

教育背景

南京大学

2023.09 - 2026.06 (预计)

硕士, 计算机科学与技术 | [Pascal Lab](#), 导师: 李 [\[redacted\]](#) | 研究方向为程序语言与程序分析

北京航空航天大学

2019.09 - 2023.06

本科, 计算机科学与技术 | GPA: 3.84/4.00

工作经历

NVIDIA · GPU 编译器 LLVM 后端实习生

2025.02 - 至今

- 合并 NVVM 与图形编译器的向量化器, 使得图形编译器的向量化器与 LLVM 上游保持一致
 - 在 LLVM 向量化器中实现了对图形访存指令的支持, 并移植了多项关键优化, 包括偏移空位填补、访存对齐推断等;
 - 成功合并两个优化, 实现了性能对等, 同时优化了编译时间, 方便了后续的维护工作;

[Rust Foundation Fellowship Program](#) · 开源社区资助

2024.09 - 2025.09

作为 rust-analyzer 的 maintainers 之一, 负责维护 rust-analyzer (Rust 语言官方 IDE) 等 rust 语言基础设施

- 社区贡献排名 21/972, 解决 ~70 issues, 参与维护过绝大多数模块;
 - 参与解决 issues、会议讨论、PR 审核等社区维护工作; 为项目添加控制流导航、快照测试更新等多项功能;
 - 为 IDE 的 unicode 断字和断行模块编写了 NEON 下的 SIMD 实现, 使该模块在 ARM 平台上提速 6.5 倍;
 - v0.3.1992 事故救火: 社区在发布小版本 4 小时后, 发现该版本存在导致资源耗尽且无法结束进程的恶性 BUG。本人在 3 小时内定位到错误算法, 并设计新算法解决了问题。该紧急修复控制了事故影响范围, 避免影响全球 Rust 开发者的工作。

奖项荣誉

- 2022 年国家奖学金 (该学年专业排名 1/195); 北京航空航天大学优秀毕业生;
- 2021 年全国大学生计算机系统能力大赛 · 编译系统设计赛 (华为毕昇杯) 一等奖;
- 蓝桥杯 C++ 程序设计竞赛北京赛区一等奖、国赛三等奖;
- 另获其他各类省、校级奖项与奖学金十余次。

项目开发

Vizsla · 现代化芯片设计 IDE (Rust / SystemVerilog)

🔗 [roife/vizsla](#) (WIP)

- 基于增量计算架构, 实现了面向 SystemVerilog 的语义分析框架以及 IDE 基础设施, 旨在为芯片设计配备现代 IDE 功能;
- 项目在功能、性能与可用性等指标上均达到业界先进水平: 已完成代码导航、重构、代码补全等数十项现代 IDE 特性, 并能够做到毫秒级延迟; 项目已通过语言服务器协议适配 VS Code、Emacs 等主流编辑器。

Ailurus · 实验性质的编程语言及工具链设计探索 (Rust)

🔗 [roife/ailurus](#) (WIP)

- 基于 Martin-Löf 类型论; 支持双向类型检查、dependent type、indexed inductive types、pattern matching、module system 等特性; 使用 NbE 进行等价检查, 实现了 propositional equality, 支持定理证明;
- 项目包含解释器、REPL 和一个 VM; VM 使用 SSA 作为 IR, 能够执行 GVN 等运行时优化, 并通过 JIT 生成目标代码。

LLVM-Lite · 面向深度学习算子的轻量编译器 / 本科毕设 (C++ / LLVM-IR)

🔗 [roife/llvm-lite](#)

- 项目包含一个轻量编译器和经裁减的 LLVM codegen 模块, 旨在利用端侧推理设备的形状信息对深度学习算子进行二次优化;
- 毕设获优秀评价; 成功将 2D 卷积算子和 softmax 算子的用户侧推理时间降低 6%, 并将生成的二进制目标文件减小 38%;

Ayame · C 子集编译器 / 毕昇杯比赛项目, 合作 (Java / LLVM-IR / ARM)

🔗 [No-SF-Work/ayame](#)

- 主要负责完成基于图着色的迭代寄存器合并算法, 以及指令调度、窥孔优化等 Machine IR 上的优化; 并用 docker 搭建了本地 CI;
- 项目在 SSA IR 与 Machine IR 上进行优化, 在比赛中获一等奖。本项目在比赛中总排名第二, 在近一半样例上排名第一, 并在 1/3 的样例上优化效果超越 gcc -O3 与 clang -O3。

开源社区贡献

- 目前是 Rust-lang Member (rust-analyzer contributors team) 的一员; 在 rust 社区的工作集中在 🔗 [rust-lang/rust-analyzer](#) 上, 同时也贡献过 🔗 [rust-lang/rust](#), 🔗 [rust-lang/rust-clippy](#), 🔗 [rust-lang/rustup](#), 🔗 [rust-lang/rust-mode](#) ;
- 🔗 [llvm/llvm-project](#), 🔗 [clangd/vscode-clangd](#), 🔗 [google/autocxx](#), 🔗 [yuin/goldmark](#), 更多项目见 [GitHub](#)。

专业技能

编程语言	不局限于特定编程语言。熟悉 C, C++, Rust, Java, Python, JavaScript/TypeScript, Verilog/SystemVerilog; 也使用过 Ruby, Swift, OCaml, Haskell, Coq, Agda 等。
程序语言理论	了解形式语义、形式验证与计算理论的相关知识, 熟悉 Hindley-Milner 等类型系统的理论和实现。
编译器	4 年经验。熟悉解释器/编译器的全 pipeline 开发; 熟悉多种 IR (如 SSA / LLVM-IR, DBI, ANF, MLIR 等) 和编译优化 (如 Mem2Reg, GVN/GCM, 寄存器分配, 指令调度等); 了解 LLVM, 熟悉部分优化和 codegen 模块; 了解 Mark-Sweep、Copying 等常见的 GC 算法原理和实现。
IDE 开发	2 年经验, 熟悉基于增量计算的 IDE 架构, 尤其熟悉 rust-analyzer 和 clangd 的架构和实现; 了解 VS Code、Emacs 等各主流编辑器的插件开发, 熟悉语言服务器协议 (Language Server Protocol) 的规范和实现。
程序分析	熟悉常见静态分析算法, 如数据流分析、区间分析、IFDS 等, 熟悉基于不同敏感性的指针分析算法;
系统编程	了解体系结构和操作系统相关知识, 能进行汇编级的开发调试工作, 了解 Docker, GDB, CMake 等工具。
开发环境	熟悉 Emacs; 习惯在 macOS / Linux 下工作; 能熟练使用 AI 提高工作效率。

其他

- 助教: 程序设计基础 (2020 秋), 面向对象设计与构建 (2021 秋, 2022 春), 编译原理 (2024 春)。
- 外语: 英语 (CET-6);