CHUNGHA SUNG

sch8906[AT]gmail[DOT]com
https://chunghasung.org
Los Angeles, CA, USA

EDUCATION

University of Southern California (USC) *Aug.2017 - May.2021 (expected)*
Ph.D. Candidate, Computer Science

Virginia Polytechnic Institute and State University (Virginia Tech) *Aug.2014 - Dec.2016*
Master of Science, Electrical and Compute Engineering

Sung Kyun Kwan University (SKKU) *Mar.2007 - Aug.2013*
Bachelor of Science in Engineering, Semiconductor Systems Engineering

EXPERIENCE

Microsoft Research, USA *May.2019 - Aug.2019*
Research Intern

· Proposed an automatic merge-conflict resolution framework based on AST-based diffing and patching in divergent-forks for the Microsoft Edge development process (published in ICSE-SEIP 2020).

MediaTek Inc., USA *May.2018 - Aug.2018*
Research Intern

· Designed an SMT-based verifier for inconsistent constraints in instruction set architecture.

Microsoft Research, India *May.2017 - Aug.2017*
Research Intern

· Achieved an extension of a Scope query optimization tool to cover various types of queries (e.g., uninterpreted function) by static analysis on queries (F#, C#, Scope query).

Google Summer of Code *May.2013 - Sep.2013*
Participant

· Contributed to publishing an interactive graphical open-source package named “RIGHT” in R project

Ahn Lab Inc., Korea *Jul.2012 - Aug.2012*
Software Development Intern

· Advanced server maintenance systems by modifying a boot loader sequence in Linux-based server OS to support multiple layers of firmware.

PUBLICATIONS

[ICSE 2021] Data-Driven Synthesis of Provably Sound Side Channel Analyses
Jingbo Wang, **Chungha Sung**, Mukund Raghothaman and Chao Wang
Proceedings of the ACM/IEEE 43rd International Conference on Software Engineering (To appear).

[ICSE-SEIP 2020] Towards Understanding and Fixing Upstream Merge Induced Conflicts in Divergent Forks: An Industrial Case Study
[Nominated as a best paper candidate]
Chungha Sung, Shuvendu Lahiri, Mike Kaufman, Pallavi Choudhury and Chao Wang
Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering: Software Engineering in Practice, pages 172-181.

[ASE 2019] Debreach: Mitigating Compression Side Channels via Static Analysis and Transformation
Brandon Paulsen, **Chungha Sung**, Peter A.H. Peterson and Chao Wang
Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering, pages 899-911.

[FSE 2019] Mitigating Power Side Channels during Compilation
Jingbo Wang, **Chungha Sung** and Chao Wang
Proceedings of the 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, pages 590-601.

[ASE 2018] Datalog-based Scalable Semantic Diffing of Concurrent Programs
Chungha Sung, Shuvendu Lahiri, Constantin Enea and Chao Wang
Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering, pages 656-666.

[ASE 2018] CANAL: A Cache Timing Analysis Framework via LLVM Transformation
Chungha Sung, Brandon Paulsen and Chao Wang
Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering, pages 904-907.

[ASE 2017] Modular Verification of Interrupt-driven Software
Chungha Sung, Markus Kusano, and Chao Wang
Proceedings of the 32nd IEEE/ACM International Conference on Automated Software Engineering, pages 206-216.

[FSE 2016] Static DOM Event Dependency Analysis for Testing Web Applications
Chungha Sung, Markus Kusano, Nishant Sinha and Chao Wang
Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, pages 447-459.

POSTERS

[ICSE 2020] Towards Understanding and Fixing Upstream Merge Induced Conflicts in Divergent Forks: An Industrial Case Study
Chungha Sung, Shuvendu Lahiri, Mike Kaufman, Pallavi Choudhury, Jessica Wolk and Chao Wang
IEEE/ACM 42nd International Conference on Software Engineering, Seoul, Korea, July 2020.

[UseR 2014] RIGHT: an HTML canvas and JavaScript-based interactive data visualization package for linked graphics
ChungHa Sung, TaeJoon Song, Jae W. Lee and Junghoon Lee
The R User Conference, UCLA, Los Angeles, California, July 2014

PROJECTS

Fast and approximate semantic diffing of concurrent programs

- Proposed a polynomial-time analysis to compute synchronization differences in multi-threaded programs based on declarative analysis framework.
- Implemented the analysis with Datalog in an LLVM front-end pass.
- Achieved a high accuracy and 10x to 1000x faster analysis than a model checking based approach.
- Participated as the first author and the work is published at ASE 2018.

A unified cache timing analysis framework

- Proposed a unified modeling framework of cache analysis for various verification tools such as model checkers, abstract interpretation tools and symbolic execution tools.

- Implemented a C/C++ code transformation in an LLVM front-end pass.
- Showed the usefulness of the unified framework with KLEE, SMACK and Crab-llvm.
- Participated as the first author and the work is published at ASE 2018.

Accurate modular abstract interpretation for interrupt-driven software

- Proposed a data flow analysis between interrupts by a constraint-based program analysis framework.
- Implemented the analysis with Datalog in an LLVM front-end pass.
- Applied the analysis to prune infeasible data flows in thread modular abstract interpretation.
- Achieved the high accuracy of the abstract interpretation by reducing false positives.
- Participated as the first author and the work is published at ASE 2017.

Optimizing web application testing by static DOM-event dependency analysis

- Proposed a constraint-based static dependency analysis on DOM in web applications.
- Integrated the analysis results into a systematic test case generation tool to prune redundant test cases by partial order reduction (POR).
- Implemented the analysis with Datalog and JavaScript.
- Achieved 20 % higher code coverage in the testing tool with the analysis-based reduction.
- Participated as the first author and the work is published at FSE 2016.

Compression side channel analysis and mitigation in web applications

- Proposed an automated approach to detect and mitigate compression side channels in web server applications.
- Implemented a taint analysis to collect sensitive data sinks and enhanced compressor to mitigate the side channel leaks.
- Participated as a collaborator and the work is published at ASE 2019.

Power channel analysis and mitigation

- Proposed a constraint-based analysis framework to detect power side channel leaks due to register reuse in compiler.
- Designed a mitigation of power side channel leaks in compiler and implemented it in LLVM.
- Achieved an efficient power side channel mitigation in terms of runtime and transformed code size.
- Participated as a collaborator and the work is published at FSE 2019.

TEACHING

Teaching Assistant

- Software Engineering (CSCI 310, USC) *Fall.2019, Spring.2021*
- Data Structures and Object Oriented Design (CSCI 104, USC) *Spring.2019*
- Microcontroller Programming and Interfacing (ECE 2534, Virginia Tech) *Fall.2014*
- Digital Systems (SKKU) *Spring.2013*
- Digital Logic Design Laboratory (SKKU) *Spring.2012*

SERVICES

Student Helper for Organization

- CAV 2020, ATVA 2018

(External) Reviewer

· FMCAD 2020, ASE 2020, IET Software 2020, ICSE 2020, FSE 2019, ICSE 2019, SAS 2018, TSE 2018, ICSE 2017, FORM 2017, FMCAD 2017, RV 2017, ISSTA 2017, ICSE 2016, SETTA 2016, TurstSoft 2016, FMCAD 2016, ATVA 2016, ICECCS 2016, TASE 2016

AWARDS

LG Electronics industrial scholarship recipient	*May.2017*
ACM SIGSOFT travel grant for FSE	*Nov.2016*
Travel award for CAV	*July.2016*
First prize for graduation thesis and project award in SKKU	*June.2013*
Full Merit-based Awards from SKKU for Full Academic Years	*Mar.2007 - Aug.2013*
Dean's List award in SKKU	*Apr.2012*
Scholarship from Samsung Electronics	*Mar.2007 - Feb.2009*

KEYWORDS

Program Analysis, Program Verification, Program Languages, Software Testing, Software Verification, Software Reliability, Software Maintenance, Automatic Repair, Concurrent programs, Multi-threaded programs, Web applications, embedded systems, C/C++, Python, Java, JavaScript, C#, F#, LLVM, Clang, Compiler front-end, SMT solver, Datalog, Side Channel