JINGYUAN CHEN

leochanj105.github.io | leocjy@princeton.edu | +19196678835

EDUCATION

Princeton University 2022 - 2027 (expected)
Master & PhD in Computer Science *Princeton, US*
· Advisor: Amit Levy
· Research Interest: Coding AI Agent, Constrained Decoding, Debugging, Formal Methods

University of North Carolina at Chapel Hill 2018 - 2022
Bachelor of Science in Computer Science, Bachelor of Science in Mathematics *Chapel Hill, US*
· Graduated with highest honor, GPA: 3.99/4.0

WORK EXPERIENCES

Google May 2026 - Aug 2026
Ph.D. Software Engineer Intern *Kirkland, US*
Team: Persistent Disk Stub & Live Migration

Developed debugging assistant agents and harness equipped with program analysis skills and tools to pinpoint root causes of real and complex production system bugs.

RESEARCH EXPERIENCES

Princeton University Princeton, NJ
Research Assistant (advised by `Amit Levy`)

Efficient Semantic Constrained Decoding Aug 2026 - Present
Designing an extensible framework for LLM constrained decoding beyond grammar constraints, targeting type correctness, ownership, null safety, and arithmetic constraints. Developing a compiler that generates incremental checkers to derive constraints on future tokens. Using precomputed masks and lightweight mask algebra for efficient token-mask generation. Engineering a runtime that overlaps mask generation with model inference to hide checker latency. Studying how semantic constraints affect output quality.

Lightweight Provenance-Guided Automatic Online Debugging Sep 2022 - Aug 2026
Built an *online* debugger that dynamically instruments tracepoints to record data-flow provenances for latent bugs in deployed distributed systems. Designed library-bypassed static analysis algorithms for identifying bug-relevant variables that overcame the *fundamental* precision-scalability trade-off of traditional approaches. Engineered a lightweight, synchronization-free tracing runtime for recording shared-state provenances utilizing the coarse-interleaving hypothesis. Evaluated the system on complex bugs in a realistic distributed system (HDFS) and demonstrated its capability to minimize diagnosis latencies while incurring *practical* runtime overheads ($\leq$10% throughput reduction).

Semantic-Aware Testing for Transpilation via LLM Agents Nov 2025 - Present
Developed a framework for semantics-aware test generation to check the correctness of C-to-Rust library transpilation using LLM agents. Designed an LLM-based semantic summarization layer to reconcile interface-level semantic mismatches between source and target libraries. Employed the extracted semantics to synthesize correct test harnesses and test cases for automated transpilation validation.

Auditing Resource Interference-Freedom of Cloud Platforms Nov 2024 - Present
Identified a new class of threat in cloud platforms where providers violate resource-isolation guarantees. Designed a mechanism to detect shared-resource side-channel interference using compiler-instrumented

auditing operations. Currently developping a *program enhancer* that automatically injects lightweight auditing operations to make victim cloud workloads interference-awareness.

Critical Path Analysis for Distributed Tracing under Asynchrony Oct 2025 - Present
Developed extensions to OpenTelemetry Java enabling context propagation and span linkage across asynchronous tasks. Implemented enhanced critical-path analysis algorithms that capture dependencies across async invocations. Evaluated the system on real-world trace datasets (CRISP, Tale of Errors) and the TrainTicket benchmark, demonstrating both the prevalence of inaccuracies in asynchrony-unaware analyses and the effectiveness of our approach in correctly identifying performance bottlenecks.

University of North Carolina at Chapel Hill Chapel Hill, NC
Research Assistant (advised by `James H. Anderson` and `F. Don Smith`)

Making Powerful Enemies on NVIDIA GPUs Aug 2021 - Aug 2022
Empirically evaluated the sensitivity of common GPU workloads to a wide range of interference channels in NVIDIA GPUs. Engineered "enemy" GPU kernels that maximize contention over hardware resources for approximating the worst-case execution times of real-time GPU kernels. Evaluated the enemies against real-world kernels and showed the effectiveness of the enemies in maximizing resource contention.

PREPRINTS

1. **Jingyuan Chen**, Lei Zhang, Leon Schuermann, Gongqi Huang, Ravi Netravali, Amit Levy. *Wherefore Art Thou? Provenance-Guided Automatic Online Debugging with Lumos.* **arXiv preprint, 2026**

PUBLICATIONS

1. Julia Cusatis, **Jingyuan Chen**, and Timothy Zhu. *Two Parents Are Better Than One*: Enhancing Critical Path Analysis for Distributed Tracing with CPA2. **SoCC '26**.
2. **Jingyuan Chen**, Amit Levy. *Progress-Guided Test Generation for C-to-Rust Transpilation: Effectiveness and Limitations.* **IEEE Security & Privacy Magazine**, Special Issue on Language Translation for Security, 2026
3. **Jingyuan Chen**, Gongqi Huang, Amit Levy. *Towards Agentic Performance Management.* **AgenticOS '26**
4. Natalie Popescu, Shai Caspin, Leon Schuermann, **Jingyuan Chen**, Amit Levy. *End-to-End Encrypted Applications with Strong Consistency Under Byzantine Actors.* **ACSAC '25**
5. Gongqi Huang, **Jingyuan Chen**, Amit Levy *Incentivizing Side-Channel Freedom with Leakage-Aware Computation.* **WACI '25**
6. Tyler Yandrofski, **Jingyuan Chen**, Nathan Otterness, James H. Anderson and F. Donelson Smith. *Making Powerful Enemies on NVIDIA GPUs.* **IEEE RTSS'22**
7. Sims Hill Osborne, Joshua Bakita, **Jingyuan Chen**, Tyler Yandrofski, and James H. Anderson. *Minimizing DAG Utilization by Exploiting SMT.* **IEEE RTAS'22**
8. Joshua Bakita, Shareef Ahmed, Sims Hill Osborne, Stephen Tang, **Jingyuan Chen**, F. Donelson Smith, and James H. Anderson. *Simultaneous Multithreading in Mixed-Criticality Real-Time Systems.* **IEEE RTAS'21**

TECHNICAL SKILLS

Programming Languages | Proficient: Java, C, Python, Go, Javascript, Datalog; Prior Experience: C++, Rust, Haskell, Common Lisp, OCaml, Prolog
Tools & Systems | Proficient: Linux, bash, Git, Vim, Spring, Docker, Distributed Tracing (Jaeger,

OpenTelemetry), Soot, Doop, HDFS; Prior Experience: XGrammar, vLLM, Coq, Dafny, TensorFlow, PyTorch, CUDA