Yiyu Liu

✉ liu_yiyu@sjtu.edu.cn · LauYeeYu

Education

Shanghai Jiao Tong University (SJTU), Shanghai, China 2021 – 2025 (Expected)

Student in **ACM Honor class**, GPA: **3.93**/4.3. Rank: **5**/33. *B.Eng. in Computer Science*

Selected courses: Computer Architecture: 99/100, Operating System: 100/100, Comprehensive Design for Computer System: 100/100, Computer Networks: 98/100.

Publication

FaaSMem: Improving Memory Efficiency of Serverless Computing with Memory Pool Architecture (ASPLOS'24)

Authors: Chuhao Xu, **Yiyu Liu**, Zijun Li, Quan Chen*, Han Zhao, Deze Zeng, Qian Peng, Xueqi Wu, Haifeng Zhao, Senbo Fu, Minyi Guo.

Improving the resource utilization of persistent memory with disaggregated architecure

Submitted to EuroSys'25 Fall.

Research Experience

Research interests: cloud computing, systems for machine learning, and hardware/software co-design.

Emerging Parallel Computing Center, SJTU, Shanghai, China Sep. 2023 – Jun. 2024

Undergraduate researcher, advised by **Prof. Quan Chen**. *Topic: cloud computing.*

- Explored the memory footprint under serverless condition and proposed a new mechanism tailored for serverless containers to improve memory efficiency. The system yields up to 71.0% reductions in local memory usage and a maximum 2.2× more deployment density. *Accepted by ASPLOS'24.*
  - Role in the project: designed the strategy to effectively and efficiently offload pages together with Chuhao; conducted experiments; wrote the paper with Chuhao.

Paul G. Allen School of Computer Science & Engineering, UW, WA, USA Jul. 2024 – Present

undergraduate researcher, advised by Prof. Baris Kasikci. *Topics: cloud computing and systems for ML.*

- Built a disaggregated system to improve resource utilization while keeping the durability guarantee of persistent memory. A few key challenges were fixed in this project to improve the performance. This system outperforms NFS by up to 85× on read-heavy workloads. *In submission.*
  - Role in the project: investigated the end-to-end procedure; made quite a few changes to the original system design; conducted experiments; wrote the paper with other authors.
- Optimized the performance of Retrieval-Augmented Generation from a system perspective. *In submission.*
  - Role in the project: helped implement and conducted core experiments; solved critical performance issues.

Teaching Experience

Programming Aug. 2022 – Jan. 2023

Data Structure Feb. 2023 – Jun. 2023

Principle and Practice of Computer Algorithms (project-based course) Jun. 2023 – Jul. 2023

Compiler Design and Implementation (project-based course) Aug. 2023 – Jan. 2024

Operating System Feb. 2024 – Jun. 2024

Roll: teaching assistant. Work includes giving lectures, writing guidebooks or guiding documents, creating exam questions, etc.

Honors and Awards

Longfor Scholarship, top 1%	Dec. 2023
2021 Zhiyuan Honors Scholarship, top 2%	Dec. 2021
2022 Zhiyuan Honors Scholarship, top 2%	Dec. 2022
2023 Zhiyuan Honors Scholarship, top 2%	Dec. 2023

Selected Project

ACM Class Online Judgement System (ACMClassOJ/TesutoHime) Sep. 2022 – Jun. 2024

Website: `https://acm.sjtu.edu.cn/OnlineJudge`
Work: Wrote the development documentation for the whole project; added new features; fixed bugs; (Feb. 2023 – Jun. 2024) operated and maintained the online judge service (∼369K submissions since 2020).

A RISCV-32I CPU (LauYeeYu/RV32I-CPU, ∼3.5K lines of Verilog code)

Adopts the speculative execution based on the Tomasulo algorithm with branch predictor, ICache, and DCache. Passed all simulation tests and FPGA tests.

A RISCV Kernel (LauYeeYu/toy-riscv-kernel, ∼3.0K lines of C code and 244 lines of asm)

A RISCV kernel that supports kernel-user space seperation and scheduling. For the user space, “init” is the first user process and can spawn into multiple processes with UNIX syscalls.

A Compiler for a C-like Language (LauYeeYu/Mx-Compiler, ∼6.2K lines of Kotlin code)

A compiler for a C-like language, Mx* on RISCV-32I.

Skills

Programming languages: C, C++, Python, Go, Java, Kotlin, Verilog and Bash.
Markup languages: HTML, Markdown and LaTeX.

Languages

English: fluent.
Chinese mandarin: native.
Shanghainese: native.