Sung Kook Kim

Email: sklkim@ucdavis.edu
Website: https://skkeem.github.io

Education

Ph.D. Student, Computer Science, University of California at Davis (UC Davis) 2018-current

- Davis Automated Reasoning Group — Advisor: Prof. Aditya V. Thakur

M.S., Electrical and Computer Engineering, Seoul National University (SNU) 2015-2017

- Virtual Machine and Optimization Lab — Advisor: Prof. Soo-Mook Moon
- Thesis: Recycling Optimized Machine Codes Generated by JavaScript Engine

B.S., Computer Science (minor in Physics), Korea Advanced Institute of Science and Technology (KAIST) 2010-2015

Publications

[1] (preprint) **Sung Kook Kim**, Arnaud J. Venet, and Aditya V. Thakur. Memory-efficient fixpoint computation. *CoRR*, abs/2009.05865, 2020. URL: https://arxiv.org/abs/2009.05865, arXiv:2009.05865

[2] **Sung Kook Kim**, Arnaud J. Venet, and Aditya V. Thakur. Deterministic parallel fixpoint computation. *Proc. ACM Program. Lang.*, 4(POPL):14:1–14:33, 2020. doi:10.1145/3371082

[3] Hyukwoo Park, **Sung Kook Kim**, Jung-Geun Park, and Soo-Mook Moon. Reusing the optimized code for javascript ahead-of-time compilation. *TACO*, 15(4):54:1–54:20, 2019. doi:10.1145/3291056

[4] Hyukwoo Park, **Sung Kook Kim**, and Soo-Mook Moon. Advanced ahead-of-time compilation for javascript engine: work-in-progress. In *Proceedings of the 2017 International Conference on Compilers, Architectures and Synthesis for Embedded Systems, CASES 2017, Seoul, Republic of Korea, October 15-20, 2017*, pages 16:1–16:2, 2017. doi:10.1145/3125501.3125512

Master's Thesis

[1] **Sung Kook Kim**. Recycling the optimized machine codes generated by javascript engine. Master's thesis, Seoul National University, South Korea, 2017

Research Experience

Graduate Student Researcher UC Davis Sep 2018 - current

- PI : Prof. Aditya V. Thakur (UC Davis)
- Objective : Design and implement scalable abstract interpretation-based static program analysis tool.
- Contributions : Designed and implemented deterministic parallel fixpoint engine and memory management technique in NASA's IKOS (C++).
- Paper accepted to POPL 2020, SAS 2020.
- The work has been pushed to Facebook's open source static program analysis tool, SPARTA.
- The work has been pushed to NASA's open source static program analysis tool, IKOS.

Graduate Student Researcher SNU & Electronics and Telecommunications Research Institute Aug 2015 - Jun 2017

- PI : Prof. Soo-Mook Moon (SNU)
- Objective : Speed up JavaScript execution engine in embedded systems by caching compiled binaries.
- Contributions : Analyzed the engine, implemented code caching & pointer patching modules in Apple's JSC (C++).
- Paper accepted to CASES 2017, TACO 2019.

Graduate Student Researcher SNU & Samsung Electronics Feb 2017 - Jun 2017

- PI : Prof. Soo-Mook Moon (SNU)
- Objective : Transpile DRAM test suites written in one language to the other.
- Contributions : Wrote front-end for both test languages (JS), implemented simulators (JS).
- Paper accepted to ETS 2019 (I'm not an author).

Independent Project Team BCDEF & Naver Mar 2015 - Aug 2015

- PI : Sung Kook Kim
- Objective : Design cruising electronic skateboard as an alternative means of transportation on campus.
- Contributions : Designed and prototyped novel cruise control for electronic skateboards.

- Demo : https://vimeo.com/174919036
- Received scholarship from Naver, won 2nd place in KAIST's Startup Competition.

Undergraduate Student Researcher Mobile Robotics & Intelligence Lab, KAIST SUMMER 2014

- PI : Prof. Jinwhan Kim
- Objective : Design a motor thrust controller for the unmanned surface vechicle (ship).
- Contributions : Analyzed protocols used in the motor thrust, designed and implemented the control system (C), implemented both server and client for the communication (C).
- The team won 2nd place in Maritime RobotX Challenge 2014, an internatial competition. Team website

OTHER EXPERIENCES

Summer School The Ninth Summer School on Formal Techniques, SRI International SUMMER 2019

- A summer school on Formal Methods held by Stanford Research Institute. SSFT19
- Topics include logic, automated theorem provers, SMT solvers, abstract interpretation, model checking.
- Took lectures and participated in labs.

Internship, Drone Engineer Nearthlab FEB 2018 - JULY 2018

- Worked as an engineer on a drone startup, Nearthlab.
- Contributions : Implemented mission planning system that controls internal & external messages (ROS, C++, JS), implemented control channel using LAN module (C++), implemented part of the web interface (React JS).

Open Source Contribution Naver SPRING 2015

- Contributed to Naver's open-source cloud monitoring tool, Pinpoint.
- Contributions : Implemented a module that allows Pinpoint to track usage of Jackson API (Java).

Developer Camp Mobile App Development Camp, Bon Angels WINTER 2014

- A mobile app development camp held by Bon Angels.
- Bon Angles is an investment company for early stage startups in South Korea.
- Built 4 Android applications: Basic text messenger (Android, Java, Node JS); Unity based MMO game (Unity, C#); Voice-based SNS (Android, Java, Node JS); App generating app (Android, Java, OpenJDK)

TEACHING EXPERIENCE

Teaching Assistant Dept. of Computer Science, UC Davis FALL 2020

- ECS 240 Programming Languages (Graduate-level, class size: 68)
- The course dealt with automated reasoning of programs to verify their correctness. In particular, the theory and pratice of static program analysis were covered. Topics include control-flow analysis of program, points-to analysis, intraprocedural and interprocedural data-flow analysis, theory of abstract interpretation, shape analysis, and model checking.
- Had weekly office hours; answered questions online; came up with some of the questions and programming assignments; marked assignments and exams

Teaching Assistant Dept. of Computer Science, UC Davis SPRING 2020

- ECS 140A Programming Languages (Undergraduate-level, class size: 76)
- Course topics include syntax and semantics of programming languages, object-oriented programming and Go, functional programming and Lisp, logic programming and Prolog, Concurrent programming in Go, etc
- Had weekly virtual office hours; answered questions online

Teaching Assistant Dept. of Computer Science, UC Davis FALL 2019

- ECS 240 Programming Languages (Graduate-level, class size: 56)
- The course dealt with automated reasoning of programs to verify their correctness. In particular, the theory and pratice of static program analysis were covered. Topics include control-flow analysis of program, points-to analysis, intraprocedural and interprocedural data-flow analysis, theory of abstract interpretation, shape analysis, and model checking.
- Had weekly office hours; answered questions in person and online; came up with some of the questions; marked quizzes, homeworks, and exams; proctored exams

Teaching Assistant Dept. of Electrical and Computer Engineering, SNU SPRING 2017

- Introduction to Computer Science (Freshman-level, class size: 80)
- Covered C programming + OOP from Java: I/O, primitive data types, variables, expressions, statements, functions, structs & arrays, pointers, and OOP

- Ran practice session every week; gave lectures twice a week with my own slides; had office hours; answered questions in person and online; gave out quizzes & projects and marked them; proctored exams

Teaching Assistant Dept. of Electrical and Computer Engineering, SNU Fall 2015

- Introduction to Compilers (Senior-level, class size: 20)
- Covered front-end of compilers: lexing and parsing
- Marked assignments and exams; proctored exams

Scholarships, Honors and Awards

- Recipient of SAS 2020 Radhia Cousot Young Researcher Best Paper Award
- Runner Up for the GGCS 2020 Best Graduate Researcher Award
- Travel Grants: POPL 2020, SSFT 2019
- Student Startup Scholarship, Naver, 2015
- Second Place in Maritime RobotX Challenge, ONR, 2014