the next revision of the standard.

LLVM concurrency model [73, 74] In its informal documentation, LLVM discusses the weak memory model it assumes about its intermediate language. The intended LLVM model is quite similar to the C/C++

one but has some differences that affect the set of allowed program transformations/optimizations.

Soham Chakraborty is working on a formalization of the LLVM memory model using event structures, and on proving that it allows the intended program transformations, as well as correct compilation to hardware architectures. As part of this work, in order to determine experimentally what the LLVM model is, he also constructed a translation validator that can check whether an LLVM optimization pass can be decomposed into a sequence of allowed reordering and elimination transformations. Using the translation validator, we found an reported three concurrency compilation bugs in LLVM 3.6, which were fixed in the next LLVM release.

Promising semantics [180, 263] Despite many years of research, it has proven very difficult to develop a memory model for concurrent programming languages that adequately balances the conflicting desiderata of programmers, compilers, and hardware. In this work, we propose the first relaxed memory model that (1) accounts for a broad spectrum of features from the C/C++11 concurrency model, (2) is implementable, in the sense that it provably validates many standard compiler optimizations and reorderings, as well as standard compilation schemes to x86-TSO, ARM, and Power, (3) justifies simple invariant-based reasoning, thus demonstrating the absence of bad "out-of-thin-air" behaviors, (4) supports "DRF" guarantees, ensuring that programmers who use sufficient synchronization need not understand the full complexities of relaxed-memory semantics, and (5) defines the semantics of racy programs without relying on undefined behaviors, which is a prerequisite for applicability to type-safe languages like Java.

The key novel idea behind our model is the notion of promises: a thread may promise to execute a write in the future, thus enabling other threads to read from that write out of order. Crucially, to prevent out-of-thin-air behaviors, a promise step requires a thread- local certification that it will be possible to execute the promised write even in the absence of the promise. To establish confidence in our model, we have formalized most of our key results in Coq.

14.2.2 Software Verification under WMC

FSL [117, 118, 163] In the previous reporting period, we had developed a number of program logics (RSL, GPS, OGRA) that were suitable for reasoning about the release-acquire fragment of the C/C++11 memory model. Other researchers have also developed techniques (such as iCAP-TSO) for

reasoning under the even stronger x86-TSO model. Yet, a number of concurrent libraries are implemented using the even weaker (less synchronizing and typically more efficient) features of the C/C++ memory model—that is, relaxed accesses and release/acquire fences.

To be able to reason about such programs, we developed *fenced separation logic (FSL)*, a program logic extending RSL with simple but powerful rules for reasoning about relaxed accesses and release/acquire fences. In follow up papers, we used FSL to verify Rust's optimized atomic reference counting (ARC) library, and extended GPS with similar rules to FSL for reasoning about fences and relaxed accesses.

iGPS [179] To establish the soundness of the various program logics (RSL, GPS, OGRA, FSL) that were developed for fragments of the C/C++11 memory model, we had devised a novel proof technique that worked directly over the C/C++11 axiomatic model by suitably annotating program execution graphs. This approach to proving soundness, while doable, is quite complex, and diverges from usual operational way of proving soundess of logics over sequential consistency. In iGPS, we therefore tried to reconcile this divergence. We took GPS, an advanced program logic earlier developed by our group, and reproved its soundness in Coq over an operational encoding of release-acquire consistency using the Iris logical framework. This has a number of benefits: it straighforwardly handles high-order code and ghost resources, and allows us to quickly experiment with alternative proof rules.

RCMC [188] Besides program logics, we also considered bounded model checking (BMC) as technique for verifying concurrent programs. While BMC can only verify correctness up to a certain bound on the length of the program's execution, it is a fully automated and widely applicable technique that can quickly expose programming errors and can achieve high levels of confidence in the correctness of software.

We therefore developed a new model checking algorithm for verifying concurrent programs running under RC11 [203] (our repaired version of the C/C++11 memory model). Our approach works directly by enumerating all consistent execution graphs of a program in a stateless matter, i.e., without recording the set of graphs already generated. Nevertheless, we show that in the absence of RMW instructions and SC atomics, our algorithm never revisits the same execution graph. We also implemented this algorithm in a tool, called RCMC, which outperforms significantly the state-of-art model checkers on a range of concurrent programs.

15 The Information Security and Cryptography Group

15.1 Overview

The report covers the period from August 2015 through July 2017. The group was led by **Michael Backes**, who was a Max Planck Fellow at the institute through July 2017. The research of his group focuses on the theoretical foundations and applied aspects of information security, privacy, and cryptography. Major research topics have included: the design and analysis of security protocols, privacy and anonymity, linking formal methods and cryptography, and novel approaches for OS and software security.

Personnel. The group is led by Max Planck Fellow **Michael Backes**. Michael additionally held the chair for information security and cryptography (IS&C) at Saarland University, and recently became the Chairman and Founding Director of the CISPA Helmholtz Center. Previously, Backes was the Director of the Center for IT Security, Privacy, and Accountability (CISPA). The group until recently had one postdoc (Dario Fiore), who is now faculty at IMDEA, Spain. The Max Planck fellowship was used to fund internships and visiting researchers (most recently Kangjie Lie from Georgia Tech), who complement the IS&C university group.

Publications. The group publishes regularly in the top conferences and journals in the field of security. During the review period, group members have co-authored 65 publications [10, 171, 247, 260, 39, 107, 34, 159, 33, 43, 44, 31, 214, 216, 194, 142, 8, 32, 22, 292, 295, 280, 42, 35, 28, 25, 27, 40, 152, 276, 193, 294, 30, 63, 37, 26, 213, 41, 317, 38, 36, 9, 7, 215, 293, 29, 24, 23, 300, 6, 3, 5, 103, 4, 186, 46, 314, 68, 14, 1, 233, 251, 2, 169, 313].

Funding, awards, and service. Backes is the recipient of an ERC Synergy award (together with Druschel, Majumdar, and Weikum). He is also the speaker for the DFG Collaborative Center (SFB) on "Understanding and Controlling Privacy." He is the Director of the CISPA-Stanford Center, jointly with John Mitchell and a Principal Investigator and Vice-coordinator of the Cluster of Excellence on Multimodal Computing and Interaction (MMCI). In addition, he recently started as the Chairman and Founding Director of CISPA Helmholtz Centrum.

He received the NSA Cybersecurity Research Award (2017), the IEEE Golden Core Award (2017), the CNIL-INRIA Privacy Award (2017), and a

Teaching Award of the State of Saarland (2015).

15.2 Research agenda

The group's research interests are in theoretical foundations and applied aspects of information security, privacy, and cryptography. Contributions have been made in the following areas: (1) the design and analysis of security protocols; (2) privacy and anonymity; (3) linking formal methods and cryptography; (4) novel approaches for OS and software security, including the analysis of mobile applications; and (5) interactions of security and privacy with machine learning. In the last two years, the group's research interest have focused increasingly on the area of privacy assessment and privacy-preserving computation.

Part II
Details

16 Details

In this section, we provide detailed information about the institute, following the outline required by the Max Planck Society's rules for scientific advisory board status reports.

16.1 Structure and organization

Faculty As discussed in Section 1.1, the institute has a flat organization, with currently thirteen independent research groups, each led by a faculty member (tenure-track, tenured, or director). There are two research group leaders and both are formally in the Rigorous Software Engineering group. In addition, Robert Harper (CMU) has an appointment as an external scientific member, and Rodrigo Rodrigues as an adjunct faculty member. Michael Backes was a Max Planck Fellow until 2017.

The faculty appointment dates, tenure status, and (for tenured faculty) retirement dates are shown in Figure 1.

Leadership As stated in the institute bylaws, institute policy is decided jointly by the faculty. The faculty typically meets weekly, with the location alternating between the two sites. The day-to-day operation of the institute is in the hands of the Managing Director (currently Paul Francis), assisted by the head of the administrative department, Volker Maria Geiss. The position of Managing Director rotates among the directors (normally every two years).

Administrative support The MPI-SWS and the MPI for Informatics (MPI-INF) in Saarbrücken are supported by a shared administrative department headed by Volker Maria Geiss. The department provides personnel, finance, and purchasing services. Geiss also handles much of the public relations, relations with local governments, and relations with other research institutions in Kaiserslautern and Saarbrücken. We share the core IT support group with MPI for Informatics. Separately MPI-SWS has its own user-facing IT support group. Both the core IT and user-facing IT groups report to Geiss as well.

MPI-SWS shares a library jointly with MPI-INF, DFKI (German Research Center for Artificial Intelligence), and the Mathematics and Computer Science departments of Saarland University. The joint library reports to Volker Maria Geiss.

Group Name	Group Leader	Start	Status	Retire
Real-Time Systems	Brandenburg	2011	tenured in 2018	2048
Foundations of Programming	Dreyer	2008	tenured in 2013	2047
Distributed Systems	Druschel	2005	director	2025
Large Scale Internet Systems	Francis	2009	director	2023
Foundations of Computer Security	Garg	2011	tenured in 2018	2046
Networked Systems	Gummadi	2005	tenured in 2012	2046
Rigorous Software Engineering	Majumdar	2010	director	2042
Software Analysis and Verification	Vafeiadis	2010	tenured in 2016	2048
Networks and Machine Learning	Gomez Rodriguez	2014	tenure-track	—
Automated Verification and Approximation	Darulova	2015	tenure-track	—
Foundation of Algorithmic Verification	Ouaknine	2016	director	2037
Practical Formal Methods	Christakis	2017	tenure-track	—
Machine Teaching	Singla	2017	tenure-track	—

Figure 1: MPI-SWS research groups

Administrative assistance for faculty, staff, postdocs and students is provided by an administrative team consisting of five members—Annika Meiser and Claudia Richter in Saarbrücken, Vera Schreiber, Susanne Girard, and Roslyn Stricker in Kaiserslautern. In addition, Maria-Louise Albrecht serves as coordinator for the MPI-SWS graduate program.

IT services Support for core information technology services (network and core network services, telephony, and storage/email/web services) is provided by a team headed by Jörg Herrmann. This team (currently 13 members) is also shared with the MPI for Informatics. Working together with the core team is a five-member IT support team (headed by Christian Mickler), which provides dedicated support for the all other IT needs of SWS researchers, such as audio/video conferencing, hardware, and software issues.

Locating this dedicated team alongside the offices of SWS researchers (both in Kaiserslautern and Saarbrücken) has made it much easier for them to respond effectively to researchers' often-spontaneous requests for assistance.

English language support It is critically important that young researchers develop their communication skills. Moreover, we feel that English language support is particularly important for non-native English speakers. Therefore, the institute has a strict policy of using English as the working language. We feel this is necessary, not only to accommodate our highly international staff, but also to help the non-native English speakers develop their language skills.

The institute employs an English support coordinator who provides English language speaking, writing, and presentation support for all institute members. Rose Hoberman, who currently occupies the position, has a Ph.D. in computer science from CMU. She offers regular courses on presentation, reading, and writing skills, and additional soft skills courses as needed. She also provides feedback on institute members' presentations, papers, and other documents. We plan to hire additional staff as the institute grows.

Research support team The institute also has several funded positions available for software developers. We have been filling these positions on a temporary per-project basis. In this reporting period, we have used six such developers, Cedric Gilbert (email attachment malware), Matthias Kretschmer, Cristian Berneanu, and Sasa Juric (anonymized analytics). We have also used additional developers, Jeff Hoye and Jeff Fischer, on a consulting basis for both research projects and institute administrative tools such as our admissions system.

16.2 Research program and groups

This information is provided in previous sections.

16.3 Personnel structure

Currently, the institute has 113 members (excluding interns and visitors). Among these, there are 77 researchers and 36 non-research staff. Of these, 23 are administrative staff shared with MPI-INF, and 13 are IT staff. During the reporting period, 48 members joined MPI-SWS, and 45 left.

Permanent faculty	9
Tenure-track faculty	4
Permanent staff	37
Temporary contracts	10
Postdocs	16
PhD students	37

16.4 Structure of the budget

The institute's total yearly budget is EUR 9.77M per year. Of that, the institute's yearly expenditure amounted to EUR 9.3M per year, including EUR 2.91M for material expenses, EUR 1.16 M for investment in equipment, EUR 4.44M for personnel expenses (excluding stipends) and EUR 807K for graduate and postdoctoral stipends and contracts. (Personnel funds can be used to fund additional stipends but not vice versa.)

The institute is alloted 6 senior faculty (director, W3) positions, and up to 12 junior and mid-career (tenure-track or tenured, W2) positions.

16.5 Provision of material, equipment, and working space

Material The nature of the institute's research in software systems is such that it does not require materials beyond normal office supplies.

Equipment The institute has a state-of-the-art, reliable and fail-safe computing infrastructure. A redundant network backbone of 10 Gigabit links connects the Kaiserslautern site, the Saarbrücken site, the MPI-INF, and Saarland University via a multi-gigabit link to the X-WIN—the German research network. Basic network services, as well as email and web servers, are implemented in a reliable and fail-safe manner. Storage services provide backup and access to more than 1PB of storage. All services are monitored by a system that notifies the IT staff via e-mail in case of trouble. Institute members have personal desktop and notebook computers.

The institute currently maintains three clusters for research. The Blade-Server cluster has thirty-two Xeon octa-core systems with each two CPUs. A second cluster of twenty-five nodes with quad twelve-core CPUs are equipped with 1.5 TB RAM. And a third cluster of five-teen nodes with extra GPUs (Maxwell-Architecture). Additionally to the clusters, the institute maintain a NVIDIA DGX-1 system. All clusters are connected to the institute's intranet and have direct access to the storage services.

The computing infrastructure will be expanded as needed to accommodate new research demands and growth. For instance, future faculty hires may require more specialized laboratories.

Space At our current level of staffing, the two buildings in Saarbrücken and Kaiserslautern provide ample office space, lab space, meeting and conference rooms, open space, event space, and machine room space. Each site is able to accommodate all members of the other site on our weekly visit days. The lecture halls and meeting rooms of both sites are periodically used by external organizations. The Kaiserslautern building is also used by a group from the TU-KL math department.

16.6 Junior scientists and guest scientists

Junior scientists Attracting, supporting, mentoring and creating opportunities for outstanding young researchers is a top priority at the institute.

The purpose of the institute's tenure-track systems is to attract the very best young PhDs internationally and provide them with conditions (independence, resources, mentorship, full participation in the institute governance) that will allow them to grow as researchers and future leaders. The institute has a formal faculty mentorship program.

We have an active program to attract and support outstanding postdoctoral researchers from diverse backgrounds. Postdoctoral positions are normally granted for two years, and can be extended to three years. Currently, we have 16 postdocs from twelve countries. A list of our current postdocs can be found online at `https://www.mpi-sws.org/people/`.

A high priority for the institute is to attract the best graduate students and provide them with the training necessary for them to obtain academic and research positions at the world's best universities and research labs. We seek to maintain a highly talented, highly motivated and diverse body of graduate students. Moreover, we provide intensive training in small groups (less than six students per faculty). We emphasize high-risk, high-impact research and publication in top venues.

We currently have 37 doctoral students from 11 countries. A list of our current doctoral students can be found online at `https://www.mpi-sws.org/people/`.

Guest researchers As part of the institute's strategy to increase visibility, create opportunities for collaborations with other institutions, and

contribute to a vibrant intellectual environment, the institute has a very active program for short and longer term visitors at all seniority levels.

During the reporting period, MPI-SWS hosted 81 undergraduate and graduate interns.

Researchers from other institutions frequently come for research visits. There were around 41 such short-term visitors, including: Enrico Bini (University of Turin), Rob Davis (Universtiy of York, UK), Geoffrey Nelissen (Universite libre de Bruxelles), Sasa Misailovic (University of Illinois at Urbana-Champaign), Magnus Myreen (Chalmers University), Zachary Tatlock (University of Washington), Eric Goubault (Ecole Polytechnique), Sylvie Putot (Ecole Polytechnique), Wolfgang Ahrendt (Chalmers University), Anastasia Volkova (INRIA), Amir Yehudayoff (Technion – Israel Institute of Technology), Thomas Colcombet (IRIF, Université Paris Diderot), Steve Zdancewic (University of Pennsylvania), Scott Owens (University of Kent), Gilles Barthe (IMDEA Software Institute), Ashutosh Gupta (IIT Bombay), Samir Khuller (University of Maryland), Damon McCoy (New York University's Tandon School of Engineering), Andrew Baumann (Microsoft Research Redmond), Oriana Riva (Microsoft Research), Amal Ahmed (Northeastern University), Marco Caccamo (University of Illinois), Cecilia Mascolo (Cambridge), Mark Crovella (Boston University),Chandu Thekkath (Microsoft Research India), Ashish Goel (Stanford University), Cecilia Mascolo (University of Cambridge), Ulfar Erlingsson (Google USA), Dina Papagiannaki (Telefonica, Barcelona), Rakesh Agrawal (EPFL), James Worrell (University of Oxford), Richard Murray (Caltech), Mijung Park (MPI-IS), Robert West (EPFL), Shuvra Bhattacharya (University of Maryland), Hakan Ferhatosmanoglu (Bilkent University), Niloy Ganguly (IIT Kharagpur), Ponnurangam Kumaraguru (IIIT Delhi), Cezara Dragoi (INRIA),Thomas Wies (NYU), Philipp Haller (KTH Royal Institute of Technology)

We have also had four long-term visitors: Geoffrey Nelissen, postdoc from CISTER - Instituto Superior de Engenharia do Porto (ISEP), Portugal, who has spent semi-monthly visits throughout 2017 to collaborate with the reseach group of Björn Brandenburg. Gummadi's group hosted three long-term visitors supported by Humboldt faculty fellowships and awards: Prof. Hakan Ferhatosmanoglu from Bilkent University, Turkey, Prof. Patrick Loiseau from EURECOM, France, and Prof. Fabricio Benevenuto from UFMG, Brazil who spent his sabbatical here in 2017. Prof. Lorenzo Alvisi (University of Texas at Austin and Cornell University) has visited the group (and the institute) in the summers of 2016 and 2017, supported by a Humboldt Research Award. Prof. Bobby Bhattacharjee (University of Maryland, College Park) visited the group and institute during

his sabbatical in 2016.

16.7 Equal opportunity

Ensuring gender diversity is a well-known perennial problem in computer science departments worldwide. At the beginning of the review period, 20.7% of the scientific staff were women. The current number is 17,11% (3 of 10 temporary contracts, 6 of 37 doctoral students, 4 of 16 postdocs, and 2 of 13 faculty).

16.8 Relations with domestic and foreign research institutions

Local network We continue to work towards integration with Saarland University (UdS) and University of Kaiserslautern (TU-KL). Professors from the two departments are present in our faculty recruitment committees and our graduate student admission committees. MPI-SWS is a member of the UdS CS graduate school. There are a number of joint research projects with UdS and TU-KL faculty, among them are: Maria Christakis, Rupak Majumdar, and Joël Ouaknine worked with Vera Demberg, Bernd Finkbeiner, Matthias Hein, Holger Hermanns, Jörg Hoffmann, and Antonio Krüger (all UdS). Maria Christakis collaborated with Jens Schmitt, Gerhard Fohler, Reinhard Gotzhein, Wolfgang Kunz, Peter Liggesmeyer, Norbert Wehn (all TU-KL). Deepak Garg works as part of DFG grants with Christian Hammer (UdS). Manuel Gomez-Rodriguez as well as Krishna Gummadi collaborated with Michael Backes (UdS/CISPA). Daniel Neider works with Martin Zimmermann (UdS) and Adish Singla with Verena Wolf (UdS).

Druschel is a PI in CISPA, the Excellence Cluster at UdS. He is also a co-PI in Saarland University's Collaborative Research Center on Methods and Tools for Understanding and Controlling Privacy. Druschel, Francis, Garg, and Gummadi are part of an SFB Grant at UdS. Druschel and Majumdar are co-PIs on the imPACT ERC Synergy Grant.

MPI-SWS faculty have taught courses in their areas of expertise. During this reporting period, faculty at MPI-SWS taught the following courses:

- Semantics, Saarland University, Winter 2015/2016
- Operating Systems, Saarland University, Winter 2015/2016
- Privacy, Accountability, Compliance, and Trust in Internet Applications, Saarland University, Winter 2015/2016

- Social and Information Networks: Models and Machine Learning Methods, TU Kaiserslautern, Winter 2015/2016
- Human vs. Algorithmic-Decision Making: Bias, Discrimination, Fairness and Transparency, Saarland University, Winter 2015/2016
- Social Media Analysis, Saarland University, 2016
- Secure information flow control in systems, Saarland University, 2016
- Approximate Computing: Promise or Hype?, Saarland University, 2016
- Complexity Theory, TU Kaiserslautern, Winter 2016/2017
- Distributed Systems , Saarland University, 2016/2017
- Advanced Automata Theory, TU Kaiserslautern, 2017
- Static Program Analysis, Saarland University 2017
- Complexity Theory, TU Kaiserslautern, Winter 2017/2018
- Operating Systems, Saarland University, Winter 2017/2018
- Concurrency Theory, TU Kaiserslautern, Winter 2017/2018
- Semantics, Saarland University, Winter 2017/2018

International Institute members maintain numerous collaborations with researchers at international universities and research institutions, including:

Universities TU Kaiserslautern, Saarland University, TU Braunschweig, Sapienza Universita di Roma, TU Dortmund, University of York, George Washington University, Technical University of Dresden, ETH Zurich, Switzerland, University of Texas at Austin, University of Memphis, Chalmers University, University of Cambridge, Ecole Polytechnique, University of Washington, TU Munich, EPFL, MIT, USC (California), UCL (London), University of Quebec, INRIA, Aarhus University, University of Maryland , University at Buffalo SUNY, University of Edinburgh, Carnegie Mellon University, TU-Vienna, KU-Leuven, University of Potsdam, University of Edinburgh, Northeastern University, IIT Kharagpur, Sharif University, New York University, KAIST, UFMG, Yale University, University of Colorado Boulder, Australian National University, Indian Institute of Science, University of

Illinois, University of Alberta, University of California, Oxford University, Universidade Federal de Minas Gerais, Université de Rennes, Ecole normale supérieure Paris-Saclay, California Institute of Technology, George Mason University, Purdue University, Uppsala University, Seoul National University, Tel Aviv University, Teesside University.

Research Institutes INRIA, IASI–CNR, Gran Sasso Science Institute, Max Planck Institute for Informatics, IMDEA Software Institute, Max Planck Institute for Intelligent Systems, Georgia Institute of Technology, Max Planck Institute for Collective Goods, Max Planck Institute for Demographic Research, Eurecat, Alan Turing Institute

Industry Verimag, ONERA, Bosch Corporate Research, SYSGO AG, Microsoft Research, Aircloak GmbH, AT&T , Toyota Research Institute

16.9 Activities regarding the transfer of knowledge/relations with industry

Brandenburg's group has a collaboration with the SYSGO AG on the problem of integrating support for latency-sensitive, low-criticality workloads into existing certified real-time operating systems for high-criticality applications resulted in a paper presented at RTNS'17 [309]. Further, there is an ongoing collaboration with Bosch Corporate Research (the group of Arne Hamann) centered on OS support for consolidated automotive workloads. Additionally, Brandenburg's group is working jointly with Microsoft Research on the problem of horizontal on-demand scaling of compute infrastructure for *machine learning as a service* (MLaaS) workloads, which resulted from Gujarati's internship at their location in Redmond, WA, has resulted in a publication accepted at Middleware'17 [161].

Christakis's group also has been working with Microsoft. They have published a paper at CHI'18, titled "CFar: A Tool to Increase Communication, Productivity, and Review Quality in Collaborative Code Reviews". In this paper, they designed a collaborative code review system, CFar, that introduces an automated code reviewer based on program-analysis technologies. In particular, their automated reviewer inserts issues detected by the analyses into an otherwise human-human collaborative code review. As a result, they observed that communication and productivity of programmers increased and that the quality of their code improved. The software they developed for this paper is currently being used by various product teams at Microsoft.

Francis' group uses the startup Aircloak as a critical part of its research agenda. During the reporting period Francis applied for one new patent (for Diffix). The primary output of Francis's group is the product developed by Aircloak that implements the jointly-developed Diffix anonymization approach, and the associated documentation. The product is in use commercially, and the software is proprietary. Three other patents that were started before the reporting period, for the most part have either been awarded or are still under evaluation.

Gummadi's group worked with Facebook engineers to plug serious privacy vulnerabilities (where any advertiser could learn about personally identifiable information such as phone numbers or website visits of a Facebook user) with their advertising APIs.

Zufferey is working with Toyota research institute on techniques for constraint solving for non-linear equations.

Majumdar's group continues to work closely with Toyota on a Conformance testing tool for Simulink models used in internal testing at Toyota.

16.10 Symposia, conferences, etc.

The institute organized the fourth institute retreat in August 2016 at Heidelberg. The primary purpose of this retreat was to make all the research groups at MPI-SWS aware of one another's ongoing work. During the retreat, faculty and students presented and discussed their current work. Faculty also used the opportunity to gather feedback and present the future goals and vision of the institute. Other activities included work-in-progress presentations, discussions devoted to academic issues and institute life, talks and discussions on the nature and methodologies of CS research, birds-of-a-feather sessions, and discussions to help students make the most of their graduate studies and prepare for future roles as leading researchers and faculty.

The institute has an ongoing distinguished lecture series. The purpose of this series is to bring senior leaders in software systems to the institute (typically, for two days), have them give a talk, showcase the institute, have them meet faculty, postdocs and students, and last but not least, seek feedback on our strategy and advice in identifying potential hires. In this reporting period, we have had 9 distinguished lecturers: Marco Caccamo (University of Illinois), Chandu Thekkath (Microsoft Research India), Ashish Goel (Stanford University), Cecilia Mascolo (University of Cambridge), Ulfar Erlingsson (Google USA), Dina Papagiannaki (Telefonica, Barcelona), Rakesh Agrawal (EPFL), James Worrell (University of Oxford), Richard

Murray (Caltech). The lecture abstracts and titles are available online at https://www.mpi-sws.org/events/

This series has been very effective in raising the institute's visibility and identifying potential hires, and we have received valuable feedback and advice regarding our strategy.

A number of the faculty have given keynote and invited talks at various institutions and conference during the reporting period. These are detailed within the individual sections.

16.11 Committee work of the faculty

MPI-SWS researchers have served on the program committees of over 135 conferences and workshops, and have chaired or co-chaired the PCs of 13 conferences and 6 workshops. The information is provided in detail in the individual research group sections.

Björn Brandenburg was PC co-chair of EMSOFT'17, is currently serving as PC chair of EMSOFT'18, and is an associate editor of ACM TECS. Members of the group have further served as publication chair of ECRTS'17, publicity chairs of RTSS'15–'17, as reviewers for various journals, and served on the PCs of RTSS'16, ECRTS'16–'17, RTAS'16–'18, EuroSys'16, EMSOFT'16, SYSTOR'16, and RTNS'16. Björn Brandenburg is the institute's current CPTS representative.

Maria Christakis is chairing the PLDI'18 Student Research Competition and the ECOOP'18 Artifact Evaluation. She was a PC member for VMCAI'18, is currently an ERC member for PLDI'18, and has accepted to serve the Program Committees of OOPSLA'18, iFM'18, ACM Student Research Competition'18, TACAS'19, and ICSE'19.

Eva Darulova was CAV'17 workshop chair, co-organiser of Dagstuhl seminar 17352 (Aug'17), and co-organiser of PLMW at POPL'17. Darulova served on the program committees of Scala'16, VMCAI'16, CC'17, WAX'17, NSV'17, Onward!'17, Scala'17, CGO'18 and PLDI'18 and on the external review committees of PLDI'16 and PLDI'17. She has also reviewed for the journals ACM Transactions on Mathematical Software, ACM TOPLAS, IEEE Transactions on Computers, Software Testing and Verification and Reliability. Eva Darulova has served as the equal opportunity officer since November 2016.

Derek Dreyer is serving as General Chair of ICFP'19 in Berlin. He is also serving as Steering Committee Chair for PLMW, and as a member of the Steering Committee for ICFP. In July 2017, Derek Dreyer was appointed as an Associate Editor of ACM Transactions on Programming Languages

and Systems (TOPLAS). He also continues to serve on the editorial board of the Journal of Functional Programming (JFP), and served as guest co-editor of a special 2016 issue of JFP [120] devoted to selected papers from ICFP'14. Derek Dreyer has served on the program committees of POPL'17 and FSCD'16, and will serve on the program committee of OOPSLA'18.

Peter Druschel is a co-PI in Saarland University's MMCI Cluster of Excellence and the Saarbrücken Graduate School in Computer Science, funded by the German National Science Foundation (DFG). He is also a co-PI in Saarland University's Collaborative Research Center on Methods and Tools for Understanding and Controlling Privacy, funded by the DFG. From 2011-2017, he was co-PI and assistant director of the Center for Information Security, Privacy and Trust, funded by the German ministry of science. Jointly with Rupak Majumdar, Michael Backes (CISPA), and Gerhard Weikum (MPI for Informatics), Druschel is a co-PIs on an ERC Synergy Grant on Privacy, Accountability, Compliance, and Trust in the Internet. Peter Druschel serves on the editorial boards of the Communications of the ACM (CACM) and the Royal Society Open Science Journal through 2017. He served on the Technical Advisory Board (TAB) of Microsoft Research, Cambridge, through 2016 and he continues to serve on the TAB of Microsoft Research, India. He also serves on the scientific committee of the Laboratory on Information, Networking and Communication Sciences (LINCS), Paris. Peter Druschel was a member of the selection committee for the EuroSys Jochen Liedtke Young Researcher Award in 2016 and chaired that committee in 2017. He was a member of the ACM SIGOPS Mark Weiser Award Committee in 2016 and 2017, and will chair that committee in 2018. He was a member of the SIGCOMM Lifetime Award Committee in 2016. Peter Druschel also served on the program committees of OSDI in 2016, SOSP and HotMobile in 2017. He served on the strategy committee (Perspektivenkommission) of the Chemistry, Physics, and Technology Section (CPTS) of the MPS through June 2016. He was elected to serve as the deputy chair of the CPTS starting in June 2018 and serve as the chair for a 3-year term starting in 2020. During the reporting period, Peter Druschel also served on two presidential committees of the MPS: The committee on the Support of Junior Scientists and the committee on IT Security. He continues to serve on the selection panel of the joint Fraunhofer/Max Planck research program. Lastly, Peter Druschel co-organized a Symposium on Foundations of Security and Privacy for the CPTS in July 2015, led a task force to develop a proposal for a new MPI for Cybersecurity and Privacy, and currently serves on committees to identity the location and founding directors for the institute, which is expected to start in 2018.

Paul Franics was NSDI 2016 PC member. Further he evaluated for NWO grant application (Netherlands Organisation for Scientific Research), for ERC SAP (Assessment of Completed Projects), and acted as WWW 2017 Security Track PC member.

Deepak Garg is the chair of the steering committee of the Foundations of Computer Security Workshop (FCS) since 2017. He is also a member of the steering committees of the IEEE Symposium on Security Foundations (CSF) since 2012 and the Conference on Principles of Security and Privacy (POST) since 2016. He chaired FCS for a second time in 2016. He and Marco Patrignani are also founding members of the new POPL-affiliated PriSC workshop on secure compilation. Deepak Garg is also a co-organizer of a Dagstuhl seminar on secure compilation to be held in May, 2018. During the reporting period, he served on the program committees of CCS '16, CSF '18, EuroS&P '17 '18, POST '16 '17 and PLAS '17. He has also been the publications chair of CSF since 2012.

Manuel Gomez-Rodriguez has served as Senior Program Committee (SPC) member at NIPS (2016-2017), AISTATS (2018), WSDM (2018) and SDM (2018) and as Program Committee (PC) of ICML (2016-2017), ICLR (2018), KDD (2015-2017), WSDM (2016-2017), WWW (2016-2018), ICWSM (2016-2018), SDM (2016-2017), AAAI (2016-2018), AISTATS (2016-2017) and IJCAI (2016). Moreover, he has served as a reviewer for the Netherlands Organization for Scientific Research and the Foundation for Polish Science.

Krishna Gummadi has served as a general co-chair for AAAI's ICWSM 2016 and program co-chair for the first Data Transparency Lab (DTL) Conference and Fairness, Transparency, and Privacy Workshop at DALI 2018. He has also served on the program committees of KDD 2016-2018, SIGIR 2018, ICDE 2018, WSDM 2016-2018, WWW 2016-2018, FAT-ML 2016-2017, FAT* 2018, WebScience 2016, and IMC 2016. Gummadi also served as an associate editor of ACM Transactions on the Web between 2014 and 2017. He is currently serving as the associate editor for the new ACM Transactions on Social Computing and EPJ Data-Science Journal. Additionally he serves as a steering committee member of the Measurement-Lab (M-Lab), Conference on Fairness, Accountability, and Transparency (FAT*), and AAAI's International Conference on the Web and Social Media (ICWSM). He also served on the selection committes for WWW 2018 Test-of-Time Award, CNIL-INRIA Privacy Research Award 2017, and Data Transparency Lab (DTL) grants 2015-2017.

Rupak Majumdar served on the program committees of several conferences in the last two years, and chaired RV 2015, POPL 2016, and CAV 2017. He organized the Dagstuhl seminars "Formal synthesis of cyber-physical sys-

tems," (jointly with Calin Belta, Majid Zamani, and Matthias Rungger) and "Game theory in AI, Logic, and Algorithms" (jointly with Swarat Chaudhuri, Sampath Kannan, and Michael Wooldridge). Rupak Majumdar serves on the POPL Steering Committee and is an Associate Editor of TOPLAS and was previously an Associate Editor of TECS.

Daniel Neider has served as program committee member at VMCAI (2018). Damien Zufferey was a PC member for CAV 2017, MEMOCODE 2017, SYNT 2017, SMT 2017, TAPAS 2017, TACAS 2018 and CAV 2018, was an ERC member for POPL 17, reviewed for CONCUR 2017, and has accepted to serve the Program Committees of SAS 2018, PLDI Student Research Competition 2018, and the Scala Symposium 2018.

Joël Ouaknine served on the PC of FoSSaCS 2018 and as PC Chair of LICS 2017. Since 2017, he also serves on the Steering Committee of LICS. He is Associate Editor for the Journal of Computer and System Sciences, Elsevier.

Adish Singla has served as a Program Committee (PC) member of NIPS conference (2017). Singla co-organized a workshop at NIPS conference (2017) on "Teaching Machines, Robots, and Humans", and also gave an opening tutorial at the workshop.

Viktor Vafeiadis was co-chair of CPP 2017. He served on the program committees of POPL 2018, CPP 2018, ESOP 2017, ITP 2016, NETYS 2016, TASE 2016, PDP/4PAD 2016, and on the external review committees of POPL 2017 and CAV 2016.

16.12 Publications

All publications are listed in the per-group sections. Here, we provide summary information.

During the reporting period, the institute produced 292 peer-reviewed conference, workshop, and journal publications: [160, 316, 52, 57, 69, 230, 241, 53, 62, 238, 256, 309, 149, 237, 161, 239, 271, 242, 172, 236, 240, 19, 162, 76, 77, 165, 104, 173, 328, 181, 180, 175, 203, 246, 176, 191, 299, 296, 179, 211, 11, 166, 210, 129, 126, 227, 249, 54, 138, 189, 301, 306, 125, 143, 287, 223, 170, 108, 145, 286, 273, 274, 124, 110, 135, 114, 275, 225, 205, 289, 224, 75, 133, 221, 168, 87, 285, 130, 146, 115, 132, 80, 100, 187, 208, 217, 218, 288, 113, 106, 131, 102, 84, 81, 88, 79, 20, 85, 83, 82, 86, 267, 111, 98, 228, 127, 250, 212, 13, 99, 296, 258, 270, 257, 92, 49, 261, 269, 56, 268, 112, 12, 235, 190, 105, 136, 323, 322, 226, 297, 321, 21, 304, 326, 122, 184, 308, 174, 34, 54, 123, 137, 150, 325, 151, 321, 155, 158, 322, 323, 156, 157, 140, 290, 147, 21, 51, 50, 319, 71, 264, 195, 318, 72, 153, 279, 70, 154, 320, 16,

128, 17, 310, 234, 312, 311, 23, 231, 232, 278, 148, 281, 277, 90, 60, 67, 59, 206, 89, 253, 91, 66, 58, 15, 61, 141, 252, 207, 134, 327, 291, 302, 265, 167, 284, 45, 201, 197, 198, 199, 181, 180, 188, 73, 74, 179, 263, 200, 146, 164, 118, 202, 246, 254, 203, 163, 204, 117, 10, 171, 247, 260, 39, 107, 34, 159, 33, 43, 44, 31, 214, 216, 194, 142, 8, 32, 22, 292, 295, 280, 42, 35, 28, 25, 27, 40, 152, 276, 193, 294, 30, 63, 37, 26, 213, 41, 317, 38, 36, 9, 7, 215, 293, 29, 24, 23, 300, 6, 3, 5, 103, 4, 186, 46, 314, 68, 14, 1, 233, 251, 2, 169, 313].

16.13 Long-term archiving of research results

MPI-SWS has a policy of keeping all source data used for published research results archived through our normal system backup procedure. When this data is useful for other researchers' work, the data—and, where appropriate, the tools used to produce the data—are also made available on our website.

16.14 Appointments, scientific awards and memberships

- Nasri won a post-doctoral Humboldt Fellowship.
- In 2016, Cerqueira et al.'s work [69] on mechanized proofs for real-time systems was recognized with the ECRTS'16 Best Paper Award.
- Nasri won the RTNS'16 Best Paper Award for her work in collaboration with Mohaqeqi et al. [230] on the problem of finding optimal harmonic periods for real-time control tasks in 2016.
- In 2016, Brandenburg and Gül's work [62] on practical, empirically near-optimal multiprocessor real-time scheduling was recognized with the RTSS'16 Best Paper Award.
- In 2017, Nasri and Brandenburg's work on space-, overhead-, and schedulability-efficient non-preemptive scheduling [238] was recognized with an Outstanding-Paper Award at RTAS'17.
- Patel et al.'s paper on a mechanism for avoiding timer interference in real-time operating systems [256] was recognized with the RTAS'17 Best Paper Award in 2017. (The first author of the paper, Pratyush Patel, was an undergraduate research intern in the group from May until August 2016.)
- Also in 2017, Gujarati received the Best Student Paper Award at Middleware'17 for his paper on resource-efficient, distributed autoscaling for "machine learning as a service" (MLaaS) providers [161].

- Christakis has been presented with a Facebook Faculty Research Award for her research on combining static and dynamic program analysis, which also received other awards, including the EAPLS Best PhD Dissertation Award.
- Dreyer received the 2017 ACM SIGPLAN Robin Milner Young Researcher Award, the highest international accolade granted to mid-career researchers in the area of programming languages.
- Dreyer was granted the title of Honorarprofessor of Computer Science at Saarland University in 2017.
- Gummadi was also granted title of Honorarprofessor of Computer Science at Saarland University in 2017.
- The PLDI'17 paper of Lahav, Vafeiadis, Kang, Hur, and Dreyer received a Best Paper Award.
- The ECOOP'17 paper of Kaiser, Dang, Dreyer, Lahav, and Vafeiadis also received a Best Paper Award.
- Lahav, former postdoc of Vafeiadis, received a Best Paper Award at TABLEAUX 2017.
- Swasey, Garg, and Dreyer won a Distinguished Paper Award for their OOPSLA 2017 paper.
- For their CSF'15 paper, Rajani, Bichhawat, Garg, and Hammer won the 2016 Best Paper Award from the DFG priority program that funded their work on browser security.
- Speicher and Gummadi received a Best Paper Nomination at the Conference on Fairness, Accountability, and Transparency (FAT*) in 2018.
- Jourdan, former postdoc of Dreyer, received the 2016 Thesis Prize of the GDR GPL (French research group on programming and software engineering) for his PhD thesis, "Verasco: A Formally Verified C Static Analyzer".
- A paper Druschel co-authored with Gummadi, Bhattacharjee, and students received the SIGCOMM Test of Time Award in 2017.
- Goga received the Best Paper Runner-Up Award at the IEEE ASONAM 2017.

- Gomez-Rodriguez and Valera, together with Gummadi, Zafar, and Weller received a Best Paper Award Honorable Mention at WWW 2017.
- Viswanath, Zafar, and Gummadi received the Best Paper Award at the Conference on Online Social Networks (COSN) 2015.
- Grgic-Hlaca, Zafar, and Gummadi received a Notable Paper Award at the NIPS Symposium on ML and Law 2017.
- Pouly, postdoc of Ouaknine, received the prestigious Ackermann Award in 2017.
- Pouly won a Best Paper Award at the CMSB 2017 conference for a paper which he co-authored.
- Pouly also won a Best Paper Award at (ICALP 2016) [58] for a paper he co-authored.
- A paper by Chonev, Ouaknine, and Worrell (J. ACM 2016) [90] was listed as a Notable Article by ACM Computing Reviews 21st Annual Best of Computing for 2016.
- Druschel received the Microsoft Research Outstanding Collaborator Award in 2016 and the EuroSys Lifetime Achievement Award in 2017.

16.15 External funding

- Brandenburg received funding from ANR-DFG for "RT-Proofs: Formal Proofs for Real-Time Systems".
- Nasri is funded by a post-doctoral Humboldt Fellowship (July 2016–July 2018).
- The research of Christakis's group has been partially funded by a Facebook Faculty Research Award for research on combining static and dynamic program analysis.
- Darulova obtained a DFG grant titled "Automated Rigorous Verification and Synthesis of Approximations" in October 2017.
- Dreyer was awarded a 2015 ERC Consolidator Grant of €1.95M, for the project "RustBelt: Logical Foundations for the Future of Safe Systems Programming". The project runs from April 2016 to March 2021.

- Dreyer obtained Microsoft Research PhD scholorship funding from January 2014 to December 2016.

- Garg's group research has been partially funded by two grants from the German Science Foundation, DFG. The first one is on information flow control in web browsers. This covered one student and one intern from 2012 to 2016. The second one, on language support for information flow control, is part of the broader SFB in collaboration with Saarland University, and pays for one graduate student from 2016 to 2020, and two graduate students from 2017 to 2020.

- Postdoctoral scholar Valera was funded by a Humboldt postdoctoral scholarship.

- Gummadi's research of the group has been partially funded by DFG's Collaborative Research Center grant on Methods and Tools for Understanding and Controlling Privacy as well as industry grants from Data Transparency Lab and AT&T research.

- Ouaknine's group has been partially funded by ERC Consolidator Grant of €1,84M, for AVS-ISS: "Analysis, Verification, and Synthesis of Infinite-State Systems", August 2015 to August 2020.

- Druschel is a co-PI in Saarland University's MMCI Cluster of Excellence and the Saarbrücken Graduate School in Computer Science, funded by the German National Science Foundation (DFG, €45M, 2013–2019). He is also a co-PI in Saarland University's Collaborative Research Center on Methods and Tools for Understanding and Controlling Privacy, funded by the DFG (€8.5M, 2016–2020). From 2011-2017, he was co-PI and assistant director of the Center for Information Security, Privacy and Trust, funded by the German ministry of science (BMBF, €21M). Jointly with Majumdar, Backes (CISPA), and Weikum (MPI for Informatics), Druschel is a co-PIs on an ERC Synergy Grant on Privacy, Accountability, Compliance, and Trust in the Internet (€9.25M, 2015–2021).

- Vafeiadis group's research has been partially funded by the European Commission's FP7 FET young explorers grant ADVENT (April 2013 – April 2016). The grant partially funded postdoc Ori Lahav.

- Majumdar has been the recipient of a Toyota research contract since 2013.

16.16 Public relations work

MPI-SWS and Diffix (jointly-developed Diffix anonymization) frequently appear in marketing and press materials produced by Aircloak. It is to Aircloak's advantage to market its relationship with MPI-SWS, and to portray Diffix as the output of research done by MPI-SWS.

Articles describing Gummadi's social network research have appeared in numerous popular news media and technology blogs including the New York Times, Harvard Business Review, MIT Technology Review, New Scientist, Wired magazine, Slashdot, Businessweek, Sueddeutsche Zeitung (Germany), Science TV (Korea), and MTV (Brazil).

In addition, the Institute participated in the following public relations activities:

- In Kaiserslautern, efficient collaboration between all relevant academic institutions and strong innovative industrial partners is achieved through the "Science Alliance" umbrella organization.

- In June 2016, the MPI-SWS was among the host institutes of the Max Planck Society's Annual meeting.

- Together with the TU Kaiserslautern, the Institute participated in April 2016 in the event "Long Night of Sciences", where the university and its affiliated institutes, presented themselves to a wider public audience. This biannual event serves as a link between research and the region, introducing visitors to the entire spectrum of scientific knowledge from basic research to laboratory samples and prototypes. This year, April 2018 the MPI-SWS will particpate in this event again.

- In April 2016 and 2017 the MPI-SWS has taken part in the nationwide Girls' Day at the Saarland University. The idea is to raise awareness of careers in science and technology for girls from grade 8 on. Our institute offered a workshop consisting of short lectures with numerous illustrative examples, laboratory tours, and experiments. Mainly female scientists at our institute gave insights into various research areas of computer science and technology.

- A similar event takes place every year at the University of Kaiserslautern, namely the "Schülerinnentag" to raise awareness of careers in science and technology for girls in which we also have been taking part in, in the fall of 2016 and 2017. In 2018 will participate in both events again.

- The Institute appears in numerous publications with the goal of making the Institute and the local scientific landscape well-known to the broader public.

References

[1] *23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016*. The Internet Society, 2016.

[2] *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*. IEEE Computer Society, 2016.

[3] *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017*. IEEE, 2017.

[4] *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. IEEE Computer Society, 2017.

[5] *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017*. The Internet Society, 2017.

[6] *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*. IEEE Computer Society, 2017.

[7] Y. Acar, M. Backes, S. Bugiel, S. Fahl, P. D. McDaniel, and M. Smith. Sok: Lessons learned from android security research for appified software platforms. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016* [2], pages 433–451.

[8] Y. Acar, M. Backes, S. Fahl, S. L. Garfinkel, D. Kim, M. L. Mazurek, and C. Stransky. Comparing the usability of cryptographic apis. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017* [4], pages 154–171.

[9] Y. Acar, M. Backes, S. Fahl, D. Kim, M. L. Mazurek, and C. Stransky. You get where you're looking for: The impact of information sources on code security. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016* [2], pages 289–305.

[10] Y. Acar, M. Backes, S. Fahl, D. Kim, M. L. Mazurek, and C. Stransky. How internet resources might be helping you develop faster but less securely. *IEEE Security & Privacy*, 15(2):50–60, 2017.

[11] P. Aditya, R. Sen, P. Druschel, S. Joon Oh, R. Benenson, M. Fritz, B. Schiele, B. Bhattacharjee, and T. T. Wu. I-pic: A platform for privacy-compliant image capture. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '16, pages 235–248, New York, NY, USA, 2016. ACM.

[12] A. Aguirre, G. Barthe, L. Birkedal, A. Bizjak, M. Gaboardi, and D. Garg. Relational reasoning for markov chains in a probabilistic guarded lambda calculus. In *European Symposium on Programming (ESOP)*, 2018. To appear.

[13] A. Aguirre, G. Barthe, M. Gaboardi, D. Garg, and P. Strub. A relational logic for higher-order programs. *PACMPL*, 1(ICFP):21:1–21:29, 2017.

[14] A. Aldini, J. Lopez, and F. Martinelli, editors. *Foundations of Security Analysis and Design VIII - FOSAD 2014/2015/2016 Tutorial Lectures*, volume 9808 of *Lecture Notes in Computer Science*. Springer, 2016.

[15] S. Almagor, J. Ouaknine, and J. Worrell. The polytope-collision problem. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, pages 24:1–24:14, 2017.

[16] A. Andreou, O. Goga, and P. Loiseau. Identity vs. attribute disclosure risks for users with multiple social profiles. In *ASONAM*, pages 163–170. ACM, 2017.

[17] A. Andreou, G. Venkatadri, O. Goga, K. P. Gummadi, P. Loiseau, and A. Mislove. Privacy risks with facebook's pii-based targeting: An audit of a data broker's advertising interface. In *NDSS*. Internet Society, 2018.

[18] A. W. Appel. *Program Logics - for Certified Compilers*. Cambridge University Press, 2014.

[19] M. Appel, A. Gujarati, and B. Brandenburg. A byzantine fault-tolerant key-value store for safety-critical distributed real-time systems. In *Proceedings of the 2nd Workshop on Security and Dependability of Critical Embedded Real-Time Systems (CERTS 2017)*, 2017.

[20] M. F. Atig, D. Chistikov, P. Hofman, K. N. Kumar, P. Saivasan, and G. Zetzsche. The complexity of regular abstractions of one-counter languages. In *LICS*, pages 207–216. ACM, 2016.

[21] M. Babaei, P. A. Grabowicz, I. Valera, K. P. Gummadi, and M. Gomez-Rodriguez. On the efficiency of the information networks in social media. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016*, pages 83–92, 2016.

[22] M. Backes, P. Berrang, M. Bieg, R. Eils, C. Herrmann, M. Humbert, and I. Lehmann. Identifying personal DNA methylation profiles by genotype inference. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017* [4], pages 957–976.

[23] M. Backes, P. Berrang, O. Goga, K. P. Gummadi, and P. Manoharan. On profile linkability despite anonymity in social media systems. In *WPES@CCS*, pages 25–35. ACM, 2016.

[24] M. Backes, P. Berrang, A. Hecksteden, M. Humbert, A. Keller, and T. Meyer. Privacy in epigenetics: Temporal linkability of microrna expression profiles. In Holz and Savage [169], pages 1223–1240.

[25] M. Backes, P. Berrang, M. Humbert, and P. Manoharan. Membership privacy in microrna-based studies. In Weippl et al. [314], pages 319–330.

[26] M. Backes, P. Berrang, and P. Manoharan. From zoos to safaris - from closed-world enforcement to open-world assessment of privacy. In Aldini et al. [14], pages 87–138.

[27] M. Backes, S. Bugiel, and E. Derr. Reliable third-party library detection in android and its security applications. In Weippl et al. [314], pages 356–367.

[28] M. Backes, S. Bugiel, E. Derr, S. Gerling, and C. Hammer. R-droid: Leveraging android app analysis with static slice optimization. In Chen et al. [78], pages 129–140.

[29] M. Backes, S. Bugiel, E. Derr, P. D. McDaniel, D. Octeau, and S. Weisgerber. On demystifying the android application framework: Revisiting android permission specification analysis. In Holz and Savage [169], pages 1101–1118.

[30] M. Backes, S. Bugiel, J. Huang, and O. Schranz. POSTER: the ART of app compartmentalization. In Weippl et al. [314], pages 1811–1813.

[31] M. Backes, S. Bugiel, O. Schranz, P. von Styp-Rekowsky, and S. Weisgerber. Artist: The android runtime instrumentation and security toolkit. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017* [3], pages 481–495.

[32] M. Backes, S. Bugiel, P. von Styp-Rekowsky, and M. WiBfeld. Seamless in-app ad blocking on stock android. In *2017 IEEE Security and Privacy Workshops, SP Workshops 2017, San Jose, CA, USA, May 25, 2017*, pages 163–168. IEEE Computer Society, 2017.

[33] M. Backes, J. Dreier, S. Kremer, and R. Künnemann. A novel approach for reasoning about liveness in cryptographic protocols and its application to fair exchange. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017* [3], pages 76–91.

[34] M. Backes, M. Gomez-Rodriguez, P. Manoharan, and B. Surma. Reconciling privacy and utility in continuous-time diffusion networks. In *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*, pages 292–304, 2017.

[35] M. Backes, N. Grimm, and A. Kate. Data lineage in malicious environments. *IEEE Trans. Dependable Sec. Comput.*, 13(2):178–191, 2016.

[36] M. Backes, C. Hammer, D. Pfaff, and M. Skoruppa. Implementation-level analysis of the javascript helios voting client. In Ossowski [251], pages 2071–2078.

[37] M. Backes, A. Herzberg, A. Kate, and I. Pryvalov. Anonymous RAM. In I. G. Askoxylakis, S. Ioannidis, S. K. Katsikas, and C. A. Meadows, editors, *Computer Security - ESORICS 2016 - 21st European Symposium on Research in Computer Security, Heraklion, Greece, September 26-30, 2016, Proceedings, Part I*, volume 9878 of *Lecture Notes in Computer Science*, pages 344–362. Springer, 2016.

[38] M. Backes, T. Holz, C. Rossow, T. Rytilahti, M. Simeonovski, and B. Stock. On the feasibility of ttl-based filtering for drdos mitigation. In Monrose et al. [233], pages 303–322.

[39] M. Backes, M. Humbert, J. Pang, and Y. Zhang. walk2friends: Inferring social links from mobility profiles. In Thuraisingham et al. [300], pages 1943–1957.

[40] M. Backes, R. Künnemann, and E. Mohammadi. Computational soundness for dalvik bytecode. In Weippl et al. [314], pages 717–730.

[41] M. Backes, S. Meiser, and D. Schröder. Delegatable functional signatures. In C. Cheng, K. Chung, G. Persiano, and B. Yang, editors, *Public-Key Cryptography - PKC 2016 - 19th IACR International Conference on Practice and Theory in Public-Key Cryptography, Taipei, Taiwan, March 6-9, 2016, Proceedings, Part I*, volume 9614 of *Lecture Notes in Computer Science*, pages 357–386. Springer, 2016.

[42] M. Backes, S. Meiser, and M. Slowik. Your choice mator(s). *PoPETs*, 2016(2):40–60, 2016.

[43] M. Backes and M. Nauman. LUNA: quantifying and leveraging uncertainty in android malware analysis through bayesian machine learning. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017* [3], pages 204–217.

[44] M. Backes, K. Rieck, M. Skoruppa, B. Stock, and F. Yamaguchi. Efficient and flexible discovery of PHP application vulnerabilities. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017* [3], pages 334–349.

[45] V. Balegas, C. Li, M. Najafzadeh, D. Porto, A. Clement, S. Duarte, C. Ferreira, J. Gehrke, J. Leitão, N. M. Preguiça, R. Rodrigues, M. Shapiro, and V. Vafeiadis. Geo-replication: Fast if possible, consistent if necessary. *IEEE Data Eng. Bull.*, 39(1):81–92, 2016.

[46] R. Barrett, R. Cummings, E. Agichtein, and E. Gabrilovich, editors. *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*. ACM, 2017.

[47] E. Bartocci and R. Majumdar. Introduction to the special issue on runtime verification. *Formal Methods in System Design*, 51(1):1–4, 2017.

[48] P. C. Bell, J. Delvenne, R. M. Jungers, and V. D. Blondel. The continuous skolem-pisot problem. *Theor. Comput. Sci.*, 411(40-42):3625–3634, 2010.

[49] A. Bichhawat, V. Rajani, J. Jain, D. Garg, and C. Hammer. Webpol: Fine-grained information flow policies for web browsers. In *Computer Security - ESORICS 2017 - 22nd European Symposium on Research in Computer Security, Oslo, Norway, September 11-15, 2017, Proceedings, Part I*, pages 242–259, 2017.

[50] A. J. Biega, A. Ghazimatin, H. Ferhatosmanoglu, K. P. Gummadi, and G. Weikum. Learning to un-rank: Quantifying search exposure for users in online communities. In *CIKM*, pages 267–276. ACM, 2017.

[51] J. A. Biega, K. P. Gummadi, I. Mele, D. Milchevski, C. Tryfonopoulos, and G. Weikum. R-susceptibility: An ir-centric approach to assessing privacy risks for users in online communities. In *SIGIR*, pages 365–374. ACM, 2016.

[52] A. Biondi and B. Brandenburg. Lightweight real-time synchronization under P-EDF on symmetric and asymmetric multiprocessors. In *Proceedings of the 28th Euromicro Conference on Real-Time Systems (ECRTS'16)*, pages 39–49, 2016.

[53] A. Biondi, B. Brandenburg, and A. Wieder. A blocking bound for nested FIFO spin locks. In *Proceedings of the 37th IEEE Real-Time Systems Symposium (RTSS'16)*, pages 291–302, 2016.

[54] S. L. Blond, C. Gilbert, U. Upadhyay, M. Gomez-Rodriguez, and D. Choffnes. A broad view of the ecosystem of socially engineered exploit documents. In *NDSS '17: Proceedings of the Network and Distributed System Security Symposium*, 2017.

[55] R. Bodík and R. Majumdar, editors. *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*. ACM, 2016.

[56] I. Bolosteanu and D. Garg. Asymmetric secure multi-execution with declassification. In *Principles of Security and Trust - 5th International Conference, POST 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*, pages 24–45, 2016.

[57] V. Bonifaci, B. Brandenburg, G. D'Angelo, and A. Marchetti-Spaccamela. Multiprocessor real-time scheduling with hierarchical

processor affinities. In *Proceedings of the 28th Euromicro Conference on Real-Time Systems (ECRTS'16)*, pages 237–247, 2016.

[58] O. Bournez, D. S. Graça, and A. Pouly. Polynomial time corresponds to solutions of polynomial ordinary differential equations of polynomial length: The general purpose analog computer and computable analysis are two efficiently equivalent models of computations. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 109:1–109:15, 2016.

[59] O. Bournez, D. S. Graça, and A. Pouly. On the functions generated by the general purpose analog computer. *Inf. Comput.*, 257:34–57, 2017.

[60] O. Bournez, D. S. Graça, and A. Pouly. Polynomial time corresponds to solutions of polynomial ordinary differential equations of polynomial length. *J. ACM*, 64(6):38:1–38:76, 2017.

[61] O. Bournez and A. Pouly. A universal ordinary differential equation. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, pages 116:1–116:14, 2017.

[62] B. Brandenburg and M. Gül. Global scheduling not required: Simple, near-optimal multiprocessor real-time scheduling with semi-partitioned reservations. In *Proceedings of the 37th IEEE Real-Time Systems Symposium (RTSS'16)*, pages 99–110, 2016.

[63] M. Brengel, M. Backes, and C. Rossow. Detecting hardware-assisted virtualization. In Caballero et al. [68], pages 207–227.

[64] S. Brookes. A semantics for concurrent separation logic. *Theor. Comput. Sci.*, 375(1-3):227–270, 2007.

[65] S. Brookes and P. W. O'Hearn. Concurrent separation logic. *SIGLOG News*, 3(3):47–65, 2016.

[66] M. Bruna, R. Grigore, S. Kiefer, J. Ouaknine, and J. Worrell. Proving the herman-protocol conjecture. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 104:1–104:12, 2016.

[67] D. Bundala and J. Ouaknine. On parametric timed automata and one-counter machines. *Inf. Comput.*, 253:272–303, 2017.

[68] J. Caballero, U. Zurutuza, and R. J. Rodríguez, editors. *Detection of Intrusions and Malware, and Vulnerability Assessment - 13th International Conference, DIMVA 2016, San Sebastián, Spain, July 7-8, 2016, Proceedings*, volume 9721 of *Lecture Notes in Computer Science*. Springer, 2016.

[69] F. Cerqueira, F. Stutz, and B. Brandenburg. PROSA: A case for readable mechanized schedulability analysis. In *Proceedings of the 28th Euromicro Conference on Real-Time Systems (ECRTS'16)*, pages 273–284, 2016.

[70] A. Chakraborty, S. Ghosh, N. Ganguly, and K. P. Gummadi. Dissemination biases of social media channels: On the topical coverage of socially shared news. In *ICWSM*, pages 559–562. AAAI Press, 2016.

[71] A. Chakraborty, S. Ghosh, N. Ganguly, and K. P. Gummadi. Optimizing the recency-relevancy trade-off in online news recommendations. In *WWW*, pages 837–846. ACM, 2017.

[72] A. Chakraborty, J. Messias, F. Benevenuto, S. Ghosh, N. Ganguly, and K. P. Gummadi. Who makes trends? understanding demographic biases in crowdsourced recommendations. In *ICWSM*, pages 22–31. AAAI Press, 2017.

[73] S. Chakraborty and V. Vafeiadis. Validating optimizations of concurrent C/C++ programs. In B. Franke, Y. Wu, and F. Rastello, editors, *Proceedings of the 2016 International Symposium on Code Generation and Optimization, CGO 2016, Barcelona, Spain, March 12-18, 2016*, pages 216–226. ACM, 2016.

[74] S. Chakraborty and V. Vafeiadis. Formalizing the concurrency semantics of an LLVM fragment. In V. J. Reddi, A. Smith, and L. Tang, editors, *Proceedings of the 2017 International Symposium on Code Generation and Optimization, CGO 2017, Austin, TX, USA, February 4-8, 2017*, pages 100–110. ACM, 2017.

[75] K. Chatterjee and V. S. Prabhu. Quantitative temporal simulation and refinement distances for timed systems. *IEEE Trans. Automat. Contr.*, 60(9):2291–2306, 2015.

[76] J.-J. Chen and B. Brandenburg. A note on the period enforcer algorithm for self-suspending tasks. *Leibniz Transactions on Embedded Systems (LITES)*, 4(1), 2017.

[77] J.-J. Chen, G. Nelissen, W.-H. Huang, M. Yang, B. Brandenburg, K. Bletsas, C. Liu, P. Richard, F. Ridouard, N. Audsley, R. Rajkumar, and D. de Niz. Many suspensions, many problems: A review of self-suspending tasks in real-time systems. Technical Report 854, Department of Computer Science, TU Dortmund, 2017.

[78] X. Chen, X. Wang, and X. Huang, editors. *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2016, Xi'an, China, May 30 - June 3, 2016*. ACM, 2016.

[79] D. Chistikov, W. Czerwinski, P. Hofman, M. Pilipczuk, and M. Wehar. Shortest paths in one-counter systems. In *FoSSaCS*, volume 9634 of *Lecture Notes in Computer Science*, pages 462–478. Springer, 2016.

[80] D. Chistikov, R. Dimitrova, and R. Majumdar. Approximate counting in SMT and value estimation for probabilistic programs. *Acta Inf.*, 54(8):729–764, 2017.

[81] D. Chistikov and C. Haase. The taming of the semi-linear set. In *ICALP*, volume 55 of *LIPIcs*, pages 128:1–128:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[82] D. Chistikov, C. Haase, and S. Halfon. Context-free commutative grammars with integer counters and resets. *Theoretical Computer Science*, 2016.

[83] D. Chistikov, S. Iván, A. Lubiw, and J. Shallit. Fractional coverings, greedy coverings, and rectifier networks. In *STACS*, volume 66 of *LIPIcs*, pages 23:1–23:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[84] D. Chistikov, S. Kiefer, I. Marusic, M. Shirmohammadi, and J. Worrell. On restricted nonnegative matrix factorization. In *ICALP*, volume 55 of *LIPIcs*, pages 103:1–103:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[85] D. Chistikov, S. Kiefer, I. Marusic, M. Shirmohammadi, and J. Worrell. On rationality of nonnegative matrix factorization. In *SODA*, pages 1290–1305. SIAM, 2017.

[86] D. Chistikov, S. Kiefer, I. Marušić, M. Shirmohammadi, and J. Worrell. Nonnegative matrix factorization requires irrationality. *SIAM Journal on Applied Algebra and Geometry*, 1(1):285–307, 2017.

[87] D. Chistikov, R. Majumdar, and F. Niksic. Hitting families of schedules for asynchronous programs. In *CAV (2)*, volume 9780 of *Lecture Notes in Computer Science*, pages 157–176. Springer, 2016.

[88] D. Chistikov, P. Martyugin, and M. Shirmohammadi. Synchronizing automata over nested words. In *FoSSaCS*, volume 9634 of *Lecture Notes in Computer Science*, pages 252–268. Springer, 2016.

[89] V. Chonev, J. Ouaknine, and J. Worrell. On recurrent reachability for continuous linear dynamical systems. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*, pages 515–524, 2016.

[90] V. Chonev, J. Ouaknine, and J. Worrell. On the complexity of the orbit problem. *J. ACM*, 63(3):23:1–23:18, 2016.

[91] V. Chonev, J. Ouaknine, and J. Worrell. On the skolem problem for continuous linear dynamical systems. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 100:1–100:13, 2016.

[92] O. Chowdhury, D. Garg, L. Jia, and A. Datta. Equivalence-based security for querying encrypted databases: Theory and application to privacy policy audits. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015*, pages 1130–1143, 2015.

[93] M. Christakis and P. Godefroid. Proving memory safety of the ANI Windows image parser using compositional exhaustive testing. In *Verification, Model Checking, and Abstract Interpretation - 16th International Conference, VMCAI 2015, Mumbai, India, January 12-14, 2015. Proceedings*, pages 373–392, 2015.

[94] M. Christakis, K. R. M. Leino, P. Müller, and V. Wüstholz. Integrated environment for diagnosing verification errors. In *Tools and Algorithms for the Construction and Analysis of Systems - 22nd International Conference, TACAS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*, pages 424–441, 2016.

[95] M. Christakis, P. Müller, and V. Wüstholz. Collaborative verification and testing with explicit assumptions. In *FM 2012: Formal Methods*

- *18th International Symposium, Paris, France, August 27-31, 2012. Proceedings*, pages 132–146, 2012.

[96] M. Christakis, P. Müller, and V. Wüstholz. An experimental evaluation of deliberate unsoundness in a static program analyzer. In *Verification, Model Checking, and Abstract Interpretation - 16th International Conference, VMCAI 2015, Mumbai, India, January 12-14, 2015. Proceedings*, pages 336–354, 2015.

[97] M. Christakis, P. Müller, and V. Wüstholz. Guiding dynamic symbolic execution toward unverified program executions. In *Proceedings of the 38th International Conference on Software Engineering, ICSE 2016, Austin, TX, USA, May 14-22, 2016*, pages 144–155, 2016.

[98] E. Çiçek, G. Barthe, M. Gaboardi, D. Garg, and J. Hoffmann. Relational cost analysis. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017*, pages 316–329, 2017.

[99] E. Çiçek, Z. Paraskevopoulou, and D. Garg. A type theory for incremental computational complexity with control flow changes. In *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming, ICFP 2016, Nara, Japan, September 18-22, 2016*, pages 132–145, 2016.

[100] S. Conchon, A. Goel, S. Krstic, R. Majumdar, and M. Roux. Far-cubicle - A new reachability algorithm for cubicle. In *FMCAD*, pages 172–175. IEEE, 2017.

[101] B. Cook, A. Podelski, and A. Rybalchenko. Termination proofs for systems code. In *Proceedings of the ACM SIGPLAN 2006 Conference on Programming Language Design and Implementation, Ottawa, Ontario, Canada, June 11-14, 2006*, pages 415–426, 2006.

[102] C. Cremers, M. Horvat, J. Hoyland, S. Scott, and T. van der Merwe. A comprehensive symbolic analysis of TLS 1.3. In *CCS*. ACM, 2017.

[103] M. Dacier, M. Bailey, M. Polychronakis, and M. Antonakakis, editors. *Research in Attacks, Intrusions, and Defenses - 20th International Symposium, RAID 2017, Atlanta, GA, USA, September 18-20, 2017, Proceedings*, volume 10453 of *Lecture Notes in Computer Science*. Springer, 2017.

[104] E. Darulova and V. Kuncak. Towards a compiler for reals. *ACM Trans. Program. Lang. Syst.*, 39(2):8:1–8:28, 2017.

[105] A. De, I. Valera, N. Ganguly, S. Bhattacharya, and M. Gomez-Rodriguez. Learning and forecasting opinion dynamics in social networks. In *Advances in Neural Information Processing Systems*, 2016.

[106] D. Deininger, R. Dimitrova, and R. Majumdar. Symbolic model checking for factored probabilistic models. In *ATVA*, volume 9938 of *Lecture Notes in Computer Science*, pages 444–460, 2016.

[107] E. Derr, S. Bugiel, S. Fahl, Y. Acar, and M. Backes. Keep me updated: An empirical study of third-party library updatability on android. In Thuraisingham et al. [300], pages 2187–2200.

[108] J. Deshmukh, M. Horvat, X. Jin, R. Majumdar, and V. S. Prabhu. Testing cyber-physical systems through bayesian optimization. *ACM Transactions on Embedded Computing Systems*, 16(5s):1–18, 2017.

[109] J. V. Deshmukh, R. Majumdar, and V. S. Prabhu. Quantifying conformance using the skorokhod metric. In *CAV (2)*, volume 9207 of *Lecture Notes in Computer Science*, pages 234–250. Springer, 2015.

[110] J. V. Deshmukh, R. Majumdar, and V. S. Prabhu. Quantifying conformance using the skorokhod metric. *Formal Methods in System Design*, 50(2-3):168–206, 2017.

[111] D. Devriese, M. Patrignani, and F. Piessens. Parametricity versus the universal type. *PACMPL*, 2(POPL):38:1–38:23, 2018.

[112] D. Devriese, M. Patrignani, F. Piessens, and S. Keuchel. Modular, fully-abstract compilation by approximate back-translation. *Logical Methods in Computer Science*, 13(4), 2017.

[113] R. Dimitrova, L. M. F. Fioriti, H. Hermanns, and R. Majumdar. Probabilistic ctl*: The deductive way. In *TACAS*, volume 9636 of *Lecture Notes in Computer Science*, pages 280–296. Springer, 2016.

[114] R. Dimitrova, J. Fu, and U. Topcu. Robust optimal policies for markov decision processes with safety-threshold constraints. In *55th IEEE Conference on Decision and Control, CDC 2016, Las Vegas, NV, USA, December 12-14, 2016*, pages 7081–7086. IEEE, 2016.

[115] R. Dimitrova, I. Gavran, R. Majumdar, V. S. Prabhu, and S. E. Z. Soudjani. The robot routing problem for collecting aggregate stochastic rewards. In *CONCUR*, volume 85 of *LIPIcs*, pages 13:1–13:17. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[116] M. Dodds, S. Jagannathan, M. J. Parkinson, K. Svendsen, and L. Birkedal. Verifying custom synchronization constructs using higher-order separation logic. *ACM Trans. Program. Lang. Syst.*, 38(2):4:1–4:72, 2016.

[117] M. Doko and V. Vafeiadis. A program logic for C11 memory fences. In B. Jobstmann and K. R. M. Leino, editors, *Verification, Model Checking, and Abstract Interpretation - 17th International Conference, VMCAI 2016, St. Petersburg, FL, USA, January 17-19, 2016. Proceedings*, volume 9583 of *Lecture Notes in Computer Science*, pages 413–430. Springer, 2016.

[118] M. Doko and V. Vafeiadis. Tackling real-life relaxed concurrency with FSL++. In H. Yang, editor, *Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, volume 10201 of *Lecture Notes in Computer Science*, pages 448–475. Springer, 2017.

[119] C. Dragoi, T. A. Henzinger, and D. Zufferey. Psync: a partially synchronous language for fault-tolerant distributed algorithms. In R. Bodík and R. Majumdar, editors, *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*, pages 400–415. ACM, 2016.

[120] D. Dreyer and M. Sheeran. Special issue dedicated to ICFP 2014: Editorial. *J. Funct. Program.*, 26:e20, 2016.

[121] D. D'Souza, P. Ezudheen, P. Garg, P. Madhusudan, and D. Neider. Horn-ice learning for synthesizing invariants and contracts. *CoRR*, abs/1712.09418, 2017.

[122] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song. Recurrent Marked Temporal Point Process: Embedding Event History to Vector. In *Proceedings of the 22nd ACM SIGKDD*

International Conference on Knowledge Discovery in Data Mining, 2016.

[123] N. Du, Y. Liang, M.-F. Balcan, M. Gomez-Rodriguez, H. Zha, and L. Song. Scalable influence maximization for multiple products in continuous-time diffusion networks. *Journal of Machine Learning Research*, 2017.

[124] A. Durand-Gasselin, J. Esparza, P. Ganty, and R. Majumdar. Model checking parameterized asynchronous shared-memory systems. *Formal Methods in System Design*, 50(2-3):140–167, 2017.

[125] E. Elnikety, D. Garg, and P. Druschel. Shai: Enforcing Data-Specific Policies with Near-Zero Runtime Overhead. Submitted for publication.

[126] E. Elnikety, A. Mehta, A. Vahldiek-Oberwagner, D. Garg, and P. Druschel. Thoth: Comprehensive policy compliance in data retrieval systems. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 637–654, Austin, TX, 2016. USENIX Association.

[127] E. Elnikety, A. Mehta, A. Vahldiek-Oberwagner, D. Garg, and P. Druschel. Thoth: Comprehensive policy compliance in data retrieval systems. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, pages 637–654, 2016.

[128] K. P. G. Emilio Zagheni, Ingmar Weber. Leveraging facebook's advertising platform to monitor stocks of migrants. In *Population and Development Review*. Wiley-Blackwell, 2017.

[129] V. Erdélyi, T.-K. Le, B. Bhattacharjee, P. Druschel, and N. Ono. Sonoloc: Scalable positioning of commodity mobile devices. In *Proceedings of the 16th ACM International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, New York, NY, USA, 2018. ACM.

[130] J. Esparza, P. Ganty, J. Leroux, and R. Majumdar. Verification of population protocols. In *CONCUR*, volume 42 of *LIPIcs*, pages 470–482. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[131] J. Esparza, P. Ganty, J. Leroux, and R. Majumdar. Model checking population protocols. In *FSTTCS*, volume 65 of *LIPIcs*, pages 27:1–27:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[132] J. Esparza, P. Ganty, J. Leroux, and R. Majumdar. Verification of population protocols. *Acta Inf.*, 54(2):191–215, 2017.

[133] J. Esparza, P. Ganty, and R. Majumdar. Parameterized verification of asynchronous shared-memory systems. *J. ACM*, 63(1):10:1–10:48, 2016.

[134] F. Fages, G. L. Guludec, O. Bournez, and A. Pouly. Strong turing completeness of continuous chemical reaction networks and compilation of mixed analog-digital programs. In *Computational Methods in Systems Biology - 15th International Conference, CMSB 2017, Darmstadt, Germany, September 27-29, 2017, Proceedings*, pages 108–127, 2017.

[135] S. S. Farahani, R. Majumdar, V. S. Prabhu, and S. E. Z. Soudjani. Shrinking horizon model predictive control with chance-constrained signal temporal logic specifications. In *ACC*, pages 1740–1746. IEEE, 2017.

[136] M. Farajtabar, Y. Wang, M. Gomez-Rodriguez, S. Li, H. Zha, and L. Song. Coevolve: A joint point process model for information diffusion and network co-evolution. In *Advances in Neural Information Processing Systems*, 2015.

[137] M. Farajtabar, Y. Wang, M. Gomez-Rodriguez, S. Li, H. Zha, and L. Song. Coevolve: A joint point process model for information diffusion and network co-evolution. *Journal of Machine Learning Research*, 2017.

[138] B. Farinholt, M. Rezaeirad, P. Pearce, H. Dharmdasani, H. Yin, S. L. Blond, D. McCoy, and K. Levchenko. To catch a ratter: Monitoring the behavior of amateur darkcomet RAT operators in the wild. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 770–787, 2017.

[139] A. Farzan and Z. Kincaid. Strategy synthesis for linear arithmetic games. *PACMPL*, 2(POPL):61:1–61:30, 2018.

[140] M. Ferreira, M. B. Zafar, and K. P. Gummadi. The case for temporal transparency: Detecting policy change events in black-box decision making systems. *CoRR*, abs/1610.10064, 2016.

[141] N. Fijalkow, P. Ohlmann, J. Ouaknine, A. Pouly, and J. Worrell. Semi-algebraic invariant synthesis for the kannan-lipton orbit problem. In *34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany*, pages 29:1–29:13, 2017.

[142] F. Fischer, K. Böttinger, H. Xiao, C. Stransky, Y. Acar, M. Backes, and S. Fahl. Stack overflow considered harmful? the impact of copy&paste on android application security. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017* [4], pages 121–136.

[143] P. Francis, S. P. Eide, and R. Munz. Diffix: High-utility database anonymization. In *Privacy Technologies and Policy - 5th Annual Privacy Forum, APF 2017, Vienna, Austria, June 7-8, 2017, Revised Selected Papers*, pages 141–158, 2017.

[144] S. Gao and D. Zufferey. Interpolants in nonlinear theories over the reals. In M. Chechik and J. Raskin, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 22nd International Conference, TACAS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*, pages 625–641, 2016.

[145] I. Gavran, R. Majumdar, and I. Saha. Antlab: A multi-robot task server. *ACM Transactions on Embedded Computing Systems*, 16(5s), 2017.

[146] I. Gavran, F. Niksic, A. Kanade, R. Majumdar, and V. Vafeiadis. Rely/guarantee reasoning for asynchronous programs. In *CONCUR*, volume 42 of *LIPIcs*, pages 483–496. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[147] O. Goga, P. Loiseau, R. Sommer, R. Teixeira, and K. P. Gummadi. On the reliability of profile matching across large online social networks. In *KDD*, pages 1799–1808. ACM, 2015.

[148] O. Goga, G. Venkatadri, and K. P. Gummadi. The doppelgänger bot attack: Exploring identity impersonation in online social networks. In *Internet Measurement Conference*, pages 141–153. ACM, 2015.

[149] V. Golyanik, M. Nasri, and D. Stricker. Towards scheduling hard real-time image processing tasks on a single GPU. In *Proceedings of the*

2017 IEEE International Conference on Image Processing (ICIP'17), 2017.

[150] M. Gomez-Rodriguez, L. Song, H. Daneshmand, and B. Schoelkopf. Estimating diffusion networks: Recovery conditions, sample complexity & soft-thresholding algorithm. *Journal of Machine Learning Research*, 2016.

[151] M. Gomez-Rodriguez, L. Song, N. Du, H. Zha, and B. Schoelkopf. Influence estimation and maximization in continuous-time diffusion networks. *ACM Transactions on Information Systems*, 2016.

[152] J. Götzfried, T. Müller, G. Drescher, S. Nürnberger, and M. Backes. Ramcrypt: Kernel-based address space encryption for user-mode processes. In Chen et al. [78], pages 919–924.

[153] P. A. Grabowicz, M. Babaei, J. Kulshrestha, and I. Weber. The road to popularity: The dilution of growing audience on twitter. In *ICWSM*, pages 567–570. AAAI Press, 2016.

[154] P. A. Grabowicz, N. Ganguly, and K. P. Gummadi. Distinguishing between topical and non-topical information diffusion mechanisms in social media. In *ICWSM*, pages 151–160. AAAI Press, 2016.

[155] N. Grgic-Hlaca, E. Redmiles, K. P. Gummadi, and A. Weller. Human perceptions of fairness in algorithmic decision making: A case study of criminal risk prediction. In *WWW*. ACM, 2018.

[156] N. Grgic-Hlaca, M. B. Zafar, K. P. Gummadi, and A. Weller. The case for process fairness in learning: Feature selection for fair decision making. *NIPS Symposium on ML and Law*, 2017.

[157] N. Grgic-Hlaca, M. B. Zafar, K. P. Gummadi, and A. Weller. On fairness, diversity and randomness in algorithmic decision making. *CoRR*, abs/1706.10208, 2017.

[158] N. Grgic-Hlaca, M. B. Zafar, K. P. Gummadi, and A. Weller. Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In *AAAI Conference on Artifical Intelligence*. AAAI, 2018.

[159] K. Grosse, N. Papernot, P. Manoharan, M. Backes, and P. D. McDaniel. Adversarial examples for malware detection. In S. N. Foley,

D. Gollmann, and E. Snekkenes, editors, *Computer Security - ESORICS 2017 - 22nd European Symposium on Research in Computer Security, Oslo, Norway, September 11-15, 2017, Proceedings, Part II*, volume 10493 of *Lecture Notes in Computer Science*, pages 62–79. Springer, 2017.

[160] A. Gujarati and B. Brandenburg. When is CAN the weakest link? — A bound on failures-in-time in CAN-based real-time systems. In *Proceedings of the 36th IEEE Real-Time Systems Symposium (RTSS'15)*, pages 249–260, 2015.

[161] A. Gujarati, S. Elnikety, Y. He, K. McKinley, and B. Brandenburg. Swayam: Distributed autoscaling to meet SLAs of machine learning inference services with resource efficiency. In *Proceedings of the 18th International Middleware Conference (Middleware 2017)*, pages 109–120, 2017.

[162] A. Gujarati, M. Nasri, and B. Brandenburg. Lower-bounding the MTTF for systems with (m, k) constraints and IID iteration failure probabilities. In *Proceedings of the 2nd Workshop on Security and Dependability of Critical Embedded Real-Time Systems (CERTS 2017)*, 2017.

[163] M. He, V. Vafeiadis, S. Qin, and J. F. Ferreira. Reasoning about fences and relaxed atomics. In *24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, PDP 2016, Heraklion, Crete, Greece, February 17-19, 2016*, pages 520–527. IEEE Computer Society, 2016.

[164] N. Hemed, N. Rinetzky, and V. Vafeiadis. Modular verification of concurrency-aware linearizability. In Y. Moses, editor, *Distributed Computing - 29th International Symposium, DISC 2015, Tokyo, Japan, October 7-9, 2015, Proceedings*, volume 9363 of *Lecture Notes in Computer Science*, pages 371–387. Springer, 2015.

[165] A. Z. Henley, K. Muslu, M. Christakis, S. D. Fleming, and C. Bird. CFar: A tool to increase communication, productivity, and review quality in collaborative code reviews. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, Montréal, Canada, April 21-26, 2018*, 2018. To appear.

[166] R. Herbster, S. DellaTorre, P. Druschel, and B. Bhattacharjee. Privacy capsules: Preventing information leaks by mobile apps. In *Proceedings*

of the 14th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '16, pages 399–411, New York, NY, USA, 2016. ACM.

[167] C. Hirnschall, A. Singla, S. Tschiatschek, and A. Krause. Learning user preferences to incentivize exploration in the sharing economy. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, February 2-7, 2018, New Orleans, Louisiana, USA.*, 2018.

[168] J. Hoenicke, R. Majumdar, and A. Podelski. Thread modularity at many levels: a pearl in compositional verification. In *POPL*, pages 473–485. ACM, 2017.

[169] T. Holz and S. Savage, editors. *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016*. USENIX Association, 2016.

[170] K. Hsu, R. Majumdar, K. Mallik, and A.-K. Schmuck. Multi-layered abstraction-based controller synthesis for continuous-time systems. In *HSCC*. ACM, 2018.

[171] J. Huang, O. Schranz, S. Bugiel, and M. Backes. The ART of app compartmentalization: Compiler-based library privilege separation on stock android. In Thuraisingham et al. [300], pages 1037–1049.

[172] H. Izadi and Nasri. On scheduling sporadic non-preemptive tasks with arbitrary deadline using non-work-conserving scheduling. In *Junior Researcher Workshop on Real-Time Computing (JRWRTC)*, pages 9–12, 2016.

[173] A. Izycheva and E. Darulova. On sound relative error bounds for floating-point arithmetic. In *2017 Formal Methods in Computer Aided Design, FMCAD 2017, Vienna, Austria, October 2-6, 2017*, pages 15–22, 2017.

[174] M. Jankowiak and M. Gomez-Rodriguez. Uncovering the spatiotemporal patterns of collective social activity. In *SIAM International Conference on Data Mining*, 2017.

[175] R. Jung, J. Jourdan, R. Krebbers, and D. Dreyer. Rustbelt: securing the foundations of the Rust programming language. *PACMPL*, 2(POPL):66:1–66:34, 2018.

[176] R. Jung, R. Krebbers, L. Birkedal, and D. Dreyer. Higher-order ghost state. In *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming, ICFP 2016, Nara, Japan, September 18-22, 2016*, pages 256–269, 2016.

[177] R. Jung, R. Krebbers, J.-H. Jourdan, A. Bizjak, L. Birkedal, and D. Dreyer. Iris from the ground up: A modular foundation for higher-order concurrent separation logic. *JFP*, 2018.

[178] R. Jung, D. Swasey, F. Sieczkowski, K. Svendsen, A. Turon, L. Birkedal, and D. Dreyer. Iris: Monoids and invariants as an orthogonal basis for concurrent reasoning. In *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2015, Mumbai, India, January 15-17, 2015*, pages 637–650, 2015.

[179] J. Kaiser, H. Dang, D. Dreyer, O. Lahav, and V. Vafeiadis. Strong logic for weak memory: Reasoning about release-acquire consistency in Iris. In *31st European Conference on Object-Oriented Programming, ECOOP 2017, June 19-23, 2017, Barcelona, Spain*, pages 17:1–17:29, 2017.

[180] J. Kang, C. Hur, O. Lahav, V. Vafeiadis, and D. Dreyer. A promising semantics for relaxed-memory concurrency. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017*, pages 175–189, 2017.

[181] J. Kang, Y. Kim, C. Hur, D. Dreyer, and V. Vafeiadis. Lightweight verification of separate compilation. In *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*, pages 178–190, 2016.

[182] R. Kannan and R. J. Lipton. The orbit problem is decidable. In *Proceedings of the 12th Annual ACM Symposium on Theory of Computing, April 28-30, 1980, Los Angeles, California, USA*, pages 252–261, 1980.

[183] R. Kannan and R. J. Lipton. Polynomial-time algorithm for the orbit problem. *J. ACM*, 33(4):808–821, 1986.

[184] M. Karimi, E. Tavakoli, M. Farajtabar, L. Song, and M. Gomez-Rodriguez. Smart Broadcasting: Do you want to be seen? In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 2016.

[185] M. Karr. Affine relationships among variables of a program. *Acta Inf.*, 6:133–151, 1976.

[186] E. Kirda and T. Ristenpart, editors. *26th USENIX Security Symposium, USENIX Security 2017, Vancouver, BC, Canada, August 16-18, 2017*. USENIX Association, 2017.

[187] J. Kloos, R. Majumdar, and F. McCabe. Deferrability analysis for javascript. In *Haifa Verification Conference*, volume 10629 of *Lecture Notes in Computer Science*, pages 35–50. Springer, 2017.

[188] M. Kokologiannakis, O. Lahav, K. Sagonas, and V. Vafeiadis. Effective stateless model checking for C/C++ concurrency. *PACMPL*, 2(POPL):17:1–17:32, 2018.

[189] R. Krahn, B. Trach, A. Vahldiek-Oberwagner, T. Knauth, P. Bhatotia, and C. Fetzer. PESOS: policy-enhanced secure object store. In *Proceedings of the 13th European Conference on Computer Systems, EuroSys 2018, Porto, Portugal*, 2018.

[190] R. Krahn, B. Trach, A. Vahldiek-Oberwagner, T. Knauth, P. Bhatotia, and C. Fetzer. PESOS: Policy enhanced secure object store. In *EuroSys*, 2018. Conditionally accepted.

[191] R. Krebbers, R. Jung, A. Bizjak, J. Jourdan, D. Dreyer, and L. Birkedal. The essence of higher-order concurrent separation logic. In *Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, pages 696–723, 2017.

[192] R. Krebbers, A. Timany, and L. Birkedal. Interactive proofs in higher-order concurrent separation logic. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017*, pages 205–217, 2017.

[193] J. Krupp, M. Backes, and C. Rossow. Identifying the scan and attack infrastructures behind amplification ddos attacks. In Weippl et al. [314], pages 1426–1437.

[194] J. Krupp, M. Karami, C. Rossow, D. McCoy, and M. Backes. Linking amplification ddos attacks to booter services. In Dacier et al. [103], pages 427–449.

[195] J. Kulshrestha, M. Eslami, J. Messias, M. B. Zafar, S. Ghosh, K. P. Gummadi, and K. Karahalios. Quantifying search bias: Investigating sources of bias for political searches in social media. In *CSCW*, pages 417–432. ACM, 2017.

[196] G. Lafferriere, G. J. Pappas, and S. Yovine. Symbolic reachability computation for families of linear vector fields. *J. Symb. Comput.*, 32(3):231–253, 2001.

[197] O. Lahav. Semantic investigation of canonical gödel hypersequent systems. *J. Log. Comput.*, 26(1):337–360, 2016.

[198] O. Lahav and A. Avron. A cut-free calculus for second-order gödel logic. *Fuzzy Sets and Systems*, 276:1–30, 2015.

[199] O. Lahav, N. Giannarakis, and V. Vafeiadis. Taming release-acquire consistency. In R. Bodík and R. Majumdar, editors, *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*, pages 649–662. ACM, 2016.

[200] O. Lahav, J. Marcos, and Y. Zohar. It ain't necessarily so: Basic sequent systems for negative modalities. In L. D. Beklemishev, S. Demri, and A. Maté, editors, *Advances in Modal Logic 11, proceedings of the 11th conference on "Advances in Modal Logic," held in Budapest, Hungary, August 30 - September 2, 2016*, pages 449–468. College Publications, 2016.

[201] O. Lahav, J. Marcos, and Y. Zohar. Sequent systems for negative modalities. *Logica Universalis*, 11(3):345–382, 2017.

[202] O. Lahav and V. Vafeiadis. Explaining relaxed memory models with program transformations. In J. S. Fitzgerald, C. L. Heitmeyer, S. Gnesi, and A. Philippou, editors, *FM 2016: Formal Methods - 21st International Symposium, Limassol, Cyprus, November 9-11, 2016,*

Proceedings, volume 9995 of *Lecture Notes in Computer Science*, pages 479–495, 2016.

[203] O. Lahav, V. Vafeiadis, J. Kang, C. Hur, and D. Dreyer. Repairing sequential consistency in C/C++11. In *Proceedings of the 38th ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2017, Barcelona, Spain, June 18-23, 2017*, pages 618–632, 2017.

[204] O. Lahav and Y. Zohar. Cut-admissibility as a corollary of the subformula property. In R. A. Schmidt and C. Nalon, editors, *Automated Reasoning with Analytic Tableaux and Related Methods - 26th International Conference, TABLEAUX 2017, Brasília, Brazil, September 25-28, 2017, Proceedings*, volume 10501 of *Lecture Notes in Computer Science*, pages 65–80. Springer, 2017.

[205] A. Lavaei, S. E. Z. Soudjani, R. Majumdar, and M. Zamani. Compositional abstractions of interconnected discrete-time stochastic control systems. In *CDC*, pages 3551–3556. IEEE, 2017.

[206] R. Lazic, J. Ouaknine, and J. Worrell. Zeno, hercules, and the hydra: Safety metric temporal logic is ackermann-complete. *ACM Trans. Comput. Log.*, 17(3):16:1–16:27, 2016.

[207] A. Lechner, R. Mayr, J. Ouaknine, A. Pouly, and J. Worrell. Model checking flat freeze LTL on one-counter automata. In *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, pages 29:1–29:14, 2016.

[208] O. Lengál, A. W. Lin, R. Majumdar, and P. Rümmer. Fair termination for parameterized probabilistic concurrent systems. In *TACAS (1)*, volume 10205 of *Lecture Notes in Computer Science*, pages 499–517, 2017.

[209] M. Lentz, V. Erdélyi, P. Aditya, E. Shi, P. Druschel, and B. Bhattacharjee. Sddr: Light-weight, secure mobile encounters. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 925–940, San Diego, CA, Aug. 2014. USENIX Association.

[210] M. Lentz, R. Sen, P. Druschel, and B. Bhattacharjee. SeCloak: ARM Trustzone-based Mobile Peripheral Control. In *Proceedings of the 16th ACM International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, New York, NY, USA, 2018. ACM.

[211] J. Litton, A. Vahldiek-Oberwagner, E. Elnikety, D. Garg, B. Bhattacharjee, and P. Druschel. Light-weight contexts: An OS abstraction for safety and performance. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 49–64, Savannah, GA, 2016. USENIX Association.

[212] J. Litton, A. Vahldiek-Oberwagner, E. Elnikety, D. Garg, B. Bhattacharjee, and P. Druschel. Light-weight contexts: An OS abstraction for safety and performance. In *12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016, Savannah, GA, USA, November 2-4, 2016.*, pages 49–64, 2016.

[213] K. Lu, W. Lee, S. Nürnberger, and M. Backes. How to make ASLR win the clone wars: Runtime re-randomization. In *23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016* [1].

[214] K. Lu, M. Walter, D. Pfaff, S. Nümberger, W. Lee, and M. Backes. Unleashing use-before-initialization vulnerabilities in the linux kernel using targeted stack spraying. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017* [5].

[215] G. Maisuradze, M. Backes, and C. Rossow. What cannot be read, cannot be leveraged? revisiting assumptions of JIT-ROP defenses. In Holz and Savage [169], pages 139–156.

[216] G. Maisuradze, M. Backes, and C. Rossow. Dachshund: Digging for and securing (non-)blinded constants in JIT code. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017* [5].

[217] P. Maiya, R. Gupta, A. Kanade, and R. Majumdar. Partial order reduction for event-driven multi-threaded programs. In *TACAS*, volume 9636 of *Lecture Notes in Computer Science*, pages 680–697. Springer, 2016.

[218] R. Majumdar. Robots at the edge of the cloud. In *TACAS*, volume 9636 of *Lecture Notes in Computer Science*, pages 3–13. Springer, 2016.

[219] R. Majumdar and V. Kuncak, editors. *Computer Aided Verification - 29th International Conference, CAV 2017, Heidelberg, Germany, July*

24-28, 2017, Proceedings, Part I, volume 10426 of *Lecture Notes in Computer Science*. Springer, 2017.

[220] R. Majumdar and V. Kuncak, editors. *Computer Aided Verification - 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part II*, volume 10427 of *Lecture Notes in Computer Science*. Springer, 2017.

[221] R. Majumdar and F. Niksic. Why is random testing effective for partition tolerance bugs? *PACMPL*, 2(POPL):46:1–46:24, 2018.

[222] R. Majumdar and V. S. Prabhu. Computing the skorokhod distance between polygonal traces. In *HSCC*, pages 199–208. ACM, 2015.

[223] R. Majumdar and V. S. Prabhu. Computing distances between reach flowpipes. In *HSCC*, pages 267–276. ACM, 2016.

[224] K. Mallik and A.-K. Schmuck. Supervisory controller synthesis for decomposable deterministic context free specification languages. In *2016 13th International Workshop on Discrete Event Systems (WODES)*, pages 22–27, May 2016.

[225] K. Mallik, S. E. Z. Soudjani, A. Schmuck, and R. Majumdar. Compositional construction of finite state abstractions for stochastic control systems. In *CDC*, pages 550–557. IEEE, 2017.

[226] C. Mavroforakis, I. Valera, and M. Gomez-Rodriguez. Modeling the dynamics of online learning activity. In *Proceedings of the 26th International World Wide Web Conference*, 2017.

[227] A. Mehta, E. Elnikety, K. Harvey, D. Garg, and P. Druschel. Qapla: Policy compliance for database-backed systems. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 1463–1479, Vancouver, BC, 2017. USENIX Association.

[228] A. Mehta, E. Elnikety, K. Harvey, D. Garg, and P. Druschel. Qapla: Policy compliance for database-backed systems. In *26th USENIX Security Symposium, USENIX Security 2017, Vancouver, BC, Canada, August 16-18, 2017.*, pages 1463–1479, 2017.

[229] A. Mislove, M. Marcon, P. K. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Internet Measurement Conference*, pages 29–42. ACM, 2007.

[230] M. Mohaqeqi, M. Nasri, Y. Zu, A. Cervin, and K.-E. Arzen. On the problem of finding optimal harmonic periods. In *International Conference on Real-Time Networks and Systems (RTNS'16)*, pages 171–180. ACM, 2016.

[231] M. Mondal, J. Messias, S. Ghosh, K. P. Gummadi, and A. Kate. Forgetting in social media: Understanding and controlling longitudinal exposure of socially shared data. In *SOUPS*, pages 287–299. USENIX Association, 2016.

[232] M. Mondal, J. Messias, S. Ghosh, K. P. Gummadi, and A. Kate. Longitudinal privacy management in social media: The need for better controls. *IEEE Internet Computing*, 21(3):48–55, 2017.

[233] F. Monrose, M. Dacier, G. Blanc, and J. García-Alfaro, editors. *Research in Attacks, Intrusions, and Defenses - 19th International Symposium, RAID 2016, Paris, France, September 19-21, 2016, Proceedings*, volume 9854 of *Lecture Notes in Computer Science*. Springer, 2016.

[234] P. Moreno-Sanchez, M. B. Zafar, and A. Kate. Listening to whispers of ripple: Linking wallets and deanonymizing transactions in the ripple network. *PoPETs*, 2016(4):436–453, 2016.

[235] R. Munz, F. Eigner, M. Maffei, P. Francis, and D. Garg. Unitrax: Protecting data privacy with discoverable biases. In *Conference on Principles of Security and Trust (POST)*, 2018. To appear.

[236] M. Nasri. On flexible and robust parameter assignment for periodic real-time components. In *Proceedings of the 9th International Workshop on Compositional Theory and Technology for Real-Time Embedded Systems (CRTS'16)*, 2016.

[237] M. Nasri and B. Brandenburg. An exact and sustainable analysis of non-preemptive scheduling. In *Proceedings of the 38th IEEE Real-Time Systems Symposium (RTSS'17)*, pages 12–23, 2017.

[238] M. Nasri and B. Brandenburg. Offline equivalence: A non-preemptive scheduling technique for resource-constrained embedded real-time systems. In *Proceedings of the 23rd IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'17)*, pages 75–86, 2017.

[239] M. Nasri, R. Davis, and B. Brandenburg. Fifo with offsets: High schedulability with low overheads. In *Proceedings of the 24th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'17)*, 2018, to appear.

[240] M. Nasri and E. Grolleau. On the existence of a cyclic schedule for non-preemptive periodic tasks with release offset. In *International Real-Time Scheduling Open Problems Seminar (RTSOPS'17)*, pages 7–8, 2017.

[241] M. Nasri, M. Mohaqeqi, and G. Fohler. Quantifying the effect of period ratios on schedulability of rate monotonic. In *International Conference on Real-Time Networks and Systems (RTNS'16)*, pages 161–170. ACM, 2016.

[242] M. Nasri and G. Nelissen. Increasing fixed-priority schedulability using non-periodic load shapers. In *International Real-Time Scheduling Open Problems Seminar (RTSOPS'17)*, pages 13–14, 2017.

[243] D. Neider, P. Garg, P. Madhusudan, S. Saha, and D. Park. Invariant synthesis for incomplete verification engines. In *Tools and Algorithms for the Construction and Analysis of Systems - 24nd International Conference, TACAS 2018*, to appear.

[244] D. Neider, S. Saha, and P. Madhusudan. Compositional synthesis of piece-wise functions by learning classifiers. *Transactions on Computational Logic*, to appear.

[245] D. Neider, A. Weinert, and M. Zimmermann. Synthesizing optimally resilient controllers. *CoRR*, abs/1709.04854, 2017.

[246] G. Neis, C. Hur, J. Kaiser, C. McLaughlin, D. Dreyer, and V. Vafeiadis. Pilsner: A compositionally verified compiler for a higher-order imperative language. In *Proceedings of the 20th ACM SIGPLAN International Conference on Functional Programming, ICFP 2015, Vancouver, BC, Canada, September 1-3, 2015*, pages 166–178, 2015.

[247] D. Nguyen, D. Wermke, Y. Acar, M. Backes, C. Weir, and S. Fahl. A stitch in time: Supporting android developers in writingsecure code. In Thuraisingham et al. [300], pages 1065–1077.

[248] P. W. O'Hearn. Resources, concurrency, and local reasoning. *Theor. Comput. Sci.*, 375(1-3):271–307, 2007.

[249] O. Ohrimenko, F. Schuster, C. Fournet, A. Mehta, S. Nowozin, K. Vaswani, and M. Costa. Oblivious multi-party machine learning on trusted processors. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 619–636, Austin, TX, 2016. USENIX Association.

[250] O. Ohrimenko, F. Schuster, C. Fournet, A. Mehta, S. Nowozin, K. Vaswani, and M. Costa. Oblivious multi-party machine learning on trusted processors. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, pages 619–636, 2016.

[251] S. Ossowski, editor. *Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pisa, Italy, April 4-8, 2016*. ACM, 2016.

[252] J. Ouaknine, J. S. Pinto, and J. Worrell. On the polytope escape problem for continuous linear dynamical systems. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control, HSCC 2017, Pittsburgh, PA, USA, April 18-20, 2017*, pages 11–17, 2017.

[253] J. Ouaknine, A. Pouly, J. S. Pinto, and J. Worrell. Solvability of matrix-exponential equations. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*, pages 798–806, 2016.

[254] A. Panda, O. Lahav, K. J. Argyraki, M. Sagiv, and S. Shenker. Verifying reachability in networks with mutable datapaths. In A. Akella and J. Howell, editors, *14th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2017, Boston, MA, USA, March 27-29, 2017*, pages 699–718. USENIX Association, 2017.

[255] M. J. Parkinson. The next 700 separation logics - (invited paper). In *Verified Software: Theories, Tools, Experiments, Third International Conference, VSTTE 2010, Edinburgh, UK, August 16-19, 2010. Proceedings*, pages 169–182, 2010.

[256] P. Patel, M. Vanga, and B. Brandenburg. TimerShield: Protecting high-priority tasks from low-priority timer interference. In *Proceedings of the 23rd IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'17)*, pages 3–12, 2017.

[257] M. Patrignani, D. Devriese, and F. Piessens. On modular and fully-abstract compilation. In *IEEE 29th Computer Security Foundations*

Symposium, CSF 2016, Lisbon, Portugal, June 27 - July 1, 2016, pages 17–30, 2016.

[258] M. Patrignani and D. Garg. Secure compilation and hyperproperty preservation. In *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*, pages 392–404, 2017.

[259] P.-M. Pédrot and N. Tabareau. Failure is not an option: An exceptional type theory. In *ESOP*, 2018.

[260] G. Pellegrino, M. Johns, S. Koch, M. Backes, and C. Rossow. Deemon: Detecting CSRF with dynamic analysis and property graphs. In Thuraisingham et al. [300], pages 1757–1771.

[261] R. Perera, D. Garg, and J. Cheney. Causally consistent dynamic slicing. In *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, pages 18:1–18:15, 2016.

[262] R. Piskac, T. Wies, and D. Zufferey. Grasshopper - complete heap verification with mixed specifications. In E. Ábrahám and K. Havelund, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 20th International Conference, TACAS 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014. Proceedings*, volume 8413 of *Lecture Notes in Computer Science*, pages 124–139. Springer, 2014.

[263] A. Podkopaev, O. Lahav, and V. Vafeiadis. Promising compilation to armv8 POP. In P. Müller, editor, *31st European Conference on Object-Oriented Programming, ECOOP 2017, June 19-23, 2017, Barcelona, Spain*, volume 74 of *LIPIcs*, pages 22:1–22:28. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[264] J. Proskurnia, P. A. Grabowicz, R. Kobayashi, C. Castillo, P. Cudré-Mauroux, and K. Aberer. Predicting the success of online petitions leveraging multidimensional time-series. In *WWW*, pages 755–764. ACM, 2017.

[265] G. Radanovic, A. Singla, A. Krause, and B. Faltings. Information gathering with peers: Submodular optimization with peer-prediction constraints. In *Proceedings of the Thirty-Second AAAI Conference on*

Artificial Intelligence, February 2-7, 2018, New Orleans, Louisiana, USA., 2018.

[266] A. Radhakrishna, N. Lewchenko, S. Meier, S. Mover, K. C. Sripada, D. Zufferey, B. E. Chang, and P. Cerný. Droidstar: Callback typestates for android classes. In M. Chechik and H. Mark, editors, *To apprear In Proceedings of the 40th International Conference on Software Engineering (ICSE 18)*. ACM, 2018.

[267] I. Radicek, G. Barthe, M. Gaboardi, D. Garg, and F. Zuleger. Monadic refinements for relational cost analysis. *PACMPL*, 2(POPL):36:1–36:32, 2018.

[268] W. Rafnsson, D. Garg, and A. Sabelfeld. Progress-sensitive security for SPARK. In *Engineering Secure Software and Systems - 8th International Symposium, ESSoS 2016, London, UK, April 6-8, 2016. Proceedings*, pages 20–37, 2016.

[269] W. Rafnsson, L. Jia, and L. Bauer. Timing-sensitive noninterference through composition. In *Principles of Security and Trust - 6th International Conference, POST 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, pages 3–25, 2017.

[270] V. Rajani, D. Garg, and T. Rezk. On access control, capabilities, their equivalence, and confused deputy attacks. In *IEEE 29th Computer Security Foundations Symposium, CSF 2016, Lisbon, Portugal, June 27 - July 1, 2016*, pages 150–163, 2016.

[271] Y. Ren, G. Liu, G. Parmer, and B. Brandenburg. Scalable memory reclamation for multi-core, real-time systems. In *Proceedings of the 24th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'17)*, 2018, to appear.

[272] J. C. Reynolds. Separation logic: A logic for shared mutable data structures. In *17th IEEE Symposium on Logic in Computer Science (LICS 2002), 22-25 July 2002, Copenhagen, Denmark, Proceedings*, pages 55–74, 2002.

[273] A. Schmuck, R. Majumdar, and A. Leva. Dynamic hierarchical reactive controller synthesis. *Discrete Event Dynamic Systems*, 27(2):261–299, 2017.

[274] A.-K. Schmuck, S. Schneider, J. Raisch, and U. Nestmann. Supervisory control synthesis for deterministic context free specification languages. *Discrete Event Dynamic Systems*, 26(1):5–32, Mar 2016.

[275] A.-K. Schmuck, P. Tabuada, and J. Raisch. Comparing asynchronous l-complete approximations and quotient based abstractions. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 6823–6829, Dec 2015.

[276] J. Schneider, N. Fleischhacker, D. Schröder, and M. Backes. Efficient cryptographic password hardening services from partially oblivious commitments. In Weippl et al. [314], pages 1192–1203.

[277] R. Sen, S. Ahmad, A. Phokeer, Z. A. Farooq, I. A. Qazi, D. R. Choffnes, and K. P. Gummadi. Inside the walled garden: Deconstructing facebook's free basics program. *Computer Communication Review*, 47(5):12–24, 2017.

[278] R. Sen, H. A. Pirzada, A. Phokeer, Z. A. Farooq, S. Sengupta, D. R. Choffnes, and K. P. Gummadi. On the free bridge across the digital divide: Assessing the quality of facebook's free basics service. In *Internet Measurement Conference*, pages 127–133. ACM, 2016.

[279] R. Sen, D. Quercia, C. K. V. Ruiz, and K. P. Gummadi. Scalable urban data collection from the web. In *ICWSM*, pages 683–686. AAAI Press, 2016.

[280] M. Simeonovski, G. Pellegrino, C. Rossow, and M. Backes. Who controls the internet?: Analyzing global threats using property graph traversals. In Barrett et al. [46], pages 647–656.

[281] S. Singh, V. Nanda, R. Sen, S. Sengupta, P. Kumaraguru, and K. P. Gummadi. Leveraging facebook's free basics engine for web service deployment in developing regions. In *ICTD*, pages 7:1–7:11. ACM, 2017.

[282] A. Singla, I. Bogunovic, G. Bartók, A. Karbasi, and A. Krause. On actively teaching the crowd to classify. In *NIPS Workshop on Data Driven Education*, 2013.

[283] A. Singla, I. Bogunovic, G. Bartók, A. Karbasi, and A. Krause. Near-optimally teaching the crowd to classify. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 154–162, 2014.

[284] A. Singla, H. Hasani, and A. Krause. Learning to interact with learning agents. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, February 2-7, 2018, New Orleans, Louisiana, USA.*, 2018.

[285] S. E. Z. Soudjani, A. Abate, and R. Majumdar. Dynamic bayesian networks as formal abstractions of structured stochastic processes. In *CONCUR*, volume 42 of *LIPIcs*, pages 169–183. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[286] S. E. Z. Soudjani, A. Abate, and R. Majumdar. Dynamic bayesian networks for formal verification of structured stochastic processes. *Acta Inf.*, 54(2):217–242, 2017.

[287] S. E. Z. Soudjani and R. Majumdar. Controller synthesis for reward collecting markov processes in continuous space. In *HSCC*, pages 45–54. ACM, 2017.

[288] S. E. Z. Soudjani, R. Majumdar, and A. Abate. Safety verification of continuous-space pure jump markov processes. In *TACAS*, volume 9636 of *Lecture Notes in Computer Science*, pages 147–163. Springer, 2016.

[289] S. E. Z. Soudjani, R. Majumdar, and T. Nagapetyan. Multilevel monte carlo method for statistical model checking of hybrid systems. In *QEST*, volume 10503 of *Lecture Notes in Computer Science*, pages 351–367. Springer, 2017.

[290] T. Speicher, M. Ali, G. Venkatadri, G. Arvanitakis, K. P. Gummadi, P. Loiseau, and A. Mislove. On the potential for discrimination in online targeted advertising. In *Conference on Fairness, Accountability, and Transparency*. FAT*, 2018.

[291] T. Speicher, M. B. Zafar, K. P. Gummadi, A. Singla, and A. Weller. Reliable learning by subsuming a trusted model: Safe exploration of the space of complex models. *ICML'17 Workshop on Reliable Machine Learning in the Wild*, 2017.

[292] B. Stock, M. Johns, M. Steffens, and M. Backes. How the web tangled itself: Uncovering the history of client-side web (in)security. In Kirda and Ristenpart [186], pages 971–987.

[293] B. Stock, G. Pellegrino, C. Rossow, M. Johns, and M. Backes. Hey, you have a problem: On the feasibility of large-scale web vulnerability notification. In Holz and Savage [169], pages 1015–1032.

[294] B. Stock, G. Pellegrino, C. Rossow, M. Johns, and M. Backes. POSTER: mapping the landscape of large-scale vulnerability notifications. In Weippl et al. [314], pages 1787–1789.

[295] C. Stransky, Y. Acar, D. Nguyen, D. Wermke, D. Kim, E. M. Redmiles, M. Backes, S. L. Garfinkel, M. L. Mazurek, and S. Fahl. Lessons learned from using an online platform to conduct large-scale, online controlled security experiments with software developers. In J. M. Fernandez and M. Payer, editors, *10th USENIX Workshop on Cyber Security Experimentation and Test, CSET 2017, Vancouver, BC, Canada, August 14, 2017*. USENIX Association, 2017.

[296] D. Swasey, D. Garg, and D. Dreyer. Robust and compositional verification of object capability patterns. *PACMPL*, 1(OOPSLA):89:1–89:26, 2017.

[297] B. Tabibian, I. Valera, M. Farajtabar, L. Song, B. Schoelkopf, and M. Gomez-Rodriguez. Distilling information reliability and source trustworthiness from digital traces. In *26th International World Wide Web Conference*, 2017.

[298] T. Tao. *Structure and randomness: pages from year one of a mathematical blog*. American Mathematical Society, 2008.

[299] J. Tassarotti, R. Jung, and R. Harper. A higher-order logic for concurrent termination-preserving refinement. In *Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, pages 909–936, 2017.

[300] B. M. Thuraisingham, D. Evans, T. Malkin, and D. Xu, editors. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*. ACM, 2017.

[301] L. Tsai, R. De Viti, M. Lentz, S. Saroiu, B. Bhattacharjee, and P. Druschel. Encounter-based Communication (EbC). Submitted for publication.

[302] S. Tschiatschek, A. Singla, M. G. Rodriguez, A. Merchant, and A. Krause. Detecting fake news in social networks via crowdsourcing. *CoRR*, abs/1711.09025, 2018.

[303] A. Turon, V. Vafeiadis, and D. Dreyer. GPS: navigating weak memory with ghosts, protocols, and separation. In *Proceedings of the 2014 ACM International Conference on Object Oriented Programming Systems Languages & Applications, OOPSLA 2014, part of SPLASH 2014, Portland, OR, USA, October 20-24, 2014*, pages 691–707, 2014.

[304] U. Upadhyay, I. Valera, and M. Gomez-Rodriguez. Uncovering the dynamics of crowdlearning and the value of knowledge. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, 2017.

[305] V. Vafeiadis and C. Narayan. Relaxed separation logic: a program logic for C11 concurrency. In *Proceedings of the 2013 ACM SIGPLAN International Conference on Object Oriented Programming Systems Languages & Applications, OOPSLA 2013, part of SPLASH 2013, Indianapolis, IN, USA, October 26-31, 2013*, pages 867–884, 2013.

[306] A. Vahldiek-Oberwagner, E. Elnikety, N. Duarte, D. Garg, and P. Druschel. ERIM: Secure and Efficient In-process Isolation with Memory Protection Keys. Submitted for publication.

[307] A. Vahldiek-Oberwagner, E. Elnikety, A. Mehta, D. Garg, P. Druschel, R. Rodrigues, J. Gehrke, and A. Post. Guardat: enforcing data policies at the storage layer. In *Proceedings of the Tenth European Conference on Computer Systems, EuroSys 2015, Bordeaux, France, April 21-24, 2015*, page 13, 2015.

[308] I. Valera and M. Gomez-Rodriguez. Modeling adoption and usage of competing products. In *IEEE International Conference on Data Mining*, 2015.

[309] M. Vanga, A. Bastoni, H. Theiling, and B. Brandenburg. Supporting low-latency, low-criticality tasks in a certified mixed-criticality OS. In *Proceedings of the 25th International Conference on Real-Time Networks and Systems (RTNS 2017)*, pages 226–235, 2017.

[310] G. Venkatadri, A. Andreou, O. Goga, K. P. Gummadi, P. Loiseau, and A. Mislove. Privacy risks with facebook's pii-based targeting: An

audit of a data broker's advertising interface. In *IEEE Symposium on Security and Privacy*. IEEE, 2018.

[311] G. Venkatadri, O. Goga, C. Zhong, B. Viswanath, K. P. Gummadi, and N. R. Sastry. Strengthening weak identities through inter-domain trust transfer. In *WWW*, pages 1249–1259. ACM, 2016.

[312] B. Viswanath, M. A. Bashir, M. B. Zafar, S. Bouget, S. Guha, K. P. Gummadi, A. Kate, and A. Mislove. Strength in numbers: Robust tamper detection in crowd computations. In *COSN*, pages 113–124. ACM, 2015.

[313] E. R. Weippl, S. Katzenbeisser, and S. D. C. di Vimercati, editors. *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society, WPES@CCS 2016, Vienna, Austria, October 24 - 28, 2016*. ACM, 2016.

[314] E. R. Weippl, S. Katzenbeisser, C. Kruegel, A. C. Myers, and S. Halevi, editors. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*. ACM, 2016.

[315] A. K. Wright and M. Felleisen. A syntactic approach to type soundness. *Inf. Comput.*, 115(1):38–94, 1994.

[316] M. Yang, A. Wieder, and B. Brandenburg. Global real-time semaphore protocols: A survey, unified analysis, and comparison. In *Proceedings of the 36th IEEE Real-Time Systems Symposium (RTSS'15)*, pages 1–12, 2015.

[317] A. Yokoyama, K. Ishii, R. Tanabe, Y. Papa, K. Yoshioka, T. Matsumoto, T. Kasama, D. Inoue, M. Brengel, M. Backes, and C. Rossow. Sandprint: Fingerprinting malware sandboxes to provide intelligence for sandbox evasion. In Monrose et al. [233], pages 165–187.

[318] M. B. Zafar, P. Bhattacharya, N. Ganguly, S. Ghosh, and K. P. Gummadi. On the wisdom of experts vs. crowds: Discovering trustworthy topical news in microblogs. In *CSCW*, pages 437–450. ACM, 2016.

[319] M. B. Zafar, P. Bhattacharya, N. Ganguly, K. P. Gummadi, and S. Ghosh. Sampling content from online social networks: Comparing random vs. expert sampling of the twitter stream. *TWEB*, 9(3):12:1–12:33, 2015.

[320] M. B. Zafar, K. P. Gummadi, and C. Danescu-Niculescu-Mizil. Message impartiality in social media discussions. In *ICWSM*, pages 466–475. AAAI Press, 2016.

[321] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 1171–1180, 2017.

[322] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, pages 962–970, 2017.

[323] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, K. P. Gummadi, and A. Weller. From parity to preference-based notions of fairness in classification. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 228–238, 2017.

[324] A. Zarezade, A. De, H. Rabiee, and M. Gomez-Rodriguez. Cheshire: An online algorithm for activity maximization in social networks. In *55th Annual Allerton Conference on Communication, Control, and Computing*, 2017.

[325] A. Zarezade, A. De, U. Upadhyay, H. Rabiee, and M. Gomez-Rodriguez. Steering social activity: A stochastic optimal control point of view. *Journal of Machine Learning Research*, 2018.

[326] A. Zarezade, U. Upadhyay, H. Rabiee, and M. Gomez-Rodriguez. Redqueen: An online algorithm for smart broadcasting in social networks. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, 2017.

[327] X. Zhu, A. Singla, S. Zilles, and A. N. Rafferty. An overview of machine teaching. *CoRR*, abs/1801.05927, 2018.

[328] B. Ziliani, D. Dreyer, N. R. Krishnaswami, A. Nanevski, and V. Vafeiadis. Mtac: A monad for typed tactic programming in Coq. *J. Funct. Program.*, 25, 2015.