COMPUTER SCIENCE & ENGINEERING
UNIVERSITY OF MICHIGAN

DISSERTATION DEFENSE

Yuhan Chen

Optimizing Graph Applications Through Graph Sparsification and Accelerator Design

Friday, June 14, 2024
10:00am – 12:00pm
3941 Beyster
Hybrid – Zoom Passcode: 123456

ABSTRACT: Emerging applications such as video transcoding and graph algorithms have seen fast development and broad adoption recently. It is crucial to improve the performance of these emerging applications for cost-efficiency and scalability. This thesis focuses on video transcoding and graph algorithms and uses software hardware co-design to optimize their execution.

Video transcoding is rapidly growing as the demand for online streaming services continues to strive, and understanding the hardware bottleneck in performing video transcoding is the stepping stone to develop dedicated hardware for it.

Graph data structure is widely used in modeling complicated relationships between entities. Algorithms and applications that utilize the expressiveness of graphs are rapidly evolving and employed in various domains like social networks, chemistry, biology, and physics. With the expanding family of graph algorithms and the exploding size of real-world graphs, it is hard for hardware to keep up with the ever-growing demand for processing power for graph algorithms. To make the issue worse, the irregular memory access pattern in graph algorithms makes it hard to fully utilize traditional hardware like CPUs and GPUs.

In this thesis, I propose software and hardware co-design to improve the performance of emerging applications. At a high level, I first present hardware characterization that reveals the hardware bottlenecks with the change in software parameters. Then I benchmark the performance of the most popular graph sparsification algorithms on their performance in preserving graph properties. Finally, I propose a power-efficient accelerator supporting multiple dataflows for Graph Convolutional Networks.

Specifically, first, I perform CPU characterization on video transcoding, revealing the hardware bottlenecks (e.g. frontend, backend, branch misprediction, stalls) and how they shift with software parameters. Second, I use graph sparsification to tackle the exploding size of real-world graphs. I conduct a comprehensive benchmark on 12 graph sparsification algorithms, exploring their performance in preserving 16 essential graph properties on 14 real-world graphs, and give insights into how to choose the appropriate sparsification method for different down-stream tasks. Last, I present PEDAL, a power-efficient Graph Convolutional Network (GCN) accelerator designed to support multiple dataflows, achieving both high execution efficiency and flexibility.

CHAIR: Prof. Trevor Mudge and Research Scientist Nishil Talati