

Shreyas Kharbanda

Department of Computer Science, Cornell University
sk2936@cornell.edu • <https://alphacode18.github.io>

RESEARCH INTERESTS

Operating Systems, Computer Architecture, Distributed Systems, Computer Networks

EDUCATION

Cornell University

Ph.D. in Computer Science

Advisor: [Prof. Rachit Agarwal](#)

Ithaca, NY

2023 – Present

Purdue University

B.S. in Computer Science

Advisor: [Prof. Pedro Fonseca](#)

GPA: 3.97/4.00

2019 – 2023

PUBLICATIONS

- ***Pronghorn: Effective Checkpoint Orchestration for Serverless Hot-Starts***
Sumer Kohli*, [Shreyas Kharbanda*](#), Rodrigo Bruno, Joao Carreira, and Pedro Fonseca.
In the 19th USENIX European Conference on Computer Systems (EuroSys), Athens, Greece, 2024.
- **Always-on Recording Framework for Serverless Computations: Opportunities and Challenges**
[Shreyas Kharbanda](#) and [Pedro Fonseca](#)
In the 1st Workshop on Serverless Systems, Applications and Methodologies (SESAME), Rome, Italy, 2023.

[* denotes equal contribution]

SELECTED RESEARCH PROJECTS

Improving Serverless Cloud Reliability with Intelligent Record-Replay Aug. 2022 – May 2023
Conducted an in-depth study on the overheads of an always-on recording framework designed to capture the fine-grained details of serverless computations. The objective was to identify opportunities to refine traditional record-replay techniques to meet the performance and scalability requirements of serverless computing. Through this project, we aimed to improve the visibility of serverless applications, simplify fault diagnosis, facilitate performance analysis, and enhance security measures. Additionally, we outlined potential future advancements in cost-effective recording, storage, and practical replay analysis tools.

Leveraging Language Runtimes for Serverless Hot-Starts Aug. 2021 – Sept. 2023
Designed Pronghorn, a snapshot orchestrator for serverless computing, which automated the monitoring of function performance and the decision-making process for taking and applying snapshots to new workers. This project aimed to realize the missed performance gains due to the loss of learned optimizations in modern JIT-based language runtimes upon worker evictions. Our approach enabled aggregating code profiles across serverless runtimes, ensuring continuous learning and rapid convergence to the most performant snapshot while being resilient to variance in function input sizes. As a result, Pronghorn demonstrated substantial improvements, with a median latency reduction of 37.2% across 9 out of 13 representative serverless benchmarks, significantly enhancing overall serverless performance compared to state-of-the-art checkpointing policies.

HONORS & AWARDS

- Received an **Honorable Mention** from the Computing Research Association for the 2023 North American Outstanding Undergraduate Research Awards, 2022
- Received a **Corporate Partners Scholarship** from Purdue CS, 2020
- Received **Dean's List** from Purdue, 2023, 2022, 2021, 2020, 2019
- Received **Semester Honors** from Purdue, 2023, 2022, 2021, 2020, 2019

WORK EXPERIENCE

Purdue University

2021 – 2023

Undergraduate Teaching Assistant

West Lafayette, IN

- CS 35100 Cloud Computing – Spring 2023
- CS 35400 Operating Systems – Fall 2022
- CS 24000 Programming in C – Fall 2021, Spring 2022

PRESENTATIONS

- *Enabling Low-Overhead Time-Travel Debugging in Serverless Workloads*, Reliable and Secure Systems Lab Seminar, Purdue University, 2022.
- *Checkpoint-Restore Orchestration of Processes for Serverless Hot-Starts*, Graduate Systems Research Seminar, Purdue University, 2021.

OPEN-SOURCE PROJECTS

Localr

Oct. 2020 – Jan. 2021

- Developed an automated, end-to-end platform to run, process, and visualize function performance for runtime code optimization analysis.
- Implemented configurable load-testing tool to experiment with different invocation patterns, time scales and failure rates.
- Integrated function suite with locally sandboxed version of AWS Lambda to simulate behavior in real-world cloud-provider environment.
- Tested the robustness and scalability of the pipeline with 1,000,000+ invocations across benchmarks in the popular benchmarking suites like SeBS and FaaSdom.