EEE598： Algorithm/Hardware Co-Design and Design Automation for Emerging AI Hardware

Course Instructor: Jiaqi Gu, ISTB4 465, jiaqigu@asu.edu

Textbooks and Materials
[Online] Efficient Processing of Deep Neural Networks, Synthesis Lectures on Computer Architecture, by Sze, Chen, Yang, and Emer, 2020

Prerequisites

This course requires basic understanding of computer architecture, machine learning, and algorithm. Students should be comfortable with Python/C programming. Recommended prerequisites are Intro to Machine Learning/Deep Learning (EEE591 or equivalent), Digital Design and Circuits (EEE 425 or CSE 320 or EEE525).

Course Description and Objectives (subject to change)

This course focuses on co-design of machine learning algorithms and emerging AI hardware accelerators. To meet the escalating computing demand, the course introduces customized parallel digital AI accelerators, e.g., GPU/TPU, and emphasizes the transformative potential of emerging technologies, including analog electronics, neuromorphic hardware, and optical/photonic accelerators. These novel hardware accelerators have showcased unprecedented efficiency and performance gains toward ultra-efficient, high-performance machine intelligence. However, they also bring more design complexity and unique challenges. The course emphasizes the importance of cross-layer co-design, domain-specific customization, and design automation to enable efficient, reliable, and adaptable deployment of those emerging AI hardware given the constraints of resources, runtime, and robustness. With the synergy between co-design and automation, we can ensure fast design, evaluation, implementation, and exploration in the huge design space, ultimately realizing beyond-human design quality and adaptability. The course focuses on a new perspective of next-generation AI hardware with emerging semiconductor technologies and provides in-depth discussion of algorithmic, architectural, and circuit-level challenges and co-design techniques for trading off accuracy, efficiency, speed, reliability, and adaptability when designing accelerators for machine learning systems.

Topics

Overview of Efficient ML acceleration basics

1. ML algorithm basics (1 week): modern DNN (MLP, CNN, RNN, Transformer) models for vision/language workloads.
2. Introduction to core ML computational kernels (1 week): high-performance matrix multiplication. Linear algebra fundamentals and accelerating linear algebra. Efficient nonlinear activations.

3. Overview of digital ML accelerator (1 week): GPU/TPU architecture, dataflow, and model/data parallelism. Performance evaluation and trade-offs.

Efficient ML algorithm optimization

1. Efficient inference (2 week): quantization, pruning, tensor decomposition, subspace linear algebra, distillation, dynamic architecture, neural architecture search.
2. Efficient training (1 week): data compression, sparse training, distributed training, online/on-chip training.

Emerging AI hardware design and optimization

1. Analog electronic processing-in-memory architectures (1 week): DRAM, SRAM, RRAM, MRAM computing architectures.
2. Photonic/optical neural computing architectures (2-3 weeks): diffractive optical neural network with free-space optics; integrated photonic circuits for universal/subspace and static/dynamic linear operations. Physics principles, device modeling, circuit analysis, and high-level abstraction.
3. Other unconventional computing (1 week): spiking neural network, reservoir computing, quantum machine learning. Modeling, implementation, and optimization.
4. Energy efficiency-driven hardware-software co-design (1 week): power optimization, memory optimization, analog-digital domain co-optimization.
5. Robustness/reliability-driven hardware-software co-design (1 week): noise modeling, robust hardware design, noise-aware training and architecture scheduling, physics-aware in-situ optimization.
6. Design automation and computer-aided design (CAD) (1 week): automatic synthesis, place-route, circuit/architecture simulation, and architecture/circuit/model topology co-search, ML for CAD.

Expected Learning Outcomes

1. Learn the basics of machine learning algorithms and accelerator designs.
2. Learn to formulate and understand the trade-offs among multiple objectives for emerging AI hardware, performance, efficiency, robustness, and reprogrammability given various hardware/resource/physics constraints.
3. Learn emerging semiconductor technology for next-generation, unconventional computing beyond digital computers. Understand the potential benefits and design challenges.
4. Learn important design methodologies including cross-layer algorithm/hardware co-design, domain-specific customization, and design automation to maximize design quality and efficiency.
5. Gain hands-on experience in parallel programming on GPUs, linear algebra acceleration, and analog neural network training and simulation using modern ML frameworks and open-source/commercial simulation facilities.
6. Build research experience through paper reading, discussion, and research-oriented class projects.

Assignments

The course includes 2-3 assignments and one research-oriented group-based course projects (milestone reports + final report + final presentation). The outcome of the course project might potentially lead to a research publication. Assignments will include problem solving and hands-on practice in programming in Python/CUDA for parallel machine learning acceleration, NN training, and circuit/model co-optimization.