Research Summary

Yuke Wang

My name is Yuke Wang, an incoming fourth-year Ph.D. student in the department of computer science at the University of California, Santa Barbara. Over my past three-year Ph.D. study, I have achieved a GPA of 3.92/4.00 and have published 14 papers in top-tier conferences/journals (e.g., SC'21, OSDI'21, PPoPP'21, and TCAD'21) in deep learning algorithms, GPU-based high-performance computing, and runtime system design and optimization. I am also actively participating in diverse professional activities, such as serving on the Artifact Evaluation committee for top venues (e.g., SC'21, MICRO'21, and SOSP'21), to facilitate the artifact opensource and reproducibility of the research field. Besides, I was joining NVIDIA's Clara Parabricks team as the high-performance engineer intern during the summer (June to September) of 2021, where I worked with a team of NVIDIA's GPU experts and successfully improved the end-to-end performance by 30% for the existing genomics processing pipeline in production.

The major research focus of my Ph.D. career is to exploit the inefficiency of the existing deep-learning (DL) applications (e.g., deep neural networks (DNNs) and Graph neural networks (GNNs)) and improve their performance on GPUs through an array of technical innovations, such as *optimizing DL algorithms to eliminate redundant computations*, *designing highly efficient GPU kernels to maximize hardware utilization,* and *parameterizing system designs to improve the design adaptability and portability*. My final research goal is to facilitate the design and implementation of the next-generation high-performance and energy-efficient DL systems for a diverse range of real-world applications that demand powerful computation support (e.g., autonomous driving, AR/VR). Below I would briefly discuss my highlighted work to achieve this goal:

Novel DL Algorithms and GPU kernels. My first-author work, named **"*DSXplore: Optimizing Convolutional Neural Networks via Sliding-Channel Convolutions*",** significantly reduces the DNN computation and memory overhead. At the algorithm level, DSXplore incorporates a novel factorized kernel -- sliding-channel convolution (SCC), featured with input-channel overlapping to balance the accuracy performance and the reduction of computation and memory cost. We also carry out an optimized GPU implementation tailored for SCC by leveraging several key techniques, such as input-centric backward propagation and channel-cyclic optimization. DSXplore can reduce up to 3x memory overhead and save 40% to 60% computations. This work is accepted at **IPDPS'21** (***IEEE International Parallel & Distributed Processing Symposium***), a top-tier venue in the parallel programming and distributed system domain, and our design has been open-sourced on GitHub.

Efficient GPU-based Runtime Systems. My first-author work, named ***"GNNAdvisor: Intelligent Runtime for GNN acceleration on GPUs",*** builds an intelligent runtime system for boosting the GNNs performance on a GPU-based platform. GNNAdvisor combines the *input-level* (e.g., graph and node embedding) and *system-level* (e.g., GPU kernel design) optimizations to improve the GNN performance comprehensively. GNNAdvisor largely reduces the execution latency by 2x-3x compared with the state-of-the-art GNN frameworks on GPUs, which can maximize the energy efficiency and performance-per-dollar gains when deploying on cloud-based platforms. This work is accepted at **OSDI'21** ***(USENIX Symposium on Operating Systems Design and Implementation***), a flagship conference in the computer system field, and our design has been open-sourced.

Novel practice of new GPU hardware features. My first-author work, ***"APNN-TC: Accelerating Arbitrary Precision Neural Networks on Ampere GPU Tensor Cores",*** boosts the performance of DNN models by

exploiting the benefits of model quantization and high-performance GPU Tensor Cores. APNN-TC incorporates a novel emulation algorithm to support arbitrary short bit-width computation with int1 compute primitives and XOR/AND Boolean operations., APNN-TC also integrates arbitrary precision layer designs to efficiently map our emulation algorithm to Tensor Cores with novel batching strategies and specialized memory organization. Besides, APNN-TC embodies a novel arbitrary precision NN design to minimize memory access across layers and further improve performance. Extensive evaluations show that APNN-TC can achieve significant speedup over CUTLASS kernels and various NN models, such as ResNet and VGG. This work is accepted at **SC'21** ***(The International Conference for High-Performance Computing, Networking, Storage, and Analysis***), the leading conference in the supercomputing field, and our design has been open-sourced on GitHub.

My recent first-author work, **"*QGTC: Accelerating Quantized GNN via GPU Tensor Core*"**, introduces the first Tensor Core (TC) based computing framework, QGTC, to support any-bitwidth computation for QGNNs on GPUs. We introduce a novel quantized low-bit arithmetic design based on the low-bit data representation and bit-decomposed computation. We craft a novel TC-tailored CUDA kernel design by incorporating 3D-stacked bit compression, zero-tile jumping, and non-zero tile reuse technique to improve the performance systematically. We incorporate an effective bandwidth-optimized subgraph packing strategy to maximize the transferring efficiency between CPU host and GPU device. We integrate QGTC with PyTorch for better programmability and extensibility. Extensive experiments demonstrate that QGTC achieves an average 3.17x speedup compared with the state-of-the-art Deep Graph Library framework across diverse settings. This project is submitted to **PPoPP'22** ***(ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming***), a top venue in parallel programming.

This fellowship will significantly facilitate my research by providing me easier access to state-of-the-art research resources (e.g., computer servers, GPUs, and online books) and encouraging me for coming up with new ideas to address real-world challenges.

Best Regards,

Yuke Wang
Ph.D. Candidate
Department of Computer Science
University of California, Santa Barbara