Eren Dogan

www.edogan.us
erdogan@ucsc.edu

Education

University of California, Santa Cruz — Santa Cruz, CA
PhD in CSE; GPA: 3.84/4.00; Advisor: Prof. Matthew Guthaus — *September 2022 – Present*

Ozyegin University — Istanbul, Turkey
BS in CS; GPA: 4.00/4.00 — *September 2017 – June 2022*

Research

University of California, Santa Cruz — Santa Cruz, CA
PhD Student, Advisor: Prof. Matthew Guthaus — *September 2022 – Present*

- **GAT-Steiner: Rectilinear Steiner Minimal Tree Prediction Using GNNs**: The RSMT problem is a fundamental problem in VLSI placement and routing and is known to be NP-hard. In this work, we propose GAT-Steiner, a graph attention network model that correctly predicts 99.846% of the nets in the ISPD19 benchmark with an average increase in wire length of only 0.480% on suboptimal wire length nets. On randomly generated benchmarks, GAT-Steiner correctly predicts 99.942% with an average increase in wire length of only 0.420% on suboptimal wire length nets.
- **Gridless Router for OpenRAM**: OpenRAM is an open-source static random access memory (SRAM) compiler. OpenRAM's supply and signal router was a grid-based router, which had issues like DRC errors and bad precision due to pin-grid misalignment. I implemented a new router that created Hanan graphs over the routing region to align pins and wires perfectly. The new router is faster, more precise, DRC-safe, and uses less wire.

Ozyegin University — Istanbul, Turkey
Research Intern, Advisor: Prof. H. Fatih Ugurdag — *October 2020 – September 2021*

- **OpenCache**: An open-source generator to create custom caches using OpenRAM's SRAM arrays. It generates a synthesizable Verilog file for cache logic and configuration files for OpenRAM to generate the internal SRAMs of the cache. OpenCache inputs a configuration file that includes various parameters about the desired cache such as total size, number of ways, replacement policy, etc. Additionally, OpenCache can use other EDA tools to verify the output cache through randomly generated testbenches. This generator is written in Python using the Amaranth library, which is a Python-to-HDL toolkit.
- **Deep Compression for PyTorch Models**: Improving a "PyTorch to C generator" by applying compression methods on Convolutional Neural Networks (CNNs). Mr. Hasan Unlu of Tesla has developed a generator to deploy PyTorch models on microcontrollers efficiently. To improve this generator, I used pruning and quantization methods. Weights are saved as compressed sparse column (CSC) format to decrease memory usage and forward pass functions are improved to use CSC arrays directly without losing performance.

Experience

University of California, Santa Cruz — Santa Cruz, CA
Graduate Student Researcher — *January 2023 – Present*
I worked on the OpenRAM project and thesis research.

University of California, Santa Cruz — Santa Cruz, CA
Teaching Assistant — *September 2022 – Present*
I assisted the following courses: Introduction to Data Structures and Algorithms (CSE101)

Ozyegin University — Istanbul, Turkey
Undergradute Teaching Assistant — *September 2019 – June 2021*
I assisted the following courses: Computer Programming (CS101), Digital Systems (EE203), Computer Architecture (CS240)

Service

Subreviewer: DAC 2025, MLCAD 2024, VLSI-SoC 2024

Programming Skills

Languages: Python, C/C++, (System)Verilog, Chisel, Java, JavaScript, SQL, Bash
Miscellaneous: Git, Linux, PyTorch, Tensorflow, PyTorch Geometric, Amaranth