Hongyu Chen

hongyc4@uci.edu | Hongyu Chen | github.com/hongyu-dev

Education

Ph.D. in Computer Science, University of California, Irvine — **September 2023 - Present**
Research in binary translation. Paper under submission. — *Advisor: Professor Michael Franz*

M.S. in Computer Science, University of California, Irvine — **September 2021 - June 2023**

B.S. in Computer Science, University of California, Irvine — **September 2016 - December 2020**

Skills

Languages: C/C++, Assembly(x86/64, AArch64), Python, Rust, Linker Scripts, Bluespec, SQL, Shell Scripting
Frameworks/Tools: LLVM, MLIR, Clang, GCC, Docker, Ghidra, CMake, Ninja, Valgrind, Maven, Git, ngrok, TensorFlow

Selected Experience

Binary Ninja Decompiler Infrastructure Enhancement — June 2025 - September 2025

Intern, Vector 35, Melbourne, FL

- Implemented Xtensa register windowing support and enhanced call site analysis by designing complete register window rotation system with modified parameter recognition, enabling efficient calls analysis with overlapping register sets.
- Architected ILTransparentCopy attribute system allowing SSA analysis algorithms to trace through architectural register assignments while preserving logical data flow, improving decompilation optimization accuracy.
- Fixed ELF binary loader to correctly handle program headers and section addressing by modifying section-to-segment mapping logic, resolving NOTE section misclassification issues that caused incorrect memory layout analysis.
- Enhanced constant folding optimization by implementing double-precision instructions evaluation in the intermediate representation, reducing unnecessary runtime computations and improving decompiled code quality.
- Architected a scalable calling convention model using register classes and lists with separate ID spaces, enabling TriCore architecture support while benefiting all processor architectures and eliminating fallback to heuristics for function parameter identification.

Improve Linker Script Handling in LLD — June 2024 - September 2024

Software Engineering Intern, Google LLC, San Jose, CA

- Enhanced the lexer to handle complex linker scripts more efficiently, reducing errors and improving maintainability.
- Replaced lambda-based expression handling with structured expression types, improving potential performance, reducing memory overhead, and enhancing debuggability.
- Collaborated with upstream maintainers to ensure smooth integration of changes, addressing bottlenecks in the review and feedback process.

Intel VTune Support to LLVM JITLink — November 2023 - February 2024

- Integrated Intel VTune profiler support into JITLink for enhanced performance analysis.
- Enabled profiling of JIT-compiled code using VTune's API.
- Developed a test based on existing LLVM JIT profiling tools.

SMPL Compiler — January 2023 - March 2023

- Developed an optimizing compiler in C++ for the SMPL programming language.
- Constructed a recursive descent parser, generating a Static Single Assignment(SSA)-based intermediate representation(IR) to facilitate advanced optimization techniques.
- Transformed the program into an SSA form, ensuring each variable is assigned once, simplifying data flow analysis.
- Utilized the SSA form to implement optimization, including copy propagation and common subexpression elimination.
- Extended the compiler to support array operations, focusing on eliminating redundant array loads.
- Developed a global register allocator, leveraging the SSA form to track live ranges, build interference graphs, and perform graph coloring for optimal register allocation.

Microbenchmark for Manycore system HammerBlade — June 2022 - September 2022

- Explored and designed a runtime system for HammerBlade to facilitate swift task migration and reduce memory-to-icache transfers.
- Implemented a microbenchmark in C to assess the trade-off between cache misses and core hopping.
- Profiled and compared the performance difference between the normal execution model and the core hopping model.

Laboratory Information Management System — September 2020 - September 2021

Software Engineer, Burning Rock Dx, Irvine, CA

- Implemented the web service with Python, Flask on MySQL for laboratory information management.
- Implemented RESTful API with Flask-WTForm, Flask-HTTPAuth, and Flask-Login for data collection and search.
- Designed and managed the database models with Flask-SQLAlchemy, Flask-Migrate, and Python Shell.
- Experienced in Agile development and maintained the new system version on Apache for staging environment.