Chengtao Lai

Department of Electronic & Computer Engineering
Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR, China
claiaf@connect.ust.hk

Education

Sep 2021– Present	Electronic and Computer Engineering, HKUST, Hong Kong SAR, China
Sep 2017 – June 2021	Engineering Mechanics, Tsinghua University, Beijing, China
July 2018	Tsinghua Distinguished Students' Program, Christ Church, Oxford, England
Sep 2014 – June 2017	Shanghai High School, Shanghai, China

Academic Performance

CGA: 3.983/4.300

Related Courses: Advanced Artificial Intelligence (4.3/4.3), Advanced Algorithms (4.3/4.3), Advanced Computer Architecture (4.3/4.3)

Research Interest

AI accelerator architectures, Neural Network scheduling and mapping, software-hardware codesign

Publications

Lai, Chengtao[*]; Zhou, Zhongchun[*]; Poptani, Akash and Zhang, Wei: "LCM: LLM-focused Hybrid SPM-cache Architecture with Cache Management for Multi-Core AI Accelerators"

– International Conference on Supercomputing (**ICS '24**), Kyoto

Lai, Chengtao and Zhang, Wei: "gem5-NVDLA: A Simulation Framework for Compiling, Scheduling and Architecture Evaluation on AI System-on-Chips"

– Presented at Embedded Systems Research Software Competition, Embedded Systems Week (**ESWEEK '23**), Hamburg

– Extended as Designer's Note in ACM Transactions on Design Automation of Electronic Systems (**TODAES**)

Contribution to the Open-source Community

Owner of Repo: gem5-NVDLA (https://github.com/suchandler96/gem5-NVDLA/tree/DMAEnable) (13 stars)

Pull request accepted: gem5-rtl (https://gitlab.bsc.es/glopez/gem5-rtl)

Research Experience

Hybrid SPM-cache Architecture with Cache Management for Multi-Core AI Accelerators

I have explored a multi-core AI accelerator system that incorporates a shared cache and application-specific management strategies, to relieve the burden of the compiler at the cost of sometimes slightly lower performance for dense workloads than SPM-based systems. By aiding the shared cache with a metadata storage of tensor and tile information, a counter-based dead block prediction method can be implemented so that the replacement policy has a global view on cache line lifecycles. Besides, data prefetching can be done from the hardware side with these metadata without the need of writing data transfer instructions explicitly in the program.

gem5-NVDLA: A Sim. Framework for Compiling, Scheduling and Architecture Evaluation on AI SoCs

I have integrated the NVDLA accelerator into gem5 with compilation support and various scheduling algorithms and architecture building blocks. This framework, as opposed to most open-source frameworks in the academia, focuses on higher-level controlling of AI accelerators like inter-operator data reuse and multi-accelerator scheduling. As far as we know, inter-operator data reuse (not operator fusing) has previously only been discussed in industrial works from Google and IBM and has been **neglected by most publications from the academia.** A case study is also conducted to demonstrate the importance of adopting different buffering strategies for activation and weight tensors in AI accelerators to acquire remarkable speedup.

Postgraduate Teaching Assistant Duties

Probability and Random Processes (ELEC2600, Spring 2021-22)

FPGA Based Design: From Theory to Practice (ELEC4320, Fall 2022-23 & Fall 2023-24)

Selected Awards and Honors

RedBird Scholarship for new research postgraduate students (2021)

Research Skills

Skilled at C/C++, python programming;

Familiar with OpenMP, MPI and CUDA optimization;

Rich experience in architecture simulation frameworks like gem5.