Sowmith Kunapaneni

sowmith.kunapaneni@wsu.edu | sowmithk.com | Pullman, WA

Education

Washington State University — Pullman, WA
PhD in Computer Science — *Aug 2024 – Expected May 2028*
Relevant Courses: ML, GPU Programming, Database Systems, AI, Programming Languages, Computational Genomics

Velagapudi Ramakrishna Siddhartha Engineering College — Vijayawada, India
Bachelor's in Electronics and Communication Engineering — *Aug 2016 – Sept 2020*

Experience

Graduate Research Assistant — Aug 2024 – Present
HARP Lab, Washington State University — *Pullman, WA*
- Advised by Dr.Thomas Gilray, working on HPC and PL research.
- Currently working on a GPU accelerated graph and program analytics engine.
- Papers under review at ASPLOS-26(Additional parallelism in Distributed Joins), VLDB-26(Accelerating Iterative Joins on GPU)

Graduate Research Assistant — Aug 2023 – Aug 2024
HARP Lab, University of Alabama at Birmingham — *Birmingham, AL*
- Research assistant for Programming Languages and HPC research.
- Learning about Control Flow Analysis, Functional Data Structures and **Parallel Datalog engines**.
- Worked on Slog Lang; A MPI based shared memory deductive database engine, paper accepted at ***VLDB 2025***.

Systems Engineer — Oct 2020 – Dec 2022
Tata Consultancy Services — *Hyderabad, India*
- **Built internal CLI tooling** to improve support workflow and automate tasks.
- Implemented log management and monitoring tools, reducing failure rates and improving SLA.
- Developed comprehensive dashboards for tracking and monitoring job health, leveraging logs and various job-specific parameters.

Projects

Analytics on the GPU | A GPU Datalog Engine | *C++, CMake, CUDA* — July 2024 – Present
- Built a **GPU Analytics engine** that supports relational algebra for graph and program analytics queries.
- Uses C++ Templating coupled with CUDA and specialized data structures; improving over existing engines by over 5x across queries and benchmarks.
- Builds on top of the GPU optimized hash based indexing structure from GDlog adding support for optimal joins and vectorized execution.

Brouhaha's Compiler & Run-Time | *C++, Racket, CMake* — May 2023 – Nov 2023
- A full program analyzing compiler for sub-set of Racket
- Wrote a **C++ Run-Time** to support primitives and data structures
- Integrated Boehm Garbage Collector and GNU MP Big Num Lib into the Run-Time
- Worked on a Hash Array Mapped Trie implementation to support 'hash' and 'set' prims.
- Wrote a tree-based index over a hash table to **enable efficient query of analysis** results by the compiler.
- https://github.com/harp-lab/brouhaha

Multi Threaded Maze Game Server | *C++, Python* — Aug 2023 – Dec 2023
- Renders frames of a 2D Maze-game simulation, communicating with an agent over stdin/out
- Implemented Radial Sweep for agent vision, multi-agent support, and game object interactions.
- Wrote reference agents that use A*, D*Lite and Flood Fill algorithms.
- https://github.com/harp-lab/maze-game

Technical Skills

Programming Languages: C++, C, Python, CUDA, Racket, Go, Datalog
Developer Tools:Perf, Git, Docker, Clang, Linux, Unix Core Utils, LSP, Vim, CLion, Tree-sitter, Gem5