Yazhou Tang

tangyazhou@zju.edu.cn | ADSWT518 | tangda.me | yazhoutang

Education

Zhejiang University 2025/09 — 2028/06 (expected)
Master of Engineering in Computer Techonology Hangzhou, Zhejiang, China
- Member of ZJU Formal Verification Group in ZJU Advanced Computing and Emerging Software Lab
- Supervised by Prof. Mingshuai Chen and co-advised by Prof. Shenghao Yuan.

Shanghai Jiao Tong University 2019/09 — 2023/06
Bachelor of Engineering in Computer Science and Technology Shangehai, China
- SJTU Outstanding Graduate (Top 15%)
- Courses: Programming Languages, Program Analysis & Verification, Operating Systems, Compiler Principles, ...

Research Experience

End-to-end Verification of in-kernel Verifier based Proof-Carrying Code 2025/06 — Now
ZJU State Key Lab of Blockchain and Data Security, advised by Prof. Shenghao Yuan Hangzhou, Zhejiang, China
- This project formalized the Tristate Number and Wrapped Interval abstract domains in Linux kernel's eBPF verifier using Rocq Prover, proving soundness of their core arithmetic operations.
- It can enhance eBPF security through a proof-carrying code framework implemented in the Linux kernel, consisting of a formal certifier and runtime checker.

Modular Verification by Integrating LLMs with Symbolic Reasoning 2025/06 — Now
ZJU Formal Verification Group, advised by Prof. Mingshuai Chen Hangzhou, Zhejiang, China
- This project develops a lightweight verification framework combining LLMs and symbolic reasoning (static analysis, theorem proving) to ensure OS kernel memory safety, targeting challenges like modular specification and refinement for systems like OpenHarmony.
- It first prioritizes code segments using static analyzers (Frama-C) and memory-safety assertions, then employs LLMs to generate and iteratively refine specifications (pre/post-conditions) via theorem prover feedback, eliminating false negatives.
- The prototype will be tested on OpenHarmony and Linux drivers, evaluating its handling of complex data structures (e.g., nested pointers) and cross-system adaptability for industrial-scale verification.

VST-IDE: An Interacitve Tool for Program Verification [Thesis Paper] [Slides] 2021/10 — 2023/06
SJTU Undergraduate Research, advised by Prof. Qinxiang Cao Shanghai, China
- Integrated the Verified Software Toolchain PrincetonUniversity/VST into the C development environment, allows users to achieve automated memory safety verification simply by writing function pre/postconditions and assertions in annotations.
- Based on the Consequence rule of Hoare logic, automated proofs are realized through alternating symbolic execution and assertion checking. Custom predicates enable the verification of data structures such as singly-linked lists and doubly-linked lists, while modular composition ensures scalability.
- The assertion checking module I developed is a separation logic-based SMT solver. Its theoretical foundation is the Frame rule of separation logic, which verifies whether the entailment relationship holds between the program state after symbolic execution and user assertions through operations like predicate splitting.
- The **QIDE Project** developed based on this project has been publicly released. [Online IDE]

Verification of Redblack Tree's Source Code Safety [Paper] [Slides] 2021/01 — 2021/09
SJTU Undergraduate Research, advised by Prof. Qinxiang Cao Shanghai, China
- Conducted formal verification of a C-implemented red-black tree (including insertion, deletion, and lookup operations) using Rocq Prover and VST, proving its behavioral consistency with the abstract map specification.

... continues on the next page ...

- Designed auxiliary data structures such as “half-tree” and “partial tree” to formally represent parent pointers in red-black tree nodes, resolving cyclic dependencies in the verification process.
- Proved the correctness of the “lookup-by-key and return value address” operation by decoupling tree nodes from stored values in the formal memory model, establishing equivalence between implementation-level and specification-level behaviors.

Work Experience

Bilibili Inc. 2023/07 — 2024/10

Multimedia Software Engineer (Cross-platform Technology Department, Playback Quality Team) Shanghai, China

- Participated in core module development of bilibili/ijkplayer (33k Stars), customizing FFmpeg components for audio/video processing.
- Optimized playback engine through performance tuning and A/B testing, boosting user experience for VOD/live streaming scenarios on mobile platforms.

Key Achievements:

- **AV1 Hardware Decoding Solution for Mobile Live Streaming**
  - Developed hardware abstraction layer for AV1 decoding (Android MediaCodec / iOS VideoToolbox) with adaptive device-tiering compatibility.
  - Collaborated with MediaTek/Xiaomi engineers on codec-level optimizations, resolving decoding issues across 10+ mainstream device models.
  - Enabled full AV1 live streaming pipeline on mobile, reducing bandwidth consumption by 30% vs. H.265 solutions.
- **Intelligent Audio Normalization System**
  - Architected dynamic loudness control leveraging FFmpeg audio filters (EBU R128 standard) with scenario-specific adjustment profiles.
  - Optimized audio sampling window from 3s to 400ms initial load, slashing 85% first-frame processing latency.
  - Rolled out feature to 100% mobile VOD users, reducing audio-related user complaints by 70%.

Competitions

2nd Prize, CCF ChinaSoft Theorem Proving Competition. [Official Website] 2025/11

2nd Prize, SJTU Android App Development Competition. NaiveGator 2021/12

Excellence Award, World Robot Contest Finals (WRCF) VEX-U Track. [SJTU Official Report] 2020/12

Honors & Awards

Outstanding Graduate Award, Shanghai Jiao Tong University 2023/06

Academic Improvement Scholarship, Shanghai Jiao Tong University 2022/12

Third-Class Academic Scholarship, Shanghai Jiao Tong University 2020/12

Triple-A Outstanding Award, Shanghai Jiao Tong University 2020/11

Skills

- **Programming Languages**: Coq (Rocq), OCaml, C/C++.
- **Formal Methods**: Programming Language Theory (PLT), Formal Verification, Interactive Theorem Proving.
- **Systems Engineering**: Linux Kernel Development, eBPF.

Miscellaneous

- **Language**: Chinese (Native), English (CET-6)
- **Hobbies**: Football, Running, Table Tennis, Badminton, Tennis.

Last Updated on December 25, 2025