

Pengfei Su

Assistant Professor

Department of Computer Science and Engineering

University of California, Merced

Email: psu9@ucmerced.edu

Website: <https://pengfei-su.github.io>

Work Experiences

- **UC Merced** Merced, CA
Assistant Professor Jan 2021- Present
- **Facebook** Menlo Park, CA
Research Engineer May 2020- Nov 2020
- **Uber** Palo Alto, CA
Software Engineering Intern May 2019 - Aug 2019

Education

- **College of William & Mary** Williamsburg, VA
Ph.D. in Computer Science Aug 2016 - Aug 2020
Advisor: Xu Liu
- **ICT, Chinese Academy of Sciences** Beijing, China
M.S. in Computer Science Aug 2013 - Jul 2016
- **Yunnan University** Yunnan, China
B.E. in Network Engineering Aug 2009 - Jul 2013

Research Interests

- Programming Languages
- Static and Dynamic Program Analysis
- High-Performance/Parallel Computing
- Software Engineering

Publications

- [arXiv'21] "DJXPerf: Identifying Memory Inefficiencies via Object-centric Profiling for Java", Bolun Li, Pengfei Su, Milind Chabbi, Shuyin Jiao, Xu Liu.
- [SC'19] "Pinpointing Performance Inefficiencies via Lightweight Variance Profiling", Pengfei Su, Shuyin Jiao, Milind Chabbi, Xu Liu, The International Conference for High Performance Computing, Networking, Storage and Analysis, Nov 17-22, 2019, Denver, CO, USA.
- [ESEC/FSE'19] "Pinpointing Performance Inefficiencies in Java", Pengfei Su, Qingsen Wang, Milind Chabbi, Xu Liu, The 27th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, Aug 26 - 30, 2019, Tallinn, Estonia.
- [ICSE'19] "Redundant Loads: A Software Inefficiency Indicator", Pengfei Su, Shasha Wen, Hailong Yang, Milind Chabbi, Xu Liu, The 41st IEEE/ACM International Conference on Software Engineering, May 25 - Jun 1, 2019, Montreal, Canada. **ACM SIGSOFT Distinguished Paper Award**
- [PPOPP'19] "Lightweight Hardware Transactional Memory Profiling", Qingsen Wang, Pengfei Su, Milind Chabbi, Xu Liu, The 24th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, Feb 16-20, 2019, Washington, D.C.. **Best Paper Award**

Honors and Awards

- Stephen K.Park Graduate Research Award 2020
- ACM SIGSOFT Travel Grant, ESEC/FSE'19 2019
- Distinguished Paper Award, ICSE'19 2019
- Best Paper Award, PPOPP'19 2019
- ACM SIGPLAN Travel Grant, PPOPP'19 2019
- NSF Travel Grant, PPOPP'19 2019
- Outstanding Student Award (Top 5%), Chinese Academy of Sciences 2014&2015
- Outstanding Student Award (Top 3%), Yunnan University 2010&2011&2012

Professional Services

- Program Committee HIPS'21, LCTES'21
- Artifact Evaluation Committee ASPLOS'20, CGO'18&19&20, PPOPP'18&19&21
- Conference Sub-reviewer HPCA'20, CGO'20, IPDPS'20, ICPP'17&19&20, BIGCOM'19
- Conference Volunteer ASPLOS'18

Teaching

- **UC Merced** Merced, CA
Instructor for Introduction to Object-orientated Programming (CSE165) Fall 2021
- **College of William & Mary** Williamsburg, VA
Teaching Assistant for Principles of Programming Languages (CSCI312) Spring 2018, Fall 2017
- **College of William & Mary** Williamsburg, VA
Teaching Assistant for Algorithms (CSCI303) Spring 2017, Fall 2016

Research Highlights

- **LoadSpy — a fine-grained performance tool for pinpointing redundant memory loads**
 - Shows that redundant memory loads are a common indicator of various forms of software inefficiencies.
 - Proposes new strategies for analyzing profiling data by attributing redundancies to runtime contexts, objects, and scopes.
 - Pinpoints large quantities of temporal and spatial redundant memory loads in well-known real-world applications that are the subjects of optimization for years.
 - Available at <https://github.com/CCTLib/cctlib>.
- **FVSampler — a lightweight performance tool for pinpointing function-level execution variance**
 - Quantifies execution variance across different invocations of the same function.
 - Uses performance monitoring units (PMU) to sample function call and uses debug registers to intercept the return from the same function invocation to monitor whole function instances.
 - Overcomes a critical missing piece in existing sampling-based tools — synchronize samples with function boundaries to monitor whole function instances.

- Available at <https://github.com/WitchTools/FVSampler>.
- **TXSampler — a lightweight performance tool for hardware transactional memory (HTM) profiling**
 - Quantifies the time spent in different components (e.g., transaction path, fallback path) of an HTM-based critical section.
 - Exploits call stack unwinding to construct the calling contexts outside transactions and exploits last branch records (LBR) to deduce the calling contexts inside transactions.
 - Available at <https://github.com/ScalableMachinesResearch/TXSampler>.
- **JXPerf — a lightweight performance tool for pinpointing redundant memory operations in Java**
 - Pinpoints three kinds of redundant memory operations (i.e., dead stores, silent stores, silent loads) by combining PMU and debug registers.
 - Works at the machine code level with no bytecode instrumentation.
 - Requires no modifications to hardware, OS, JVM, or monitored applications.
 - Available at <https://github.com/ScalableMachinesResearch/JXPerf>.
- **DJXPerf — a lightweight object-centric performance tool for pinpointing locality issues in Java**
 - Develops a novel object-centric profiling technique.
 - Combines PMU and lightweight bytecode instrumentation to reduce runtime and memory overheads.
 - Addresses the distinct challenges caused by just-in-time compilation and garbage collection.