VIKRANT GAJRIA Email: vikrantgajria@gmail.com

EDUCATION

Purdue University, USA *Master of Science, Computer Science* GPA: 4.00 Aug. '21 - May '23
University of Mumbai, India *Bachelor of Engineering, Computer Engineering* GPA: 3.90 Aug. '17 - May '21

RELEVANT WORK EXPERIENCE

Compiler R&D Intern May '22 - Aug. '22
Synopsys, Mountain View

- Researched utility of AI models in LLVM middle-end to reduce compilation time.
- Reduced compilation time of SystemVerilog benchmarks by upto 20% without degradation of runtime performance by finding optimal pass pipelines using Genetic Algorithms.
- Developed analysis passes to dump LLVM function features for ML model training.
- Trained and implemented XGBoost model to predict best optimization level for given functions.
- Extended pass manager to use supervised-learning heuristics for customization of pass pipelines per function.

SELECTED PROJECTS **github.com/vixrant**

OS kernel modifications for Xinu Aug. '22 - Dec. '22

- Added isolation of user-mode processes by implemnting trapped system calls using interrupts in x86.
- Implemented multilevel feedback queue for O(1) time-shared and real-time process scheduling.
- Designed virtual memory frame management with O(1) operations for fast page fault servicing.
- Implemented paging and virtual memory heap memory management from scratch on 32-bit x86 platform.

GPU acceleration of k-SAT solving Jan. '22 - Present

- Studied problem decomposition to accelerate DPLL SAT solving algorithm on SIMT architectures.
- Developed C++ OpenMP kernels to eliminate boolean clauses from CNF expression using unit propagation.
- Improved runtime performance over serial algorithm by upto 72% with 32 threads on Intel Xeon 32-core CPU.

Memory profiling and characteristic analysis of CUDA programs Jan. '22 - May '22

- Developed instrumentation programs using NVBit in C++ to log memory addresses accessed by CUDA kernel.
- Modeled runtime load-store relationship of memory accesses as a directed graph for analysis in Python.
- Studied the relationship graph with plotting libraries and identified instructions that produce frequently used data using heavy-tail distribution.

LLVM frontend and optimization passes for subset of C language Aug. '21 - Dec. '21

- Translated source code to LLVM SSA translation for language specification using recursive-descent parser.
- Developed LLVM passes for constant folding, dead block elimination, and loop invariant code motion.
- Studied and implemented Chaitin-Brigg's algorithm for register allocation and spilling.

Compiler and virtual machine for IEC 61131 Oct. '20 - May '21

- Developed two-phase compiler for "Structured Text" language to a stack machine using C++ and ANTLR4.
- Engineered a stack-based bytecode interpreter based on Pascal-P ISA in C.
- Automated testing of code generator with hand-written examples using Catch2.

Video hand gesture recognition using 3D point cloud data Aug. '20 - May '21

- Achieved 96.4% accuracy of video gesture recognition using ResNet18 and 1D CNN in Tensorflow.
- Developed Tensorflow computation graph functions for GPU-accelerated preprocessing of input data.
- Deployed input video input recording and model inference workers as producer-consumer threads in Python.

TEACHING EXPERIENCE

Parallel Computing CS525, Purdue University Jan. '23 - May. '23
Computer Architecture CS250, Purdue University Aug. '22 - Dec. '22

TECHNICAL SKILLS

Languages: C++, C, Python, ARM64, x86, Coq
Tools: LLVM, CMake, Flex, Bison, Boost, OpenMPI, OpenMP, Pthreads, CUDA, Unix, Git, Tensorflow, CI/ CD
Domain knowledge: Compilers, Parallel Computing, Machine Learning, Data Analysis, Test Driven Development