

Zuo Tianyou

✉ fztym@sjtu.edu.cn | 📞 +86.131.6290.9941

EDUCATION

Shanghai Jiao Tong University

B.S. in Electrical and Computer Engineering, Minor in Computer Science
(GPA: 3.62/4.0)

Shanghai, China

Sep. 2023 - Present

(Rank: 51/215)

RESEARCH INTERESTS

High-performance computing and GPU architecture; design and analysis of algorithms.

PUBLICATIONS

Disproofs of the Four-Coin Hereditary Subcurrency Conjecture and the Prefix Pattern Representative Conjecture. (Submitted to SODA 2027) Studied orderly currency systems of the change-making problem in which the greedy algorithm remains optimal, and constructed counterexamples that disprove two existing conjectures on hereditary subcurrency patterns.

RESEARCH & ENGINEERING EXPERIENCE

Alibaba | Inference Engine · Operator Optimization Intern

Jul. 2026 - Present

- Working on operator optimization for **PAI-vLLM**, Alibaba's in-house vLLM-based LLM inference engine: the Gated DeltaNet (GDN) linear-attention decode / MTP-verify kernels of **Qwen3-Next**, implemented in Triton.
- Identified the serial dependency of outputs on state updates as the bottleneck of the fused-recurrent kernel, and rewrote the recurrence algebraically ($o_t = d(h_{t-1}^\top q_t) + u_t(k_t^\top q_t)$) so that outputs are computed from the pre-update state: the two state scans merge into one, and the output computation leaves the per-step critical path.
- Designed a bf16 mixed-precision pipeline with fp32 only at the inputs and outputs, and justified its numerical safety by showing that decay gating is a contraction mapping on accumulated error ($\sim 6e-3$, non-growing in sequence length); state registers halved, SM occupancy doubled.
- Built the evaluation methodology (correctness against fp64 reference implementations, CUPTI kernel timing, K V and non-power-of-two shapes), which also uncovered a Triton autotune bug where benchmarking corrupted the in-place kernel state; MTP-verify kernels **10.9% faster**, 28/28 tests pass.

Moore Threads | GPU Computing & AI Software Intern

Jan. 2026 - May 2026

- Contributed to core kernels of muBLAS, Moore Threads's in-house high-performance GPU math library (counterpart to NVIDIA cuBLAS); independently responsible for the design and implementation of batched dense linear algebra routines: batched LU factorization **getrfBatched**, linear system solving **getrsBatched**, and matrix inversion **getriBatched** / **matinvBatched**.
- Systematically analyzed how shared-memory layout, memory coalescing, and reduction strategies affect performance and numerical stability; located bottlenecks through profiling and iterated on kernels, developing a complete “modeling–experiment–validation” performance-tuning methodology.
- Surveyed industrial implementations (cuBLAS, MAGMA) and related literature, comparing pivoting strategies for accuracy–performance trade-offs on small batched matrices.

LemonDB: Modern C++ Multi-threaded In-Memory Database

Repo

- Built a high-concurrency relational database in C++20; designed a pull-based thread pool (batch task fetching reduces global lock contention by 16x) and a dependency-aware scheduler that automatically resolves inter-query resource dependencies for topological scheduling.

MSD Distributed Music Recommendation System

Repo

- Built an end-to-end big-data pipeline over the Million Song Dataset (288GB of raw data compressed to 884MB Avro, cutting PySpark processing time from 5.5 to 1.3 hours) and implemented sub-second approximate-nearest-neighbor recommendation with FAISS.

TEACHING

CS4770J Introduction to Algorithms | Teaching Assistant

Sep. 2025 - Dec. 2025

Conducted lab sessions on algorithm design and implementation in OCaml, focusing on tail-recursion optimization and continuation-passing style (CPS); maintained and extended the Online Judge system, reinforced test cases, and debugged grading scripts.

CS2810J Data Structures and Algorithms | Teaching Assistant

Sep. 2025 - Dec. 2025

Led recitations covering advanced data structures and complexity analysis; designed ACM/ICPC-inspired supplementary problem sets and maintained the course OJ system.

COURSES

ENGR1510J	Accelerated Introduction to Computers and Programming	(4.0/4.0)
MATH1860J	Honors Mathematics II	(4.0/4.0)
CS2810J	Data Structures and Algorithms	(4.0/4.0)
CS4770J	Introduction to Algorithms	(4.0/4.0)
DS4720J	Methods and Tools for Big Data	(4.0/4.0)
CS4821J	Introduction to Operating Systems	(4.0/4.0)

PRIZES & HONORS

First Prize, NOIP 2020

SKILLS

Programming Languages	C/C++, Python, OCaml, Shell
Systems & Tools	CUDA / Triton kernel development, Linux, Git, Docker, vLLM, Spark, SSH
Language Skills	Mandarin (native), English