吴家焱

L (+86) 000-0000-0000 | **≥** roifewu@gmail.com | **♦** roife | **⊕** roife.github.io

m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: 李樾 | 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

盘工作经历

NVIDIA OCG 2025.02 - 至今

GPU 编译器 LLVM 后端实习生

- 负责统一 NVIDIA GPU 图形编译器与 NVVM 的向量化器,使得 GPU 图形编译器的向量化器与 LLVM 上游保持一致
 - ▶ 通过指令替换,在新向量化器中添加了 surface.load 等多条 GPU 图形访存指令的支持,并尽可能减小与上游的差异;
 - 向新向量化器中移植了多项访存向量化相关的优化,包括偏移空位填补、访存对齐推断、已知比特推断等;

Rust Foundation Fellowship Program

2024.09 - 2025.09

Rust Foundation 开源社区资助(全球约 20 人)

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。

- 在社区中贡献排名 21/972, 解决 ~70 issues; 参与 issues 处理、会议讨论、PR 审核等维护工作, 参与维护过项目绝大多数模块;
- 在 rust-analyzer 中实现了控制流导航、快照测试更新等多项功能,并参与了大量 bug 修复;
- 为项目的 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

面向芯片前端设计的现代化 IDE:硕士毕设项目

noife/vizsla (WIP)

Rust / SystemVerilog

- 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**:已完成面向 SystemVerilog 的代码导航、语义重构、代码补全、语义高亮、代码诊断等**数十项**现代 IDE 特性,并能够利用增量语义分析在各项功能上做到**毫秒级**延迟;
- 基于语言服务器协议,已适配 VS Code、Emacs、NeoVim 等主流编辑器。

编程语言及工具链设计探索实验 · 个人兴趣项目

noife/ailurus (WIP)

湘往市百及工共促议自体系表型。十八八座次百

Rust

- 基于 Martin-Löf 类型论; 支持双向类型检查、dependent type、模式匹配、indexed inductive types、module system 等特性;
- 使用 Normalization by Evaluation 进行等价检查,实现了 propositional equality,支持定理证明;

Ayame

Ailurus

No-SF-Work/ayame

SysY(C子集)到 ARMv7的编译器·毕昇杯比赛项目(合作)

Java / LLVM-IR / ARM

- 主要负责完成面向 Machine IR 的后端优化,包括基于图着色的**迭代寄存器合并**算法、**指令调度**、窥孔优化等;
- 同时负责项目的测试和 DevOps, 利用 docker 和 GitLab CI 搭建了测试流程,并编写了 Python 脚本自动分析测试结果;
- 项目从零开始完成了一个完整的编译器 pipeline,编写了大量 SSA IR 与 Machine IR 上的优化,最终在比赛中获一等奖。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 -03 与 clang -03。

noife/llvm-lite

面向深度学习神经网络算子的轻量端侧编译器 · 本科毕设项目

C++ / LLVM-IR

- 课题旨在利用端侧推理设备已知的形状信息,对深度学习算子进行二次优化,以减少算子运行时的时空开销;
- 项目成果包含一个运行在推理设备的轻量编译器和经裁减的 LLVM codegen 模块;成功将卷积算子和 softmax 算子的推理时间降低 6%,并将生成的二进制目标文件减小 38%;同时针对项目特性实现了**解析时优化**,将编译时间降低了 60%,将编译时内存开销降低了 60%;在毕设中获**优秀**评价。

2 开源社区贡献

- Rust Organization (rust-analyzer contributors team) 成员之一,主要负责维护 **?** rust-lang/rust-analyzer;
- 在 rust 社区也贡献过 **?** rust-lang/rust, **?** rust-lang/rust-clippy, **?** rust-lang/rustup, **?** rust-lang/rust-mode 等;
- Ollvm/llvm-project, Oclangd/vscode-clangd, Ogoogle/autocxx, Oyuin/goldmark, 更多项目见 GitHub。

四 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, JavsScript/TypeScript, Verilog/SystemVerilog; 学习 并使用过 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, System F, Dependent Type 等**类型系统** 的理论和实现;有使用 Coq, Agda 等定理证明器的经验。

编译器 / VM 4年经验, 熟悉解释器/编译器的全 pipeline 开发;

- 熟悉多种中间代码表示(如 SSA, MLIR, DBI, ANF等)、编译优化(如 Mem2Reg、GVN/GCM、寄存器分配、指令调度、自动向量化等)、垃圾回收算法(Mark-Sweep、Copying、Mark-Copy等);
- ▶ 熟悉 LLVM 和 LLVM-IR; 阅读过 LLVM 部分优化, 熟悉 LLVM 的 codegen 模块及相关优化;
- ▶ 了解 NVIDIA GPU 编译器架构和相关优化,熟悉 NVVM IR 和 PTX 指令;

静态分析 熟悉常见静态分析算法,如数据流分析、区间分析、IFDS、采用不同敏感性的指针分析等;

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。

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 工具提高工作效率。

田其他

- **助教工作**: **程序设计基础** (2020 秋), **面向对象设计与构建** (S.T.A.R. 教辅团队 / 2021 秋, 2022 春), **编译原理** (2024 春); 主 要参与课程设计、课程系统开发与运维等。
- **外语**: 英语 (CET-6);