# Xueshen **Liu**

(+1) 734-546-5139 | liuxs@umich.edu | xenshinu.github.io | xenshinu | publications

## Introduction

I am a 4th-year Ph.D. candidate in Computer Science & Engineering (CSE) Division at the University of Michigan, advised by Prof. Z. Morley Mao. My research interests focus on **distributed systems** and **parallel computing**. Currently, I am exploring efficient solutions for training, inference, and reinforcement learning (RL) of **large language models (LLMs)** by designing **elastic** and **heterogeneous** systems.

## Education

**University of Michigan (UMich)** — *Ann Arbor, Michigan, U.S.*

Ph.D. & B.S. in Computer Science and Engineering (CSE) GPA: 3.9/4.0 — *Aug. 2020 - Aug. 2022 - Present*

**Shanghai Jiao Tong University (SJTU)** — *Shanghai, China*

B.S. in Electrical and Computer Engineering (ECE) GPA: 3.7/4.0 *Outstanding Graduates* — *Aug. 2018 - Aug. 2022*

## Projects & Publications

**GPU States Checkpointing for Distributed Elastic Serving Systems** — *Ongoing*

**Keywords:** CUDA driver, CUDA graph, checkpointing, elastic serving — *Sept. 2025 – Present*

- Save a selected range of CUDA states into an image to skip warmup during elastic serving.

**RLBoost: Harvesting Preemptible Resources for Cost-Efficient Reinforcement Learning on LLMs** — *NSDI'26*

**Keywords:** LLM RL, Spot Instances, Kubernetes, VeRL, SGLang, FSDP — *May 2025 – Dec. 2025*

- RLBoost **adaptively offloads rollout** workloads to preemptible instances, achieving up to **49% cost reduction** for LLM RL.
- Designed **token-level rollout tracking** to minimize preemption loss and balance workload across heterogeneous instances.
- Designed a **pull-based weight transfer** mechanism that allows dynamic resources to join rollout seamlessly with minimal overhead.

**HeterMoE: Efficient Training of Mixture-of-Experts Models on Heterogeneous GPUs** — *In Submission*

**Keywords:** LLM Training, NCCL, Mixture-of-Expert, DeepSpeed, Heterogeneity — *April. 2024 - April. 2025*

- HeterMoE disaggregates MoE models and **assigns experts to old GPUs** (e.g. V100, T4) to maximize hardware utilization and reduce cost.
- Designed **zebra parallelism** to overlap the communication and computation between attention and experts.
- Achieved fine-grained automatic load balancing between GPUs of different generations through **asymmetric expert assignment**.

**Plato: Plan to Efficiently Decode for Large Language Model Inference** — *COLM'25*

**Keywords:** LLM Inference, Parallel Decoding, Structured Decoding, KV-cache — *Oct. 2024 - Jul. 2025*

- *Plato* decomposes complicated questions into sub-problems with a **dependency graph**, and accelerates generation through context-aware parallel decoding.

**Compute Or Load KV Cache? Why Not Both? (CAKE)** — *ICML'25*

**Keywords:** LLM Inference, KV-cache, Chunk Prefill, Long Context, vLLM, LMCache — *Sept. 2024 - Feb. 2025*

- *CAKE* reduces LLM prefill latency on long-context through a **bidirectional KV cache generation** strategy, overlapping computation and I/O transfer. Implementation is based on vLLM and LMCache.

**Learn-To-be-Efficient (LTE): Build Structured Sparsity in Large Language Models** — *NeurIPS'24 (Spotlight)*

**Keywords:** LLM Efficiency, Structured Sparsity, MoE, Gather-scatter, Triton — *Mar. 2024 - Oct. 2024*

- *LTE* trains LLMs to activate fewer neurons through structured sparsity while maintaining accuracy.
- *LTE* proposes an efficient **gather-scatter MLP kernel** that achieves linear speedup w.r.t. sparsity.

**mm2-gb: GPU Accelerated Minimap2 for Long Read DNA Mapping** — *ACM BCB'24 (Oral)*

**Keywords:** GPU, DNA Mapping, Minimap2, ROCm, HIP, Persistent Kernel — *May. 2022 - Oct. 2024*

- *mm2-gb* is based on minimap2-v2.24 with AMD GPU accelerated chaining kernel for high throughput accurate mapping of ultra-long DNA reads.
- *mm2-gb* exploits finer levels of parallelism by dividing reads into segments. It then leverages **split-kernels** and **prioritized scheduling with persistent kernel** to tackle extremely irregular workloads.

## Experience

**Student Researcher at Google** — *Seattle, Washington, U.S.*

System Research @ Google, Google | Hosts: Juncheng Gu, Arvind Krishnamurthy, Hank Levy — *May. 2025 - Dec. 2025*

- Characterized workload bottlenecks across the LLM RL pipeline, identifying rollout as a dominant yet highly elastic component suitable for small, dynamically available instances.
- Designed and implemented **RLBoost** on Google Cloud Platform (GCP), harvesting fragmented spot resources to lower RL training cost and improve the resource utilization on the cloud.
- Explored heterogeneous compute options on GCP (multi-generation GPUs & TPUs) to evaluate rollout efficiency under diverse RL rollout workloads (sequence length, tool calling, etc.).
- Contributed to an NL2SQL agentic training pipeline, optimizing multi-node communication and applying asynchronous tool calling.

### Graduate Student Instructor

*Ann Arbor, Michigan, U.S.*

CSE-589 Advanced Computer Networks, University of Michigan — *Sept. 2024 - Dec. 2024*

- Led in-class discussions, held office hours, and delivered a lecture on distributed software-defined networking (dSDN).
- Mentored graduate students on research projects, providing guidance on methodologies, technical skills, and project development.

### Intern Researcher at General Motors

*Warren, Michigan, U.S.*

Connected Autonomous Vehicle (CAV) Lab, General Motors | Hosts: Fan Bai, Bo Yu — *May. 2024 - Aug. 2024*

- Designed a large scale latency-tolerant vehicle positioning system on the edge/cloud servers.
- Developed a **deep factor graph** model to handle delayed perception data, ensuring real-time responsiveness through parallelism.

## Services & Honors

2024-2025 **Reviewer**, ICLR'26, ICLR'25, COLING'25

Aug. 2024 **Invited Talk**, Scalable & Latency-tolerant Edge/cloud Computing via Deep Factor Graph — *General Motors*

May 2024 **Invited Talk**, AMD HPC Apps Knowledge Sync: Minimap2-gigabases (mm2-gb) — *AMD*

Aug. 2021 **Roger King Scholarship**, College of Engineering of University of Michigan — *UMich*

Aug. 2019 **Runner-up Team & Grand Prize**, 18th Robomaster Final Competition — *DJI*

## Skills

**Machine Learning** VeRL, Pytorch, DeepSpeed, NCCL, SGLang, vLLM, Flash-attn, LMCache, HuggingFace, CUTLASS

**Programming Language** Python, Rust, Triton, CUDA, HIP, C/C++, Golang, LLVM

**Development & Profiling** Kubernetes, Nsight-system/compute, MCP, Cusor/Codex, Perfetto, Slurm, Docker, Git