Zhiqiang Zang

Education

2018–May 2024 (Expected) **Ph.D.**, *The University of Texas at Austin (UT Austin)*, Austin, TX, USA.
Software Engineering and Systems — Advisor: Milos Gligoric

2014–2018 **B.E.**, *Beijing University of Posts and Telecommunications (BUPT)*, Beijing, China.
Telecommunication Engineering

Publications

[7] **Zhiqiang Zang**, Fu-Yao Yu, Aditya Thimmaiah, August Shi, and Milos Gligoric. Java JIT Testing with Template Extraction. In *International Conference on the Foundations of Software Engineering*, page to appear, 2024.

[6] **Zhiqiang Zang**, Aditya Thimmaiah, and Milos Gligoric. JOG: Java JIT Peephole Optimizations and Tests from Patterns. In *International Conference on Software Engineering, Tool Demonstrations Track*, page to apear, 2024.

[5] **Zhiqiang Zang**, Aditya Thimmaiah, and Milos Gligoric. Pattern-Based Peephole Optimizations with Java JIT Tests. In *International Symposium on Software Testing and Analysis*, pages 64–75, 2023.

[4] **Zhiqiang Zang**, Fu-Yao Yu, Nathaniel Wiatrek, Milos Gligoric, and August Shi. JAttack: Java JIT Testing using Template Programs. In *International Conference on Software Engineering, Tool Demonstrations Track*, pages 6–10, 2023.

[3] **Zhiqiang Zang**, Nathaniel Wiatrek, Milos Gligoric, and August Shi. Compiler Testing using Template Java Programs. In *International Conference on Automated Software Engineering*, pages 23:1–23:13, 2022. **ACM SIGSOFT Distinguished Paper Award**.

[2] Pengyu Nie, Marinela Parovic, **Zhiqiang Zang**, Sarfraz Khurshid, Aleksandar Milicevic, and Milos Gligoric. Unifying Execution of Imperative Generators and Declarative Specifications. In *International Conference on Object-Oriented Programming, Systems, Languages, and Applications*, pages 217:1–217:26, 2020.

[1] Ben Buhse, Thomas Wei, **Zhiqiang Zang**, Aleksandar Milicevic, and Milos Gligoric. VeDebug: Regression Debugging Tool for Java. In *International Conference on Software Engineering, Tool Demonstrations Track*, pages 15–18, 2019.

Patents

[1] Hiroaki Yoshida, **Zhiqiang Zang**, Mukul R. Prasad. Generation of Software Program Repair Examples. US11099817B1. Aug 24, 2021.

Industry Experience

Feb 2024–Present **Research Scientist**, *Meta Platforms*, Menlo Park, CA, USA.
Worked on AI infra efficiency.

+1 (737) 333 2408 • zhiqiang.zang@utexas.edu • zzqpro.net
cptgit

May–Aug 2023 **Software Engineer Intern**, *Uber Technologies*, San Francisco, CA, USA.
Supported contracts in a Go static analyzer called NilAway (`https://github.com/uber-go/nilaway`) to reduce false positive errors.

May–Aug 2022 **Software Engineer Intern**, *Meta Platforms*, Bellevue, WA, USA.
Migrated a large-scale data analytic engine onto a new environment to improve performance and maintainability.

May–Aug 2020 **Research Intern**, *Fujitsu Laboratories of America*, ~~Sunnyvale, CA, USA~~ Remote.
Improved software program repair examples by minimizing buggy/fixed pairs of programs.

Jul–Aug 2019 **Software Engineer Intern**, *NIO*, Beijing, China.
Developed and improved a physical and visual simulator for autonomous vehicles (Unreal Engine).

Selected Projects

JAttack `https://github.com/EngineeringSoftware/jattack`
A template-based compiler testing framework that 1) accepts as input a template Java program written in a domain-specific language in Java; 2) executes the template to generate concrete programs and 3) tests Java JIT compilers using the generated programs. JavaParser, OW2 ASM, DSL, JIT, Java. Team Project. Discovered CVEs: CVE-2020-14792, CVE-2022-21305, CVE-2023-22044, CVE-2023-22045.

JOG `https://github.com/EngineeringSoftware/jog`
A framework that facilitates developing Java JIT peephole optimizations: 1) accepts as input a pattern written in the same way that tests are written; 2) translates the pattern into C/C++ code as a JIT optimization pass, and generates Java tests for the optimization; 3) detects shadow relation between a pair of optimizations. JavaParser, Z3, DSL, JIT, Java. Team Project. 7 pull requests have been integrated into the master branch of OpenJDK.

VeDebug `https://github.com/EngineeringSoftware/VeDebug`
A Java debugging tool that 1) automatically sets breakpoints where the current execution diverges from the previously captured one, and 2) provides video player features e.g., speed up/slow down the replay. Bytecode instrumentation(OW2 ASM, javaagent), Java. Team project.

Honors & Awards

2022 ACM SIGSOFT Distinguished Paper Award [3], ASE 2022

2017 Outstanding Undergraduate Award Finalist, BUPT

2015–2016 Qualcomm Innovation Scholarship, BUPT

2014–2015 Qualcomm Innovation Scholarship, BUPT

Presentations

Jul 2023 *Pattern-Based Peephole Optimizations with Java JIT Tests* [5] at ISSTA 2023, Seattle, WA, USA

Oct 2022 *Compiler Testing using Template Java Programs* [3] at ASE 2022, Oakland Center, MI, USA

Teaching Experience

Fall 2023 Teaching Assistant, UT Austin, ECE 382V: *Programming Paradigms* (26 students)

Sep 2022 Guest lecture, *Compiler Testing using Template Java Programs* [3], UT Austin, ECE 382V: *Software Tests in the Era of Nondeterminism*

Fall 2021 Teaching Assistant, UT Austin, EE 379K: *Programming Paradigms* (38 students)

Spring 2021 Teaching Assistant, UT Austin, EE 360T: *Software Testing* (101 students)

Mar 2021 Guest lecture, *(JIT) Compiler Testing*, UT Austin, EE 360T: *Software Testing*

Spring 2020 Teaching Assistant, UT Austin, EE 312H: *Software Design and Implementation I* (35 students)

Fall 2019 Teaching Assistant, UT Austin, EE 312: *Software Design and Implementation I* (60 students)

Spring 2019 Teaching Assistant, UT Austin, EE 422C: *Software Design and Implementation II* (172 students)

Professional Service

CGO 2024 Artifact evaluation committee member, *International Symposium on Code Generation and Optimization*

ICSE 2023 External reviewer, *International Conference on Software Engineering*

Fall 2022 Co-organizer, *Joint UT-Cornell Software Engineering Seminar*

ICSE 2022 External reviewer, *International Conference on Software Engineering*

ASE 2021 External reviewer, *International Conference on Automated Software Engineering*

ISSTA 2020 Artifact evaluation committee member, *International Symposium on Software Testing and Analysis*

ISSTA 2020 External reviewer, *International Symposium on Software Testing and Analysis*

Technical Skills

- Programming Languages: Java (everyday), Bash (everyday), Python, C, Go
- Tools: Emacs (everyday), Git (everyday), OW2 ASM, JavaParser