Andy Wanna

404-660-0194 — awanna3@gatech.edu — Web Page — LinkedIn — GitHub — Google Scholar

Summary — 2nd-year Ph.D. student in Computer Engineering researching hardware design automation and formal methods, with a focus on AI-driven arithmetic datapath flows and HLS. My work examines the intersection of programming languages, machine learning, and hardware design techniques to enhance scalability, correctness, and optimization in next-generation hardware design workflows

Skills

Programming Rust, Python, C/C++, Lean, CUDA
Hardware SystemVerilog, HLS, Formal Verification
Machine Learning PyTorch, Nvidia TensorRT, Quantization
Software Git, AWS, SQL

Education

Georgia Institute of Technology **Aug 2024 – May 2029**
Ph.D. in Electronic and Computer Engineering - 4.0 GPA
Relevant Classes

- SAT/SMT Solvers, Internet Architecture and Protocols
- High Dimensional Statistics and Optimization, Advanced Programming Techniques, Hardware-Software Co-Design in ML systems, Graphical Models in ML

Imperial College London **Oct 2020 – Jun 2024**
Masters of Engineering in Electronic and Information Engineering - 1^{st} Class Honours (4.0 GPA)
Dean's List 3^{rd} Year (Top 10%), Dean's List 4^{th} Year (Top 5%)
Relevant Classes

- Advanced DL Systems, Topics in Large Dimensional Data Processing, Mathematics for ML, Financial Signal Processing and ML, Probability and Stochastic Processes
- Advanced Computer Architecture, Digital Systems Design, Computer Vision, ML, DL, Digital Signal Processing, Operations Research, Embedded Systems, Compilers

Technical Experience

Numerical Hardware Engineer Intern at AMD **May 2025 – Aug 2025**

- Invented novel integer and constant multiplier architectures, improving both minimum delay and area
- Designed E-Graph multiplier framework capturing array creation and reduction design space, enabling the exploration of arbitrarily hybrid architectures
- Developed methods to automatically generate multipliers optimized to input data properties such as arrival time and toggle rates

Numerical Hardware Engineer Intern at Intel **Jun 2024 – Aug 2024**

- Automated mixed-precision floating point multiplier design optimization in Rust using E-Graphs, improving PPA of existing designs by up to 10%.
- Investigated glitch power minimization in hardware designs using Integer Linear Programming
- Explored techniques and methods for automatic dot-product hardware unit generation

FPGA Engineering Co-Op at Quantum Motion **Apr 2023 – Oct 2023**

- Designed and integrated bespoke signal generator for High-Speed Qubit Feedback on an FPGA
- Built custom Python code-base using PYNQ and QICK interface with the FPGA
- Programmed RP2040 MicroController to communicate using TCP/IP over ethernet connections in C

Analogue and Digital IC Validation Intern at Quantum Motion **Jul 2022 – Oct 2022**

- Validated operation of Ring Oscillator hardware on silicon chips
- Analysed transistor properties across a range of chips, at room and cryogenic Temperatures (2K) with Python

Research Experience

E-Graph Reinforcement Learning Framework
Georgia Institute of Technology - AMD - Current Research

- Building a generalized RL framework to streamline e-graph exploration, addressing scalability limitations

Hardware Arithmetic Formal Verification in Lean
Georgia Institute of Technology - Current Research

- Framework built in Lean to formally verify arithmetic RL designs against a strict bitvector arithmetic DSL
- Developed a Yosys-based frontend for Verilog designs alongside the DSL parser, in Rust
- Support for SMT-based proofs, parameterized model checks, multiplier verification, and floating point properties.

Automated HLS Design Modularization using E-Graphs
Georgia Institute of Technology - First Author - Submitted to ASPLOS 2026, Presented at PLDI 2025 - Available Here

- Developed a framework that integrates e-graphs, anti-unification, and polyhedral loop analysis to automatically identify and extract reusable hardware modules from HLS designs
- Introduced hardware-aware cost functions to guide module selection and design extraction, enabling scalable reuse across diverse workloads, including GEMM, GEMV, and self-attention
- Demonstrated up to 99% hardware reuse and significant area reductions across benchmarks, with modularized designs synthesized on FPGA/ASIC flows (Catapult HLS, Vitis HLS)

ForgeBench: A Machine Learning Benchmark Suite and Auto-Generation Framework
Georgia Institute of Technology - First Co-Author - Available Here

- An extensible benchmark framework auto-generating HLS ML designs across GEMM, DNN, and LLM workloads
- Created a modularized benchmark test suite demonstrating the necessity of architecture-oriented HLS tools

Quantifying and Automating Interpretability for Deep Learning
Imperial College London - Final Year Project

- Designed metric to quantitatively evaluate the interpretability of a deep learning model, enabling trade-off analysis
- Developed an automated workflow to add interpretability to CNNs, achieving up to an 84% inference latency improvement over the manual INN model
- Demonstrated comparable interpretability results on standard CNNs with state-of-the-art GradCAM methods

OptiMult - Multiplier Optimization via E-Graph Rewriting
Imperial College London - First Author - ASILOMAR 2023 - Available Here

- Created tool in Rust to compile hardware arithmetic expressions into optimized representations for area and latency
- Demonstrated up to a 46% latency reduction in squarer circuits and 9% latency reduction in general multiplication against industry standard logic synthesis tools

Notable Projects

Deep Learning Activation Function Hardware Accelerators in SystemVerilog **Jan 2024 – Mar 2024**

- Designed and Verified 4 – 16 bit parameterized implementations of Softmax and element-wise activation functions
- Integrated activation functions into a re-configurable, dynamically scheduled hardware architecture

Deep Learning Models in PyTorch - U-Net, VAE Transformer, DCGAN **Jan 2023 – Mar 2024**

- Implemented a U-Net CNN architecture to perform image classification and segmentation on brain scans
- Analyzed a VAE transformer architecture on generating and classifying the MNIST dataset
- Investigated Class Conditional & Wasserstein-GP DCGAN architectures on generating in the CIFAR-10 images

Cosine Accelerator on FPGA in SystemVerilog **Jan 2023 – Mar 2023**

- Reduced the latency of a vector function by 77% on an FPGA using an optimized CORDIC block in SystemVerilog
- Created customized floating-point arithmetic hardware, with low-level programming in C for further speed-ups