

# LEI HUANG

Last update: 19 Feb 2025

☎ +86 18974889415 | @winlere07@gmail.com | 📝 Blog(cnblogs.com/winlere) | 🌐 winlere.github.io | 📍 Shanghai, China

## EDUCATION

---

### ShanghaiTech University

B.Eng. in Computer Science and Technology; GPA: 3.78/4.00 (6/178)

The Outstanding Students Award (Top 1)

Shanghai, China

Sep 2021 – Jul 2025 (Expected)

## HONORS & ACHIEVEMENT

---

### Outstanding Reproducibility Report

SC23 (Student Cluster Competition Track), Denver, Nov, 2023

Profiled, optimized, and researched scientific application on a computer cluster in a 48-hour non-stop challenge.

### The 3rd Place Overall Winner

ISC23 (Student Cluster Competition Track), Hamburg, May 2023

Compiled, ran, profiled, and optimized scientific applications on supercomputers.

### Silver Medal

ICPC Asia EC Regional, Hefei, Nov, 2022

Wrote programs to solve 7 out of 13 algorithm questions in a non-stop, 5-hour challenge

## PUBLICATION

---

### Verification of Bit-Flip Attacks against Quantized Neural Networks

OOPSLA'25 (CCF-A)

Yedi Zhang, Lei Huang, Pengfei Gao, Fu Song, Jun Sun, and Jin Song Dong

- Extremely few bit-flips in parameters can reduce neural networks accuracy significantly. To formally prove the absence or identify all vulnerable parameters of quantized neural networks under bit flip attacks, we present BFAVerifier with abstraction-based method GPU-accelerated BFA\_RA and MILP-based method BFA\_MILP.

## EXPERIENCE

---

### Quant Software Engineer

Quant Dev and Infrastructure Intern

Sixie Capital

Dec 2024 – Present

- PyTorch Support: identified one bug in cutlass used by PyTorch, confirmed by the PyTorch Team.
- Testcase Compiler: designed automatic market data correctness testcase compiler to enable continuous test
- GPU Performance Profile: measured the training performance of the ML, identified no need for optimization

### Understanding Neural Network Verification Solver Research Development

Researcher

Jul 2024 – Present

- To understand current and historical research status of neural network verification, I systematically reviewed the literature from 2019 to present (2024). Currently, I am summarizing ideas from 60 major works to a slides.

### Verification of Robustness of Quantified Neural Network

Research Assistant

National University of Singapore

Dec 2023 – Jul 2024

- To understand the background of bit-flip attacks against neural networks, I conducted a thorough survey and defined the threat model. To handle the combinatorial explosion, I co-designed and developed the BFA\_RA algorithm. To conduct large-scale experiments, I accelerated BFA\_RA using C++/CUDA and PyTorch, benchmarking them on HPC clusters. The paper meets the high-quality standards of OOPSLA.

### Scalable and Exact Verification of Robustness of Neural Networks by CEGAR

Research Assistant

ShanghaiTech University

Dec 2022 – Jun 2023

- Worked on the benchmark of the project “Abstraction and Refinement: Towards Scalable and Exact Verification of Neural Networks”, working with Dr. Zhe Zhao and Prof. Fu Song. Later published on TOSEM'24.

### Teaching Assistants

Programming Lang and Compilers(23S), Intro to (modern C++) Programming(23F), Parallel Computing(24F)

ShanghaiTech University

### Student Leader

Serving as the leader of ACM and GeekPie, currently the best student associations in school.

ShanghaiTech University

Sep 2022 – Present

## PROJECTS

---

### ChocoPy: A Modern C++ based compiler from ChocoPy (subset of Python) to LLVM-IR and RISC-V | [GitHub1](#) [GitHub2](#)

- To understand compiler design, I made use of C++ visitor pattern, Flex, Bison, LLVM-IR and a RISC-V stack machine, resulting in ChocoPy Language that supports a subset of Python and cross-platform compiling.

### LBM: Microarchitecture-aware design and optimization using SIMD and other techs | [GitHub](#)

- To exploit the hardware, we leverage SSE2, OpenMP, Cache Blocking and Software Pipelining to reach 20 speedup using 4 cores.

### CuckooHash: A high performace CUDA CuckooHash implementation. | [GitHub](#)

- To understand and exploit GPU programming, I accelerated CuckooHash Table by CUDA and profiled it. It could complete  $1 \times 10^9$  insertions or  $2 \times 10^9$  lookups per second.

### TorchPoly: A clean, modern and extensible implementation for CROWN/DeepPoly using PyTorch | [GitHub](#)

- To facilitate the ML verification community with high quality software, I re-wrote DeepPoly algorithm in PyTorch by encoding it to the equivalent linear operations and provided standard benchmarks.

### PintOS: A simple operating system kernel for the 80 x 86 architecture. | [GitHub](#)

- To understand Operating System knowledge, we wrote pure C and tested on QEMU, the kernel is bootable and supports multiuser, virtual memory and large file.

### Rathernet: A full stack network virtual device written in Rust. | [GitHub](#)

- To understand Computer Network, we implemented a network virtual device starting from moduling soundwave. Finally, our set, equipped with complete functionality, can be seamlessly integrated into Windows 11.

### GuitarFEM: A Time-Space Simulation of a Guitar | [GitHub](#)

- To practice Finite-Element Method, we followed a mathematical model of three major governing differential equations to conduct simulation of a guitar.

## SKILLS

---

**Programming:** Experienced in C/C++ (including modern C++), Python, CUDA, MATLAB. Not limited to any language.

**Technologies:** Docker, PyTorch,  $\LaTeX$ , CMake, Spack, OpenMP, MPI, Gurobi and Linux Dev/Ops

**Languages:** Chinese (Native), English (Fluent)

Theory & Algorithm	Software Eng & HPC	Mathematics & ML
Theory of Computation*	Parallel Computing	Mathematical Analysis I&II
Computer Aided Verification*	Operating System	Linear Algebra I&II
Algorithm Analysis*	Computer Network	Intro to Machine Learning
Algorithm and Data Structure	Computer Architecture	Probability and Statistics
Artificial Intelligence	(Modern C++) Programming	Discrete Mathematics
Programming Language and- Information Theory	Compilers	Numerical Optimization
		Computational Science and Eng General Physics

Table 1: Courses Taken, \* indicates graduate level course