

# Keyur Parag Joshi

Room 3107, Thomas M. Siebel Center for Computer Science  
University of Illinois Urbana-Champaign, USA

Email: [kpjoshi2@illinois.edu](mailto:kpjoshi2@illinois.edu) Phone: +1 217-904-6096 Website: [kpjoshi.com](http://kpjoshi.com)

## Education

- **University of Illinois Urbana-Champaign (UIUC), USA**  
Ph.D in Computer Science advised by Associate Professor Sasa Misailovic  
August 2017 – December 2023 (Expected)
- **Indian Institute of Technology, Hyderabad (IITH), India**  
Bachelor of Technology (Honours) in Computer Science and Engineering; Valedictorian  
August 2013 – May 2017

## Research Interests

- Testing and analysis of approximate and/or unreliable software and systems
- Effective and safe application of approximations in emerging domains

## Current Research

- **Efficient analysis of program uncertainty in the face of SDCs:** Unlike easily detectable crashes and out-of-bounds outputs, Silent Data Corruptions (SDCs) incorrectly alter program data in an insidious manner. We explore the combination of tools for analyzing the effects of injected errors on programs (including SDCs) with tools for analyzing the propagation of error through programs. This approach allows for compositional analysis of software pipelines, reducing the need for re-running expensive error injection analyses during development.
- **Surrogate models for autonomous vehicle systems:** Modern autonomous vehicles use neural networks and other complex components to perceive the environment and/or to make control decisions. Simulating these systems to ensure they do not violate safety properties is costly. Our novel approach uses Generalized Polynomial Chaos (GPC) to create surrogate models of the complete vehicle system (perception, control, and dynamics). We use these surrogate models to efficiently estimate the probability that the vehicle violates a safety property over time, and for sensitivity analysis of the vehicle system.

## Publications

Publications can also be found at [scholar.google.com/citations?user=ewi6R3UAAAAJ](https://scholar.google.com/citations?user=ewi6R3UAAAAJ) and [kpjoshi.com](http://kpjoshi.com)

- **Verifying Controllers with Vision-based Perception Using Safe Approximate Abstractions**  
Chiao Hsieh, Yangge Li, Dawei Sun, **Keyur Joshi**, Sasa Misailovic, Sayan Mitra  
*Embedded Software (EMSOFT 2022)*
- **Diamont: Dynamic Monitoring of Uncertainty for Distributed Asynchronous Programs**  
Vimuth Fernando, **Keyur Joshi**, Jacob Laurel, Sasa Misailovic  
*International Conference on Runtime Verification (RV 2021)*
- **ApproxTuner: A Compiler and Runtime System for Adaptive Approximations**  
Hashim Sharif, Maria Kotsifakou, Yifan Zhao, Akash Kothari, Ben Schreiber, Elizabeth Wang, Yasmin Sarita, Nathan Zhao, **Keyur Joshi**, Vikram Adve, Sasa Misailovic, Sarita Adve  
*ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPOPP 2021)*
- **Aloe: Verifying Reliability of Approximate Programs in the Presence of Recovery Mechanisms**  
**Keyur Joshi**, Vimuth Fernando, Sasa Misailovic  
*IEEE/ACM International Symposium on Code Generation and Optimization (CGO 2020)*
- **Statistical Algorithmic Profiling for Randomized Approximate Programs**  
**Keyur Joshi**, Vimuth Fernando, Sasa Misailovic  
*41st ACM/IEEE International Conference on Software Engineering (ICSE 2019)*

- **Verifying Safety and Accuracy of Approximate Parallel Programs via Canonical Sequentialization**  
Vimuth Fernando, **Keyur Joshi**, Sasa Misailovic  
*34th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA/SPLASH 2019)*
- **ApproxHPVM: A Portable Compiler IR for Accuracy-Aware Optimizations**  
Hashim Sharif, Prakash Srivastava, Muhammad Huzaifa, Maria Kotsifakou, **Keyur Joshi**, Yasmin Sarita, Nathan Zhao, Vikram S. Adve, Sasa Misailovic, Sarita Adve  
*34th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA/SPLASH 2019)*
- **Identifying Optimal Parameters for Randomized Approximate Algorithms**  
Vimuth Fernando, **Keyur Joshi**, Darko Marinov, Sasa Misailovic  
*Workshop on Approximate Computing Across the Stack (WAX 2019) (Co-located with PLDI 2019)*

## Open-Source Tools

- **Fixed-Point-RNN-Training**: A tool for training RNNs using (almost) exclusively fixed-point arithmetic, available at [github.com/KPJoshi/Fixed-Point-RNN-Training](https://github.com/KPJoshi/Fixed-Point-RNN-Training)
- **Parallely**: Verifying Safety and Accuracy of Approximate Parallel Programs via Canonical Sequentialization: tool and instructions available at [github.com/uiuc-arc/parallely](https://github.com/uiuc-arc/parallely)
- **AxProf**: Statistical Algorithmic Profiling for Randomized Approximate Programs: tool, examples, and tutorial available at [axprof.org](https://axprof.org)

## Professional Experiences

- Summer 2022 – Research Intern at Microsoft Research India
- Spring 2022 – Member of the PLDI 2022 Artifact Evaluation Committee
- Summer 2021 – Member of the OOPSLA 2021 Artifact Evaluation Committee
- Fall 2020 – Teaching Assistant for CS 427 Software Engineering I at UIUC; responsibilities including office hours, class logistics, homework, and projects
- Spring 2018 – Co-organized the Brett Daniel Software Engineering Seminar at UIUC

## Talks and Presentations

- **Conference Talk**: Aloe: Verifying Reliability of Approximate Programs in the Presence of Recovery Mechanisms (*CGO 2020*)
- **Conference Talk**: Statistical Algorithmic Profiling for Randomized Approximate Programs (*ICSE 2019*)
- **Seminar Talk**: Monitor-Based Statistical Model Checking for Weighted Metric Temporal Logic (*Brett Daniel Software Engineering Seminar, UIUC*)
- **Lightning Talk**: Implementation of a Cache Miss Calculator in LLVM/Polly (*LLVM in HPC Workshop, SC 2017*)
- **Seminar Talk**: Triangular inequality for compiler-based strength reduction (*Brett Daniel Software Engineering Seminar, UIUC*)

## Skills

- **Programming**: Extensive experience in programming with Python and C/C++.
- **Teamwork**: Extensive experience collaborating on research and engineering projects in teams of 2-5 individuals using tools such as git, Matplotlib, Slack, and PowerPoint.
- **Documentation**: Experience writing documentation for developers as well as end users.