SANA DAMANI

sdamani@gatech.edu
(+1) 678-365-1962
www.linkedin.com/in/sanadamani
www.sanadamani.com

Research Interest

- Compiler optimizations for parallel architectures

Skills

- Compiler optimizations
- GPU architecture
- Program performance analysis
- Familiar with LLVM, MLIR
- Programming in C, C++

Education

Georgia Institute of Technology

PhD candidate, Computer Science — Aug 2017 – Present

- Advisor: Professor Vivek Sarkar
- Projects:
  - Run-time branch divergence detection for optimal per-warp branch handling in the LLVM-based HARP compiler for the SIMD HARMONICA architecture under the advisement of Professor Hyesoon Kim
  - Implemented and improved on the path-sensitive analysis to reduce def-use chains as described in “Refining Data Flow Information Using Infeasible Paths” by Bodik et al.
  - Implemented the path-sensitive constant propagation optimization as described in “Comprehensive Path-Sensitive Data-flow Analysis” by Thakur and Govindarajan in LLVM
  - Used machine learning techniques to study the impact of election policies and voting infrastructure on voter turnout
- Courses:
  - Compiler Design, High Performance Computer Architecture, Parallelizing Compilers, Machine Learning

University of Pune

Bachelor’s degree in Computer Engineering — July 2008 - May 2012

Research

Publications

- Common Subexpression Convergence: A New Code Optimization for SIMT processors, LCPC 2019
- Speculative Reconvergence for Improved SIMT Efficiency, CGO 2020

Miscellaneous

- Developed and co-presented a tutorial on MLIR at the Workshop on MLIR for HPC, 2019
- Awarded the NVIDIA Graduate Fellowship, 2021-2022

Work Experience

Nvidia Research

Graduate Intern, Architecture Research, 2020

- Designed, implemented, and evaluated a hardware feature to improve warp latency of divergent programs on Nvidia GPUs. This included making changes to the in-house simulator code written in C++.

Intel

Graduate Intern, TensorFlow, 2019

- Enabled quantization-aware training on object detection programs in the TensorFlow toolchain for improved accuracy of high-performance, quantized inference.
- Developed a prototype MLIR dialect for quantization related optimizations.
- Designed a toolchain for integration of Intel's TensorFlow tools into the MLIR compiler.

NVIDIA

Graduate Intern, GPU Compilers, 2018

- Designed and implemented a compiler optimization that improved SIMT efficiency and run-time for programs with divergent execution on Volta GPUs.

Senior System Software Engineer, GPU Compiler, 2012 – 2017

- Provided compiler back-end support for Nintendo Switch and collaborated with customers and application developers to help analyze programs and improve performance and compilation time for key games and applications.
- Led a cross-team design for an optimization to eliminate redundant stores.
- Analyzed compile time bottlenecks and reduced the back-end compile time for shaders by a maximum of 45% and an average of 10% across the board.
- Analyzed performance characteristics of key benchmarks to discover optimization opportunities.
- Designed and implemented a redundant conversion elimination optimization.
- Enhanced loop unrolling to improve the performance of key games by over 20%.
- Designed and implemented a code hoisting pass to improve performance of programs with long latency instructions.
- Simplified the handling of instruction dependencies within the compiler.
- Enhanced instruction scheduling and register allocation heuristics to meet performance goals for mobile benchmarks.
- Implemented an early exit optimization that showed a 5% performance improvement in key benchmarks.

Undergraduate Intern, GPU Compiler, 2011 - 2012

- Implemented a graph-coloring based register allocator for LLVM to perform a comparative study against the greedy linear scan register allocation algorithm.
- Designed and implemented a data flow analysis-based register promotion pass in the LLVM compiler based on "A New Algorithm for Scalar Register Promotion Based on SSA Form" by A.V.S. Sastry, Roy D.C. Ju.