Hanmin **LI**

Thuwal, SA 23955 **|Cell**: (+966) 054 807 9372 (SA) ; (+86) 180 1998 0640 (CN) **| E-MAIL**: hanmin.li@kaust.edu.sa **|**
Google Scholar: Hanmin Li - Google Scholar **| Website:** Hanmin Li **| LinkedIn:** Hanmin Li

CS PhD Student at KAUST CoE for Generative AI ● LLM Training ● Optimizers ● Distributed Training

PROFILE

- CS PhD candidate at KAUST supervised by Prof. Peter Richtárik, focuses on **large language models (LLMs) training** and **optimization**, with emphasis on training efficiency, scalability, and distributed learning on large GPU clusters.
- Hands-on experience with LLM pretraining and fine-tuning using **PyTorch** (DDP, FSDP, DeepSpeed, Hugging Face Accelerate), including optimizer/operator customization, communication optimization, and familiarity with end-to-end LLM training pipelines.
- Solid theoretical background in (non)convex optimization, reinforcement learning and policy optimization methods (e.g. PPO-style algorithms) as applied to LLM alignment and agentic systems.
- Broad interest in reinforcement learning, LLM-based agents, and decision-making systems, particularly at the intersection of optimization, scalability, and learning dynamics.

SKILLS

- **LLM Training & Fine-tuning:** Pretraining, SFT, RLHF (PPO/DPO), evaluation pipelines, custom optimizer and operator development, communication optimization.
- **Distributed Training Frameworks:** PyTorch (DDP, FSDP), DeepSpeed, Hugging Face Accelerate, CUDA, HPC (Slurm)
- **Programming:** Python (PyTorch, NumPy, Pandas, Matplotlib), C++, R, Bash scripting**.**
- **Theory:** (non)convex optimization,optimizers, reinforcement learning, operator theory, probability theory.
- **Research Tools:** Git, Linux, LaTeX/Overleaf, Weights & Biases, Jupyter, academic publishing.

EDUCATION

- **King Abdullah University of Science and Technology (KAUST) | Thuwal, Saudi Arabia** — *2027* **(Expected)**
  PhD in Computer Science | Generative AI CoE
  Advisor: *Prof. Peter Richtárik*

- **King Abdullah University of Science and Technology (KAUST) | Thuwal, Saudi Arabia** — *Dec 2022*
  Master of Engineering in Computer Science

- **University of Science and Technology of China | Hefei, China** — *July 2021*
  School of Gifted Young, Bachelor of Engineering in Computer Science

PROJECTS & PAPERS

Efficient Layerwise Optimizers (e.g., Muon) and Distributed Training for LLMs (Ongoing Research)
Tech: PyTorch, PyTorch Distributed (DDP/FSDP), DeepSpeed, HF Accelerate, CUDA, Pretraining, SFT, LLM Pipelines

- ***Designed and improved*** *Muon variants to boost optimizer efficiency and scalability, introduced block-splitting strategies (row/column sub-blocks, top-k, magnitude-based selection) to improve stability in large-scale training.*
- ***Benchmarked*** *optimizers against Adam/AdamW and Muon on pretraining models (GPT-2 124M, Qwen2-7B, GPT-NeoX-20B) and on SFT tasks for instruction tuning.*
- ***Built distributed pipelines*** *on HPC GPU clusters (KAUST Ibex/Shaheen-III) using DeepSpeed ZeRO-3, FSDP/DDP sharding, and gradient checkpointing for long-context models.*
- ***Developed LLM workflows*** *covering Pretraining to SFT with response-only masking, mixed precision, gradient clipping, and automated evaluation dashboards.*

Broximal Point Method (BPM): A Foundational Framework for Muon, Gluon Optimizers in Large-Scale LLM Training
Tech: PyTorch, PyTorch Distributed (DDP/FSDP), Pretraining, Distributed Training, Non-convex Optimization

- ***Introduced and implemented*** *a new proximal optimization algorithm with formal convergence guarantees, revealing its connections to Trust-Region methods, classical Proximal Point methods, SGD, Muon optimizers, and Linear Minimization Oracles (LMOs).*
- ***Pioneered*** *a new proximal optimization framework, establishing the foundations that directly shaped layerwise Muon optimizers and enabled the development of Gluon, an adaptive variant for scalable LLM training.*
- ***Validated the method on NanoGPT-124M*** *pretraining, demonstrating fast, stable convergence and estimating trajectory smoothness as a quantitative measure of optimizer stability.*

FedExProx with Extrapolation and Inexact Prox: Extrapolation-Based Efficient Distributed Training

Tech: Python, PyTorch, NumPy, Distributed Training, Federated Learning

- ***Introduced*** *FedExProx with extrapolation and inexact proximal updates, creating a more flexible framework for efficient distributed optimization.*
- *Improved training efficiency with* ***zero additional runtime overhead****, making it highly scalable for large systems.*
- ***Lightweight and practical****, easy to implement and tune, robust even under coarse approximations, and delivers up to* ***4× performance gains*** *on benchmarks such as* ***CIFAR*** *and* ***Shakespeare****.*
- *Presented at* ***NeurIPS 2024*** *OPT Workshop (Poster)*

The Power of Extrapolation: From Parallel Projection to Distributed Training

Tech: Python, PyTorch, NumPy, Distributed Training, Convex Optimization

- ***Proposed extrapolation-based variants*** *of federated algorithms to accelerate convergence and efficiency.*
- ***Demonstrated consistent empirical gains****, with extrapolation achieving faster convergence and higher accuracy than FedAvg and FedProx across benchmarks such as CIFAR-10 with no overhead.*
- ***Provided rigorous theoretical analysis****, connecting extrapolation to* ***parallel projection methods*** *and establishing a unified perspective on its convergence behavior.*
- *Presented at* ***NeurIPS 2024*** *(Poster)*

Variance-Reduced Distributed Non-Convex Optimization Using Matrix Stepsizes

Tech: Python, PyTorch, Distributed Training, Non-convex Optimization, Matrix Stepsize, Variance Reduction

- ***Introduced*** *a novel variance-reduction framework for non-convex optimization by integrating matrix stepsizes with techniques such as* ***Momentum Variance Reduction (MVR)****.*
- *Presented at* ***NeurIPS 2023*** *FL@FM Workshop (Poster)*

Det-CGD: Layerwise Gradient Compression with Matrix Stepsizes for Scalable Training

Tech: Python, PyTorch, Distributed Training, Compression, Matrix Stepsize, Layerwise Training, Progressive Training

- ***Introduced Det-CGD*** *with* ***layerwise Bernoulli-style compression****, achieving communication reduction with no additional overhead.*
- *Incorporated* ***matrix stepsizes*** *to adapt learning dynamics in a* ***layer-wise*** *manner.* ***Validated experimentally****, demonstrating consistent performance improvements on deep learning benchmarks.*
- *Presented at* ***ICLR 2024*** *(Poster)*

WORK EXPERIENCE

Applied Scientist Intern @ Microsoft AI *June 2026 - Sept 2026 (Ongoing)*

HONOR & AWARDS

- KAUST CEMSE Dean's List Award 2026 (top 20%)
- Invited as a visiting scholar to ELLIIT Focused Period Lund 2026 Optimization for Learning.
- Invited as a visiting scholar to EUROPT 2024 hosted by Lund University in Sweden.
- KAUST Graduate Student Fellowship.