Ruqing Yang

yangrq.lambda@gmail.com github.com/waterlens

Research Interests

I aim to improve programming languages to enhance performance and provide stronger guarantees for users.

Education

Hong Kong University of Science & Technology **Sept. 2023 - Aug. 2025 (expected)**
M. Phil. in *Computer Science and Engineering*. Supervised by Lionel Parreaux. *Hong Kong S.A.R., China*

Zhejiang University **Sept. 2019 - June 2023**
B. Eng. in *Computer Science and Technology*. GPA: 3.84/4.0 *Hangzhou, China*

Projects

Calocom 🔗 ***written in Rust*** **Spring 2022**
- A coursework implementing a programming language with algebraic data type, closure, and pattern matching.
- Type checking, closure conversion, LLVM-based code generation included.

SyOC 🔗 ***written in C++*** **Spring 2022 - Summer 2022**
- An optimizing compiler for a subset of C language, typical dataflow analysis: immediate dominator analysis, iterated domination frontier analysis for SSA construction.
- Optimizations: constant propagation, CFG simplification, and dead code elimination.

MLScript 🔗 ***written in Scala*** **Autumn 2023 - Now**
- ANF-based IR with join points extension with a non-duplicate partial inliner leveraging function splitting.
- C++ backend with a universal object representation, and reference counting for memory management.

MMM 🔗 ***written in MoonBit*** **Autumn 2024**
- An Optimizing compiler for the *Mini MoonBit* language with JS, RISC-V and WASM backends.
- Selective CPS transformation and thunking on function calls to avoid stack overflow in the JavaScript backend.
- Tree-pattern covering instruction selector and chordal graph coloring register allocator.
- Optimizations: lambda lifting, loop invariant code motion, local value numbering, and guaranteed tail recursion elimination.

RMatch 🔗 ***A regular expression engine of NFA VM and JIT compilation*** **Autumn 2021**

Apple μArch Bench 🔗 ***Measure architecture characteristics of Apple Silicon*** **Spring 2024**

SIB Optimization for OCaml 🔗 ***Internally used in MoonBit compiler*** **Spring 2025**

Monoid Hash 🔗 ***Extending fast-crc32 with hardware accelerated monoid combination*** **Spring 2025**

Publications

Smart Inlining through Function Splitting, *PLDI SRC 2025* **April 2025**

Experience

Intern for Programming Language Tool Development, *at IDEA* **Mar. 2025 - June 2025 (expected)**

Student Volunteer, *ICFP 2024* **Sept. 2024**

Teaching Assistant, *Programming with C++* **Jan. 2024 - June 2024**

Remote Research Intern, *hosted by Yizhou Zhang* **Sept. 2022 - Jan. 2023**

Undergraduate Teaching Assistant, *Principles of Programming Languages* **Sept. 2022 - Jan. 2023**

Skills

Programming Languages: OCaml, Rust, C/C++, Scala, Java, Python, etc.
Proof Assistant: Coq