Qiancheng Fu

qcfu@bu.edu
https://github.com/qcfu-bu

Education

2019 – present **Ph.D. Computer Science**, Boston University, (Thesis Advisor: Hongwei Xi)

2019 – 2021 **M.Sc. Computer Science**, Boston University

2015 – 2019 **B.Eng. Computer Science**, North China University of Technology

Research Publications

Conference Proceedings

1 Q. Fu and H. Xi, "A calculus of inductive linear constructions," in *Proceedings of the 8th ACM SIGPLAN International Workshop on Type-Driven Development*, ser. TyDe 2023, Seattle, WA, USA: Association for Computing Machinery, 2023, pp. 1–13, ISBN: 9798400702990. DOI: 10.1145/3609027.3609404.

2 M. Lemay, Q. Fu, W. Blair, C. Zhang, and H. Xi, "A dependently typed language with dynamic equality," in *Proceedings of the 8th ACM SIGPLAN International Workshop on Type-Driven Development*, ser. TyDe 2023, Seattle, WA, USA: Association for Computing Machinery, 2023, pp. 44–57, ISBN: 9798400702990. DOI: 10.1145/3609027.3609407.

Journal Articles

1 Q. Fu and H. Xi, "A two-level linear dependent type theory," 2023. arXiv: 2309.08673 [cs.PL]. URL: https://arxiv.org/abs/2309.08673.

2 Y.-T. Sun, Q.-C. Fu, Y.-R. Jiang, *et al.*, "Human motion transfer with 3d constraints and detail enhancement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–12, 2022. DOI: 10.1109/TPAMI.2022.3201904.

Research Projects

- **PReST** (accepted to PLDI25): We develop a novel theory of probabilistic refinement session types (PReST) to *symbolically* specify and reason about probabilistic message passing concurrent programs. The soundness of the PReST type system ensures that the probabilistic distributions specified in communication protocols are respected at runtime. Most surprisingly, probabilistic refinement types can even be used to statically verify parametric distributions such as the uniform distribution over $\{0, ..., k\}$. Using PReST we are able to specify probabilistic distributed protocols such as Leader Election, Bounded Retransmission and Crowd Forwarding. We implement a type checker in OCaml using Z3 and CVC5 to efficiently solve complex generated constraints.
- **On-the-fly GKAT**: The theory of Kleene Algebra with Tests (KAT) allows one to decide the equivalence of imperative programs using an elegant equational theory. The theory of Guarded Kleene Algebra with Tests (GKAT) promises to improve the performance of KAT by restricting its scope to the deterministic fragment of KAT. In practice, however, GKAT decision algorithms tend to be even slower than KAT algorithms due to the necessity of performing additional normalization steps. To solve the issue of using GKAT in practice, we introduce a novel *on-the-fly* algorithm for GKAT which performs bisimulation in a greedy manner and defers normalization to when it is absolutely necessary. We develop the `rust-gkat` tool in Rust. Through experiments, we show that our tool performs orders of magnitudes faster than existing KAT solvers.

Research Projects (continued)

- **TLL**: The Two-Level Linear (TLL) dependent type theory is a programming language which combines dependent types and linear types. TLL features full Martin-Löf style dependent types to precisely reason about linearly typed programs. We fully verify the theory of TLL in Coq. A prototype optimizing compiler is implemented. It emits safe C code that is memory clean without need of runtime garbage collection. We also extend the compiler with features such as dependent session types for concurrency and *sort-polymorphic schemes* to write code generically for both linear and unrestricted types.
- **CILC**: The Calculus of Inductive Linear Constructions (CILC) is a linear dependent type theory which fully formalizes the mechanism for defining linear dependent inductive types. Linear dependent inductive types allow for the language runtime to perform mutation on data which appears to be immutable from the user's perspective. The entire theory of CILC is formalized in Coq and proven to be sound. CILC is one of the few dependent inductive type systems ever formalized in a theorem prover and, to the best of our knowledge, is also the only formalized linear dependent inductive type system.
- **MT/DE-Net**: We propose a pipeline for retargeting character motion in videos by using 2 novel generative adversarial networks MT-Net and DE-Net. First, MT-Net uses of the projection of 3D human models to maintain the structural integrity of character poses during motion transfer. Next, DE-Net enhances details in generated frames by reusing the details in real source frames. This essentially eliminates the need for generating reusable details from scratch. Extensive experiments show that our approach yields better results both qualitatively and quantitatively than the state-of-the-art methods.

Talks

fall 2022	**A Calculus of Linear Constructions**. Boston University POPV Seminar
summer 2023	**A Calculus of Inductive Linear Constructions**. TyDe 2023, Seattle, WA
fall 2023	**A Two-level Linear Dependent Type Theory**. Boston University POPV Seminar
	Linear Dependent Types in Practical Programming. Harvard University PL Seminar
fall 2024	**Probabilistic Refinement Session Types**. Boston University POPV Seminar

Work Experience

fall 2022	**Instructor**. Boston University, Principles of Programming Languages.
2020 – 2024	**Teaching Fellow**. Boston University, Principles of Programming Languages.
2018 – 2019	**Research Assistant**. Institute of Computing, Chinese Academy of Sciences.

Skills

programming	OCaml, Rust, Haskell, ATS, C/C++/CUDA
proof assistants	Coq, Lean4, EasyCrypt, Agda
SMT solvers	Z3, CVC5, Alt-ergo
misc.	Type systems (dependent, linear, session), formal analysis of programming languages, compiler construction and optimization.