

MIKE (DEYUAN) HE

✉ mikehe@princeton.edu · in @Mike He 🔗 cs.princeton.edu/~dh7120

EDUCATION

 PRINCETON UNIVERSITY, Princeton, NJ

2022 – Est. 2027

Ph.D. in Computer Science

M.A. in Computer Science (conferred in May. 2024)

Advisor: Prof. Aarti Gupta

Fields of study: Programming Languages; Formal Methods; Distributed and Agentic Systems

 University of Washington, Seattle, WA

2018 – 2022

B.S. in Computer Science, GPA: 3.89/4.0 (CUM LAUDE)

Advisors: Prof. Zachary Tatlock & Dr. Steven Lyubomirsky

Selected Honor: CRA Outstanding Undergraduate Researcher Award, Honorable Mention (2022)

PUBLICATIONS & PRE-PRINTS

- **Mike He**, Zhendong Ang, Ankush Desai, and Aarti Gupta. Distilling distributed system specifications with large language models. In *International Workshop on Language Models and Programming Languages (LMPL'25)*, 2025
- **Mike He**, Ankush Desai, Aishwarya Jagarapu, Doug Terry, Sharad Malik, and Aarti Gupta. PINFER: Automatically learning specifications for distributed systems from event traces [*in submission*]
- Akash Gaonkar, **Mike He**, Yi Li, Bo-Yuan Huang, Andrew Cheung, Vishal Canumalla, Gus Smith, Zachary Tatlock, Sharad Malik, and Aarti Gupta. Verification of software-to-hardware mappings for machine learning accelerators [*in submission*]
- **Mike He**, Haichen Dong, Sharad Malik, and Aarti Gupta. Improving term extraction with acyclic constraints. In *E-Graph Research, Applications, Practices, and Human-factors Symposium* [Paper]
- Marisa Kirisame*, Steven Lyubomirsky*, Altan Haan*, Jennifer Brennan, **Mike He**, Jared Roesch, Tianqi Chen, and Zachary Tatlock. Dynamic tensor rematerialization. In *International Conference on Learning Representations (ICLR'21)* [ArXiv]
- Bo-Yuan Huang*, Steven Lyubomirsky*, Thierry Tamba*, Yi Li, **Mike He**, Gus Smith, Gu-Yeon Wei, Aarti Gupta, Sharad Malik, and Zachary Tatlock. From dsls to accelerator-rich platform implementations: Addressing the mapping gap. In *Workshop on Languages, Tools, and Techniques for Accelerator Design (LATTE'21)* [Paper]
- Bo-Yuan Huang*, Steven Lyubomirsky*, Yi Li, **Mike He**, Thierry Tamba, Gus Henry Smith, Akash Gaonkar, Vishal Canumalla, Gu-Yeon Wei, Aarti Gupta, Sharad Malik, and Zachary Tatlock. Application-level validation of accelerator designs using a formal software/hardware interface. *ACM Trans. Des. Autom. Electron. Syst.* [Paper]

INTERNSHIPS

Amazon Web Services, Santa Clara, CA

May. 2025 – Aug. 2025

Applied Scientist Intern (C#/Python/P/Distributed Systems/Formal Methods)

- Led the development of a specification ranking and distillation algorithm for distributed systems (workshop paper accepted at LMPL'25)
- Developed [SOPY](#), a prototype of contract monitoring framework for guaranteeing tool call correctness for multi-agent applications

Amazon Web Services, Santa Clara, CA

May. 2024 – Aug. 2024

Applied Scientist Intern (C#/Java/P/Distributed Systems/Formal Methods)

- Developed the *PInfer* framework that **discovers likely invariants for distributed systems** from event traces
- Implemented a formal model and specifications of the **Raft** consensus algorithm in [P framework](#)
- Modeled and proved the correctness of Ring leader election, Paxos, and an AWS internal service in *PVerifier*

Taichi Graphics, Remote and Beijing, China

June. 2022 – Sep. 2022

Compiler R&D Intern (C++/Python)

- Refactored the intermediate representation (IR) of Taichi Language
- Implemented standalone **Tensor type** for better compilation speed
- Adapted **compiler passes** (e.g. Load/Store forwarding, Dead code elimination, reaching definition, etc.) to optimize for tensor type expressions
- Implemented **LLVM**-based code generation for tensor type for Superword-level vectorization

Intel Labs, Hillsboro, OR

Mar. 2022 – June. 2022

Formal Verification Research Intern (Formal Methods/Python/Dafny)

Developed the **Pyrope** framework for **correct-by-construction** hardware modeling.

- Facilitated **correct-by-construction** hardware modeling purely in Python
- Encoded the correctness proof of (multi-)montgomery reduction algorithm in Python and **verified successfully by compiling to Dafny**
- Unified “sources of truth” for correctness proofs and programming model implementations

👋 SERVICE

- Reviewer: IEEE Transactions on Mobile Computing, AAE’25@KDD, SciPy’25
- Sub-reviewer: OOPSLA’24
- Artifact Evaluation Committee: POPL’25, PLDI’24, POPL’24, MLSys’23, MICRO’21
- Mentor in the Ph.D. application mentoring program (Princeton, 2023)

💡 TEACHING

- [COS 516: Automated Reasoning about Software](#) (TA, Princeton University, Fall 23 & 24)
- [CSE 505: Principles of Programming Languages](#) (TA, University of Washington, Spring 21)

⚙️ SKILLS

- **Languages:** C/C++, Python, Rust, OCaml, Coq, Dafny, etc. (Open to other languages)
- **Compiler & Applied PL:** Equality Saturation, Static Analysis, Computer-aided Reasoning, SMT
- **PL Theory:** Formal Verification, Type Theory, Mathematical Logic
- **Systems:** Distributed Systems, Machine Learning Systems, Data Center Systems
- **Fun Fact:** I am more seasoned in playing the violin than coding 🎻; I have:
 1. an Lv.9 certificate* (similar to [ABRSM](#) Grade 8) issued by [Central Conservative of Music](#);
 2. > **20-year** violin solo experience;
 3. Multiple 1st Prizes (various local competitions in Beijing) and a Silver medal (Beijing regional)[†];
 4. 6-year experience with symphony orchestras; 3-year experience as *the Principal 2nd Violinist*;
 5. 3-year experience with a piano quartet/quintet and multiple string quartets (with 1 CD produced);
 6. ~ 4 public concerts with a philharmonic orchestra* at the [The Giant Egg](#), Beijing, China.

* : *The highest level for non-professionals*

[†] : *Awarded during middle and high school years.*

★ : *The Beijing National Day School Philharmonic Orchestra*