Junrui Liu

junrui@ucsb.edu · Personal Website · Google Scholar · GitHub

Research Interests

Programming Languages, Formal Methods, Software Engineering

Education

2021 – Present **University of California, Santa Barbara** – Santa Barbara, CA
PhD in Computer Science.

2020 – 2021 **Yale University** – New Haven, CT
Master of Science in Computer Science.

2016 – 2020 **Vassar College** – Poughkeepsie, NY
BA in Computer Science (General Honors).

Honors & Awards

2025 **Teaching Assistant of the Year** (UCSB Department of Computer Science)

2024 **Outstanding Teaching Assistant** (UCSB College of Engineering)

2023 **Outstanding Teaching Assistant** (UCSB College of Engineering)

2020 The Janet Holdeen-Adams Prize for Excellence in Computer Science (Vassar College)

2020 Sigma Xi (Vassar College)

2020 Phi Beta Kappa (Vassar College)

Peer-Reviewed Publications

[U] indicates undergraduate mentee co-author

[†] indicates equal contribution

OOPSLA 2025 **Tabby: A Synthesis-Aided Compiler for High-Performance Zero-Knowledge Proof Circuits**
Junrui Liu, Jiaxin Song[U], Yanning Chen[U], Hanzhi Liu, Hongbo Wen, Luke Pearson, Yanju Chen, Yu Feng.
Proceedings of the ACM on Programming Languages, Vol. 9, OOPSLA2, Article 332 (August 2025), 27 pages. https://doi.org/10.1145/3763110.

ASE 2024 **Refinement Types for Visualization**
Junrui Liu[†], Jingtao Xia[†], Nicholas Brown[U], Yanju Chen, and Yu Feng.
Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering (ASE '24). Association for Computing Machinery, New York, NY, USA, 1871–1881. https://doi.org/10.1145/3691620.3695550.

S&P 2024 **Certifying Zero-Knowledge Circuits with Refinement Types**
Junrui Liu, Ian Kretz, Hanzhi Liu[U], Bryan Tan, Jonathan Wang, Yi Sun, Luke Pearson, Anders Miltner, Isil Dillig, Yu Feng.
2024 IEEE Symposium on Security and Privacy (SP '24), San Francisco, CA, USA, pp. 1741-1759. https://doi.org/10.1109/SP54263.2024.00078.

PLDI 2023 **Conflict-Driven Synthesis for Layout Engines**
Junrui Liu, Yanju Chen, Eric Atkinson, Yu Feng, and Rastislav Bodik.
Proceedings of the ACM on Programming Languages, Vol. 7, PLDI, Article 132 (June 2023), 22 pages. https://doi.org/10.1145/3591246.

ASE 2022 **Learning Contract Invariants Using Reinforcement Learning**
Junrui Liu[†], Yanju Chen[†], Bryan Tan, Isil Dillig, and Yu Feng.
Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering (ASE '22). Association for Computing Machinery, New York, NY, USA, Article 63, 1–11. https://doi.org/10.1145/3551349.3556962.

ASPLOS 2022 **Tree Traversal Synthesis Using Domain-Specific Symbolic Compilation**
Yanju Chen, Junrui Liu, Yu Feng, and Rastislav Bodik.
Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS '22). Association for Computing Machinery, New York, NY, USA, 1030–1042. https://doi.org/10.1145/3503222.3507751.

Manuscripts

2022 **A Study of HTTP/2's Server Push Performance Potential**
Rui Meireles, Junrui Liu, Peter Steenkiste.
arXiv manuscript. https://doi.org/10.48550/arXiv.2207.05885

Invited Talks, Workshops & Tutorials

September 2025 Workshop on Leading Computer-based Labs
UCSB Teaching Assistant Orientation

August 2025 Tessel: An Optimizing Compiler for Efficient Zero-Knowledge Circuits
The Science of Blockchain Conference, UC Berkeley

February 2023 Polymorphism, Curry-Howard, and Program Verification
Guest lecture for CS 162: Programming Languages, UCSB

July 2022 Formal Verification for Zero-Knowledge Proofs
Applied ZK Workshop, The Science of Blockchain Conference, Stanford University

July 2022 Introduction to Interactive Theorem Proving in Coq
0xPARC Summer Residency in NYC

February 2022 Refinement Types and Program Verification
Guest lecture for CS 162: Programming Languages, UCSB

Teaching Experience

Fall 2025 — **Instructor of Record**, CS 501: Techniques of Computer Science Teaching (UCSB)
Develop materials on effective teaching techniques, resources for supporting TAs and undergraduates, and university policies. Facilitate weekly seminars and hold office hours. This is the required training course taken by new Computer Science TAs.

Summer 2025 — **Instructor of Record**, CS 162: Programming Languages (UCSB)
Develop new course materials, emphasizing active, discovery learning. Prepare and deliver three 75-minute lectures to a class with 11 students. Hold 4 hours of office hours each week. Manage 1 TA who helps deliver weekly problem sessions and develops autograders for assignments.
Average student rating: **4.94/5**.

Spring 2024 — **Co-Instructor**, CS 292C: Computer-Aided Reasoning for Software (UCSB)
Design and deliver two 1-hour tutorial-style lectures on formal verification each week. Hold 1-hour office hour weekly. Design three programming projects and develop autograders.

Fall 2025 - Present — **Lead Teaching Assistant for Computer Science Department** (UCSB)
Plan and develop Computer Science Department's TA Training program (CS 501), orientation, workshops, and practical materials. Train and evaluate TAs, including observing new TAs and providing constructive feedback, and holding mid-quarter student evaluations. Hold weekly office hours.

Winters 2022-2025 — **Head Teaching Assistant**, CS 162: Programming Languages (UCSB)
Develop new programming projects and exam problems. Plan and deliver 1-hour review sessions and hold 2-hour office hours weekly. Answer students' questions on Slack. Manage 2-3 other TAs.
Average student rating: 4.8/5.

Spring 2025, Spring 2023, Fall 2021 — **Teaching Assistant**, CS 160: Compilers (UCSB)
Plan and deliver 1-hour review sessions and hold 2-hour office hours weekly. Answer students' questions on Slack.
Average student rating: 4.6/5.

Spring 2024 — **Teaching Assistant**, CS 190J: Blockchain Technologies and Security (UCSB)

Fall 2018 — **Grader**, Phil 125: Logic (Vassar College)

Teaching Certificates & Training

Present — **Certificate in College and University Teaching (UCSB)**

Summer 2025 — **Summer Teaching Institute for Associates (UCSB)**

Summer 2025 — **Lead TA Institute (UCSB)**

Mentorship

June 2023 - Present — **Jiaxin Song**, Visiting Undergraduate Research Assistant, UCSB
Research project: *Synthesis-Aided Compiler for High-Performance Zero-Knowledge Proof Circuits*, published in OOPSLA 2025.
Now PhD student at UIUC.

September 2022 - December 2022	**Yanning Chen**, Visiting Undergraduate Research Assistant, UCSB Research project: *Synthesis-Aided Compiler for High-Performance Zero-Knowledge Proof Circuits*, published in OOPSLA 2025. Now PhD student at the University of Toronto.
June 2022 - March 2023	**Hanzhi Liu**, Visiting Undergraduate Research Assistant, UCSB Research project: *Certifying Zero-Knowledge Circuits with Refinement Types*, published in S&P 2024. Now PhD student at UCSB.
March 2022 - December 2022	**Nicholas Brown**, UCSB Undergraduate Research Assistant Research project: *Refinement types for visualization*, published in ASE 2024. Now software engineer at Meta.
Spring 2022	**Surendra Ghentiyala**, UCSB Undergraduate Research Assistant Research project: *Visualization Program Synthesis*. Now PhD student at Cornell University.

Service

2024	**External Reviewer** IEEE Transactions on Software Engineering
2022	**Student Volunteer** Programming Language Design and Implementation (PLDI)

Industry Experience

Summer 2020	**Veridise (Research Scientist)** – New York City, New York Researched and developed the Coda tool for formally verifying zero-knowledge proof circuits using refinement types.