48th International Symposium on Computer Architecture

June 14-19, 2021

Program

https://iscaconf.org

PROGRAM

Monday, June 14th

10:00 - 11:00AM Keynote by Hillery Hunter

10:00 AM – 11:00 AM (EDT/New York)

7:00 AM (PDT/San Francisco), 16:00 (CEST/Brussels), 22:00 (CST/Beijing)

From Transistors to Enclaves: How Architects Have Helped Secure Digital Assets, Healthcare Records, and So Much More

Abstract:

Cloud computing today offers a wealth of capacity, elasticity, and computing choice. As architects, we often focus on the speeds and feeds that keep this computing humming along, but in today's climate, security is top of mind. In this talk, we'll explore the journey which has led to at-scale, at-speed confidential computing — capabilities which now enable protection of sensitive data sets, with privacy and authority over computation and data, even when leveraging the advantages of a multi-tenant cloud environment. From custody of digital assets to protection of healthcare records and financial payments, we'll look at how the work of architects is having real-world impact each and every day.

Bio

Hillery Hunter is CTO of IBM Cloud, responsible for technical strategy for IBM's cloud-native and infrastructure offerings. Prior to this role, she served as Director of Accelerated Cognitive Infrastructure in IBM Research, leading a team doing cross-stack (hardware through software) optimization of AI workloads, producing productivity breakthroughs of 40x and greater which were transferred into IBM product offerings. Her technical interests have always been interdisciplinary, spanning from silicon technology through system software, and she has served in technical and leadership roles in memory technology, Systems for AI, and other areas. She is a member of the IBM Academy of Technology and was appointed as an IBM Fellow in 2017. Hillery is a BS, MS, and PhD graduate of the University of Illinois at Urbana–Champaign.

11:00 - 12:00AM Session 1. Industry Track

11:00 AM – 12:00 PM (EDT/New York)

8:00 AM (PDT/San Francisco), 17:00 (CEST/Brussels), 23:00 (CST/Beijing)

11:00 - 11:12AM Ten Lessons From Three Generations Shaped Google's TPUv4i

Norman P. Jouppi, Doe Hyun Yoon, Matthew Ashcraft, Mark Gottscho, Thomas B. Jablin, George Kurian, James Laudon, Sheng Li, Peter Ma, Xiaoyu Ma, Nishant Patil, Sushma Prasad, Clifford Young, Zongwei Zhou (Google); David Patterson (Google / Berkeley)

Abstract:

Google deployed several TPU generations since 2015, teaching us lessons that changed our views: semiconductor technology advances unequally; compiler compatibility trumps binary compatibility, especially for VLIW domain-specific architectures (DSA); target total cost of ownership vs initial cost; support multi-tenancy; deep neural networks (DNN) grow 1.5X annually; DNN advances evolve workloads; some inference tasks require floating point; inference DSAs need air-cooling; apps limit latency, not batch size; and backwards ML compatibility helps deploy DNNs quickly. These lessons molded TPUv4i, an inference DSA deployed since 2020.

11:12 - 11:24AM Sparsity-Aware and Re-Configurable NPU Architecture for Samsung Flagship Mobile SoC

Jun-Woo Jang, Sehwan Lee, Dongyoung Kim, Hyunsun Park (Samsung Advanced Institute of Technology); Ali Shafiee Ardestani (Samsung Semiconductor); Yeongjae Choi, Channoh Kim, Yoojin Kim, Hyeongseok Yu (Samsung Advanced Institute of Technology); Hamzah Ahmed Ali Abdelaziz (Samsung Semiconductor); Jun-Seok Park, Heonsoo Lee, Dongwoo Lee (Samsung Electronics); Myeong Woo Kim, Hanwoong Jung, Heewoo Nam, Dongguen Lim, Seungwon Lee, Joon Ho Song (Samsung Advanced Institute of Technology); Suknam Kwon (Samsung Electronics); Joseph Hassoun (Samsung Semiconductor); SukHwan Lim (Samsung Electronics); Changkyu Choi (Samsung Advanced Institute of Technology)

Abstract

Of late, deep neural networks have become ubiquitous in mobile applications. As mobile devices generally require immediate response while maintaining user privacy, the demand for on-device machine learning technology is on the increase. Nevertheless, mobile devices suffer from restricted hardware resources, whereas deep neural networks involve considerable computation and communication. Therefore, the implementation of a neural-network specialized hardware accelerator, generally called neural processing unit (NPU), has started to gain attention for the mobile application processor (AP). However, NPUs for commercial mobile AP face two challenges that are difficult to realize simultaneously: execution of a wide range of applications and efficient performance.

In this paper, we propose a flexible but efficient NPU architecture for a Samsung flagship mobile system-on-chip (SoC). To implement an efficient NPU, we design an energy-efficient inner-product engine that utilizes the input feature map sparsity. We propose a re-configurable MAC array to enhance the flexibility of the proposed NPU, dynamic internal memory port assignment to maximize onchip memory bandwidth utilization, and efficient architecture to support mixed-precision arithmetic. We implement the proposed NPU using the Samsung 5nm library. Our silicon measurement experiments demonstrate that the proposed NPU achieves 290.7 FPS and 13.6 TOPS/W, when executing an 8-bit quantized Inception-v3 model with a single NPU core. In addition, we analyze the proposed zero-skipping architecture in detail. Finally, we present the findings and lessons learned when implementing the commercial mobile NPU and interesting avenues for future work.

11:24 - 11:36AM Energy Efficiency Boost in the AI-Infused POWER10 Processor

Brian W. Thompto, Dung Q. Nguyen, Jose E. Moreira, Ramon Bertran, Hans Jacobson, Richard J. Eickemeyer, Rahul M. Rao, Michael Goulet, Marcy Byers, Christopher J. Gonzalez, Karthik Swaminathan, Nagu R. Dhanwada, Silvia M. M¨uller, Andreas Wagner, Satish Kumar Sadasivam, Robert K. Montoye, William J. Starke, Christian G. Zoellin, Michael S. Floyd, Jeffrey Stuecheli, Nandhini Chandramoorthy, John-David Wellman, Alper Buyuktosunoglu, Matthias Pflanz, Balaram Sinharoy, Pradip Bose (IBM Corp.)

Abstract

We present the novel micro-architectural features, supported by an innovative and novel pre-silicon methodology in the design of POWER10. The resulting projected energy efficiency boost over POWER9 is 2.6x at core level (for SPECint) and up to 3x at socket level. In addition, a new feature supporting inline AI acceleration was added to the POWER ISA and incorporated into the POWER10 processor core design. The resulting boost in SIMD/AI socket performance is projected to be up to 10x for FP32 and 21x for INT8 models of ResNet-50 and BERT-Large. In this paper, we describe the novel methodology deployed and used not only to obtain these efficiency boosts for traditional workloads, but also to infuse AI/ML/HPC capability directly into the POWER10 core.

11:36 - 11:48AM Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology

Sukhan Lee, Shin-haeng Kang, Jaehoon Lee, Hyeonsu Kim, Eojin Lee, Seungwoo Seo, Hosang Yoon, Seungwon Lee, Kyounghwan Lim, Hyunsung Shin, Jinhyun Kim, Seongil O, Anand Iyer, David Wang, Kyomin Sohn, Nam Sung Kim (Samsung Electronics)

Abstract:

Emerging applications such as deep neural networks demand high off-chip memory bandwidth for high performance. However, under stringent physical constraints of chip packages and system boards, it becomes very expensive to further increase the bandwidth of off-chip memory. Besides, transferring data across the memory

hierarchy constitutes a large fraction of total energy consumption of systems, and the fraction has steadily increased with the stagnant technology scaling and poor data reuse characteristics of such emerging applications. To cost-effectively increase the bandwidth and energy efficiency, researchers began to reconsider the past processing-in-memory (PIM) architectures and advance them further, especially with recent integration technologies such as 2.5D/3D stacking. Albeit the recent advances, no major memory manufacturer has developed even a proof-ofconcept silicon yet, not to mention a product. This is because the past PIM architectures often require notable changes in host processors and/or application code which memory manufacturers cannot easily govern. In this paper, elegantly tackling the aforementioned challenges, we propose a function-in-memory (FIM) architecture. To demonstrate its feasibility and effectiveness at the system level, we implement it with a 20nm DRAM technology, integrate it with an unmodified commercial processor, develop the necessary software stack, and run existing applications without any change. Our evaluation at the system level shows that FIM reduces the end-to-end execution time of memory-bound neural network kernels and applications by 8.7× and 3.5×, respectively. Atop the performance improvement, FIM also reduces the energy per bit transfer by 3.5×, and the overall energy efficiency of the system running the applications by 3.2×.

11:48 - 12:00AM Pioneering Chiplet Technology and Design for the AMD EPYC™ and Ryzen™ Processor Families

Samuel Naffziger, Kevin Lepak, Mahesh Subramony, Noah Beck, Sean White, Gabriel Loh (AMD)

12:00 - 12:45PM Session 2A. Microarchitecture I

12:00 PM – 12:45 PM (EDT/New York)

9:00 AM (PDT/San Francisco), 18:00 (CEST/Brussels), Tue 00:00 (CST/Beijing)

12:00 - 12:15PM Zero Inclusion Victim: Isolating Core Caches from Inclusive Last-Level Cache Evictions

Mainak Chaudhuri (IIT Kanpur)

Abstract

The most widely used last-level cache (LLC) architecture in the microprocessors has been the inclusive LLC design. The popularity of the inclusive design stems from the bandwidth optimization and simplification it offers to the implementation of the cache coherence protocols. However, inclusive LLCs have always been associated with the curse of inclusion victims. An inclusion victim is a block that must be forcefully replaced from the inner levels of the cache hierarchy when the copy of the block is replaced from the inclusive LLC. This tight coupling between the LLC victims and the inner-level cache contents leads to three major drawbacks. First, live inclusion victims can lead to severe performance degradation depending on the LLC replacement policies. Second, a process can victimize the blocks of another process in an LLC shared by multiple cores and this can be exploited to leak information through well-known eviction-based timing side-channels. An inclusive LLC makes these channels much less noisy due to the presence of inclusion victims which allow the malicious processes to control the contents of the per-core private caches through LLC evictions. Third, to reduce the impact of the aforementioned two drawbacks, the inner-level caches, particularly the mid-level cache in a three-level inclusive cache hierarchy, must be kept small even if a larger mid-level cache could have been beneficial in the absence of inclusion victims.

We observe that inclusion victims are not fundamental to the inclusion property, but arise due to the way the contents of an inclusive LLC are managed. Motivated by this observation, we introduce a fundamentally new inclusive LLC design named the Zero Inclusion Victim (ZIV) LLC that guarantees freedom from inclusion victims while retaining all advantages of an inclusive LLC. This is the first inclusive LLC design proposal to offer such a guarantee, thereby completely isolating the core caches from LLC evictions. We observe that the root cause of inclusion victims is the constraint that an LLC victim must be chosen from the set pointed to by the set indexing function. The ZIV LLC relaxes this constraint only when necessary by efficiently and minimally enabling a global victim selection scheme in the inclusive LLC to avoid generation of inclusion victims. Detailed simulations conducted with a chip-multiprocessor model using multi-programmed and multi-threaded workloads show that the ZIV LLC gracefully supports large mid-level caches (e.g., half the size of the LLC) and delivers performance close to a non-inclusive LLC for different classes of LLC replacement policies. We also show that the ZIV LLC comfortably outperforms the existing related proposals and its performance lead grows with increasing mid-level cache capacity.

12:15 - 12:30PM Exploiting Page Table Locality for Agile TLB Prefetching

Georgios Vavouliotis (UPC / BSC); Lluc Alvarez (BSC); Vasileios Karakostas, Konstantinos Nikas, Nectarios Koziris (NTU Athens); Daniel Jiménez (Texas A&M); Marc Casas (BSC)

Abstract

Frequent Translation Lookaside Buffer (TLB) misses incur high performance and energy costs due to page walks required for fetching the corresponding address translations. Prefetching page table entries (PTEs) ahead of demand TLB accesses can mitigate the address translation performance bottleneck, but each prefetch requires traversing the page table, triggering additional accesses to the memory hierarchy. Therefore, TLB prefetching is a costly technique that may undermine performance when the prefetches are not accurate.

In this paper we exploit the locality in the last level of the page table to reduce the cost and enhance the effectiveness of TLB prefetching by fetching cache-line adjacent PTEs "for free". We propose Sampling-Based Free TLB Prefetching (SBFP), a dynamic scheme that predicts the usefulness of these "free" PTEs and prefetches only the ones most likely to prevent TLB misses. We demonstrate that combining SBFP with novel and state-of-the- art TLB prefetchers significantly improves miss coverage and reduces most memory accesses due to page walks.

Moreover, we propose Agile TLB Prefetcher (ATP), a novel composite TLB prefetcher particularly designed to maximize the benefits of SBFP. ATP efficiently combines three low-cost TLB prefetchers and disables TLB prefetching for those execution phases that do not benefit from it. Unlike state-of-the-art TLB prefetchers that correlate patterns with only one feature (e.g., strides, PC, distances), ATP correlates patterns with multiple features and dynamically enables the most appropriate TLB prefetcher per TLB miss.

To alleviate the address translation performance bottleneck, we propose a unified solution that combines ATP and SBFP. Across an extensive set of industrial workloads provided by Qualcomm, ATP coupled with SBFP improves geometric speedup by 16.2%, and eliminates on average 37% of the memory references due to page walks. Considering the SPEC CPU 2006 and SPEC CPU 2017 benchmark suites, ATP with SBFP increases geometric speedup by 11.1%, and eliminates page walk memory references by 26%. Applied to big data workloads (GAP suite, XSBench), ATP with SBFP yields a geometric speedup of 11.8% while reducing page walk memory references by 5%. Over the best state-of-the-art TLB prefetcher for each benchmark suite, ATP with SBFP achieves speedups of 8.7%, 3.4%, and 4.2% for the Qualcomm, SPEC, and GAP+XSBench workloads, respectively.

12:30 - 12:45PM A Cost-Effective Entangling Prefetcher for Instructions

Alberto Ros, Alexandra Jimborean (Murcia)

Abstract:

Prefetching instructions in the instruction cache is a fundamental technique for designing high-performance computers. There are three key properties to consider when designing an efficient and effective prefetcher: timeliness, coverage, and accuracy. Timeliness is essential, as bringing instructions too early increases the risk of the instructions being evicted from the cache before their use and requesting them too late can lead to the instructions arriving after they are demanded. Coverage is important to reduce the number of instruction cache misses and accuracy to ensure that the prefetcher does not pollute the cache or interacts negatively with the other hardware mechanisms.

This paper presents the Entangling Prefetcher for Instructions that entangles instructions to maximize timeliness. The prefetcher works by finding which instruction should trigger the prefetch for a subsequent instruction, accounting for the latency of each cache miss. The prefetcher is carefully adjusted to account for both coverage and accuracy. Our evaluation shows that with 40KB of storage, Entangling can increase performance up to 23%, outperforming state-of-the-art prefetchers.

12:00 - 12:45PM Session 2B. Memory I

12:00 PM – 12:45 PM (EDT/New York)

9:00 AM (PDT/San Francisco), 18:00 (CEST/Brussels), Tue 00:00 (CST/Beijing)

12:00 - 12:15PM Don't Forget the I/O When Allocating Your LLC

Yifan Yuan (UIUC); Mohammad Alian (Kansas); Yipeng Wang, Ren Wang (Intel Labs); Ilia Kurakin (Intel); Charlie Tai (Intel Labs); Nam Sung Kim (UIUC)

Abstract:

In modern server CPUs, last-level cache (LLC) is a critical hardware resource that exerts significant influence on the performance of the workloads, and how to manage LLC is a key to the performance isolation and QoS in the cloud with multi-tenancy.

In this paper, we argue that in addition to CPU cores, high-speed I/O is also important for LLC management.

This is because of an Intel architectural innovation -- Data Direct I/O (DDIO) -- that directly injects the inbound I/O traffic to (part of) the LLC instead of the main memory.

We summarize two problems caused by DDIO and show that

(1) the default DDIO configuration may not always achieve optimal performance,

(2) DDIO can decrease the performance of non-I/O workloads that share LLC with it by as high as 32%.

We then present IAT, the first LLC management mechanism that treats the I/O as the first-class citizen.

IAT monitors and analyzes the performance of the core/LLC/DDIO using CPU's hardware performance counters and adaptively adjusts the number of LLC ways for DDIO or the tenants that demand more LLC capacity.

In addition, IAT dynamically chooses the tenants that share its LLC resource with DDIO to minimize the performance interference by both the tenants and the I/O.

Our experiments with multiple microbenchmarks and real-world applications demonstrate that with minimal overhead, IAT can effectively and stably reduce the performance degradation caused by DDIO.

12:15 - 12:30PM PF-DRAM: A Precharge-Free DRAM Structure

Nezam Rohbani (IPM); Sina Darabi (Sharif); Hamid Sarbazi-Azad (Sharif / IPM)

Abstract:

Although DRAM capacity and bandwidth have increased sharply by the advances in technology and standards, its latency and energy per access have remained almost constant in recent generations. The main portion of DRAM power/energy is dissipated by Read, Write, and Refresh operations, all initiated by a Precharge phase. Precharge phase not only imposes a large amount of energy consumption, but also increases the delay of closing a row in a memory block to open another one. By reduction of row-hit rate in recent workloads, especially in multi-core systems, precharge rate increases which exacerbates DRAM power dissipation and access latency. This work proposes a novel DRAM structure, called Precharge-Free DRAM (PFDRAM), that eliminates the Precharge phase of DRAM. PFDRAM uses the charge on bitlines from the previous Activation phase, as the starting point for the next Activation. The difference between PF-DRAM and conventional DRAM structure is limited to precharge and equalizer circuitry and simple modifications in sense amplifier, which are all limited to subarray level. PF-DRAM is compatible with the mainstream JEDEC memory standards like DDRx and HBM, with minimum modifications in memory controller. Furthermore, almost all of the previously proposed power/energy reduction techniques in DRAM are still applicable to PF-DRAM for further improvement. Our experimental results on a 8GB memory system running SPEC CPU2017 and PARSEC 2.1 workloads show an average of 35.3% memory power consumption reduction (up to 54.2%) achieved by the system using PF-DRAM with respect to the system using conventional DRAM. Moreover, the overall performance is improved by 8.6%, in average (up to 24.3%). According to our analysis, all such improvements are achieved for less than 9% area overhead.

12:30 - 12:45PM Efficient Multi-GPU Shared Memory via Automatic Optimization of Fine-Grained Transfers

Harini Muthukrishnan (Michigan); David Nellans, Daniel Lustig (NVIDIA); Jeffrey A. Fessler, Thomas Wenisch (Michigan)

Abstract:

Despite continuing research into inter-GPU communication mechanisms, extracting performance from multi-GPU systems remains a significant challenge. Inter-GPU communication via bulk DMA-based transfers exposes transfer latency on the GPU's' critical execution path because these large transfers are logically interleaved between compute kernels. Conversely, fine-grained peer-to-peer memory accesses during kernel execution lead to memory stalls that can exceed the GPUs' ability to cover these operations via multi-threading. Worse yet, these sub-cacheline transfers are highly inefficient on current inter-GPU interconnects. To remedy these issues, we propose PROACT, a system enabling remote memory transfers with the programmability and pipeline advantages of peer-to-peer stores, while achieving interconnect efficiency that rivals bulk DMA transfers. Combining compile-time instrumentation with fine-grain tracking of data block readiness within each GPU, PROACT enables interconnect-friendly data transfers while hiding the transfer latency via pipelining during kernel execution. This

work describes both hardware and software implementations of PROACT and demonstrates the effectiveness of a PROACT software prototype on three generations of GPU hardware and interconnects. Achieving near-ideal interconnect efficiency, PROACT realizes a mean speedup of 3.0x over single-GPU performance for 4-GPU systems, capturing 83% of available performance opportunity. On a 16-GPU NVIDIA DGX-2 system, we demonstrate an 11.0x average strong-scaling speedup over single-GPU performance, 5.3x better than a bulk DMA-based approach.

12:45 - 1:30PM Session 3A. Machine Learning I

12:45 PM – 1:30 PM (EDT/New York)

9:45 AM (PDT/San Francisco), 18:45 (CEST/Brussels), Tue 0:45 (CST/Beijing)

12:45 - 01:00PM RaPiD: AI Accelerator for Ultra-Low Precision Training and Inference

Swagath Venkataramani, Vijayalakshmi Srinivasan, Wei Wang, Sanchari Sen, Jintao Zhang, Ankur Agrawal, Monodeep Kar, Shubham Jain, Alberto Mannari, Hoang Tran, Yulong Li, Eri Ogawa, Kazuaki Ishizaki, Hiroshi Inoue, Marcel Schaal, Mauricio Serrano, Jungwook Choi, Xiao Sun, Naigang Wang, Chia-Yu Chen, Allison Allain, James Bonano, Nianzheng Cao, Robert Casatuta, Matthew Cohen, Bruce Fleischer, Michael Guillorn, Howard Haynie, Jinwook Jung, Mingu Kang, Kyu-hyoun Kim, Siyu Koswatta, Saekyu Lee, Martin Lutz, Silvia Mueller, Jinwook Oh, Ashish Ranjan, Zhibin Ren, Scot Rider, Kerstin Schelm, Michael Scheuermann, Joel Silberman, Jie Yang, Vidhi Zalani, Xin Zhang, Ching Zhou, Matt Ziegler, Vinay Shah, Moriyoshi Ohara, Pong-Fei Lu, Brian Curran, Sunil Shukla, Leland Chang, Kailash Gopalakrishnan (IBM Corp.)

Abstract:

The growing prevalence and computational demands of Artificial Intelligence (AI) workloads has led to widespread use of hardware accelerators in their execution. Scaling the performance of AI accelerators across generations is pivotal to their success in commercial deployments. The intrinsic error-resilient nature of AI workloads present a unique opportunity for performance/energy improvemen tthrough precision scaling. Motivated by the recent algorithmic advances in precision scaling for inference and training, we designed RAPIDa 4-core AI accelerator chip supporting aspectrum of precisions, namely, 16 and 8-bit floating-point and 4 and 2-bit fixed-point. The 36mm2 RAPID chip fabricated in 7nm EUV technology delivers a peak 3.5 TFLOPS/W in HFP8 mode and 16.5 TOPS/W in INT4 mode at nominal voltage. Using a performance model calibrated to within 1% of the measurement results, we evaluated DNN inference using 4-bitixed-point representation or a 4-core 1 RAPID chip system and DNN training using 8-bit floating point representation fora 768 TFLOPs AI system comprising 4 32-core RAPID chips. Our results show INT4 inference for batch size of 1 achieves 3 13.5 (average 7) TOPS/W and FP8 training for a mini-batch of 512 achieves a sustained 102 - 588 (average 203) TFLOPS across a wide range of applications.

01:00 - 01:15PM REDUCT: Keep It Close, Keep It Cool! - Scaling DNN Inference on Multi-Core CPUs with Near-Cache Compute

Anant Nori (Intel Labs); Rahul Bera (ETH Zurich); Shankar Balachandran, Joydeep Rakshit, Om J Omer (Intel Labs); Avishaii Abuhatzera, Belliappa Kuttanna (Intel); Sreenivas Subramoney (Intel Labs)

Abstract

Deep Neural Networks (DNN) are used in a variety of applications and services. With the evolving nature of DNNs, the race to build optimal hardware (both in datacenter and edge) continues. General purpose multi-core CPUs offer unique attractive advantages for DNN inference at both datacenter [60] and edge [71]. Most of the CPU pipeline design complexity is targeted towards optimizing general-purpose single thread performance, and is overkill for relatively simpler, but still hugely important, data parallel DNN inference workloads. Addressing this disparity efficiently can enable both raw performance scaling and overall performance/Watt improvements for multi-core CPU DNN inference.

We present REDUCT, where we build innovative solutions that bypass traditional CPU resources which impact DNN inference power and limit its performance. Fundamentally, REDUCT's "Keep it close" policy enables consecutive pieces of work to be executed close to each other. REDUCT enables instruction delivery/decode close to execution and instruction execution close to data. Simple ISA extensions encode the fixed iteration count loop-y workload behavior enabling an effective bypass of many power-hungry front-end stages of the wide Out-of-Order (OoO) CPU pipeline. Per core performance scales efficiently by distributing lightweight

tensor compute near all caches in a multi-level cache hierarchy. This maximizes the cumulative utilization of the existing architectural bandwidth resources in the system and minimizes movement of data.

Across a number of DNN models, REDUCT achieves a 2.3× increase in convolution performance/Watt with a 2× to 3.94× scaling in raw performance. Similarly, REDUCT achieves a 1.8× increase in inner-product performance/Watt with 2.8× scaling in performance. REDUCT performance/power scaling is achieved with no increase to cache capacity or bandwidth and a mere 2.63% increase in area. Crucially, REDUCT operates entirely within the CPU programming and memory model, simplifying software development, while achieving performance similar to or better than state-ofthe-art Domain Specific Accelerators (DSA) for DNN inference, providing fresh design choices in the AI era.

01:15 - 01:30PM Communication Algorithm-Architecture Co-Design for Distributed Deep Learning

Jiayi Huang (UCSB); Pritam Majumder, Sungkeun Kim, Abdullah Muzahid, Ki Hwan Yum, Eun Jung Kim (Texas A&M)

Abstract:

Large-scale distributed deep learning training has enabled developments of more complex deep neural network models to learn from larger datasets for sophisticated tasks. In particular, distributed stochastic gradient descent intensively invokes all-reduce operations for gradient update, which dominates communication time during iterative training epochs. In this work, we identify the inefficiency in widely used all-reduce algorithms, and the opportunity of algorithm-architecture co-design. We propose MultiTree all-reduce algorithm with topology and resource utilization awareness for efficient and scalable all-reduce operations, which is applicable to different interconnect topologies. Moreover, we co-design the network interface to schedule and coordinate the all-reduce messages for contention-free communications, working in synergy with the algorithm. The flow control is also simplified to exploit the bulk data transfer of big gradient exchange. We evaluate the co-design using different all-reduce data sizes for synthetic study, demonstrating its effectiveness on various interconnection network topologies, in addition to state-of-the-art deep neural networks for real workload experiments. The results show that MultiTree achieves 2.3× and 1.56× communication speedup, as well as up to 81% and 30% training time reduction compared to ring all-reduce and state-of-the-art approaches, respectively.

12:45 - 1.30PM Session 3B. Microarchitecture II

12:45 PM – 1:30 PM (EDT/New York)

09:45 AM (PDT/San Francisco), 18:45 (CEST/Brussels), Tue 0:45 (CST/Beijing)

12:45 - 01:00PM Vector Runahead

Ajeya Naithani (UGhent); Sam Ainsworth (Edinburgh); Timothy Jones (Cambridge); Lieven Eeckhout (UGhent)

01:00 - 01:15PM Unlimited Vector Extension with Data Streaming Support

João Domingos (INESC-ID / Instituto Superior Técnico, Universidade de Lisboa); Nuno Neves (INESC-ID / Instituto de Telecomunicações); Nuno Roma and Pedro Tomás (INESC-ID / Instituto Superior Técnico, Universidade de Lisboa)

Abstract

Unlimited vector extension (UVE) is a novel instruction set architecture extension that takes streaming and SIMD processing together into the modern computing scenario. It aims to overcome the shortcomings of state-of-the-art scalable vector extensions by adding data streaming as a way to simultaneously reduce the overheads associated with loop control and memory access indexing, as well as with memory access latency. This is achieved through a new set of instructions that pre-configure the loop memory access patterns. These attain accurate and timely data prefetching on predictable access patterns, such as in multidimensional arrays or in indirect memory access patterns. Each of the configured data streams is associated to a general purpose vector register, which is then used to interface with the streams. In particular, iterating over a given stream is simply achieved by reading/writing to the corresponding input/output stream, as the data is instantly consumed/produced. To evaluate the proposed UVE, a proof-of-concept gem5 implementation was integrated in an out-of-order processor model, based on the ARM Cortex-A76, thus taking into consideration the typical speculative and out-of-order execution paradigms found in high-performance computing

processors. The evaluation was carried out with a set of representative kernels, by assessing the number of executed instructions, its impact on the memory bus and its overall performance. Compared to other state-of-the-art solutions, such as the upcoming ARM Scalable Vector Extension (SVE), the obtained results show that the proposed extension attains average performance speedups over 2:4x for the same processor configuration, including vector length.

01:15 - 01:30PM Speculative Vectorisation with Selective Replay

Peng Sun (Cambridge); Giacomo Gabrielli (Arm); Timothy Jones (Cambridge)

Abstract:

While industry continues to develop SIMD vector ISAs by providing new instructions and wider data-paths, modern SIMD architectures still rely on the programmer or compiler to transform code to vector form only when it is safe.

Limitations in the power of a compiler's memory alias analysis and the presence of infrequent memory data dependences mean that whole regions of code cannot be safely vectorised without risking changing the semantics of the application, restricting the available performance.

We present a new SIMD architecture to address this issue, which relies on speculation to identify and catch memory-dependence violations that occur during vector execution.

Once identified, only those SIMD lanes that have used erroneous data are replayed; other lanes, both older and younger, keep the results of their latest execution.

We use the compiler to mark loops with possible cross-iteration dependences and safely vectorise them by executing on our architecture, termed selective-replay vectorisation (SRV).

Evaluating on a range of general-purpose and HPC benchmarks gives an average loop speedup of 2.9x, and up to 5.3x in the best case, over already-vectorised code.

This leads to a whole-program speedup of up to 1.19x (average 1.06x) over already-vectorised applications.

01:30 - 03:30PM Panel 1: The Microprocessor at 50

01:30 PM – 03:00 PM (EDT/New York)

10:30 AM (PDT/San Francisco), 19:30 (CEST/Brussels), Tue 01:30 (CST/Beijing)

This year marks the 50th anniversary of the Intel 4004, the world's first microprocessor and an engineering achievement that continues to evolve at a blistering pace. This technical and visionary panel offers the rare opportunity to bring together microprocessor experts who have been part of this evolution and watch them look back at 5 decades of achievement. We expect a lively discussion as the panel exchanges ideas about what the microprocessor might be in another 25 years (assuming it still exists in a recognizable form). The panelists collectively span most major microprocessor architectures and spent their careers at companies such as AMD, Acorn/Arm, Tensilica, Centaur, IBM, and Intel:

Federico Faggin: designer of the first commercial microprocessor (Intel 4004), awarded National Medal of Technology and Innovation

John Hennessy: co-founder of MIPS Technologies, pioneer of RISC (shared Turing award)

David Patterson: led Berkeley RISC project (which became the basis for Sun SPARC), pioneer of RISC (shared Turing award)

Glenn Henry: designed computers spanning from IBM mainframes to personal computers and custom x86 CPUs, IBM Fellow

Kathy Papermaster: led multiple IBM projects, including the Cell Broadband Engine microprocessor

Lee Smith: led development of software tools at Acorn and Arm, Arm Fellow

Shekhar Borkar: directed Intel microprocessor design for 34 years, former Intel Fellow

Chris Rowen: co-founder of MIPS Technologies, pioneer of microprocessor hardware/software co-design

The panel will be moderated by J. Scott Gardner, an independent microprocessor-technology analyst.

08:00 - 09:00PM Session 4A. Processing in/near Memory

08:00 PM – 09:00 PM (EDT/New York)

05:00 PM (PDT/San Francisco), Tue 02:00 (CEST/Brussels), Tue 08:00 (CST/Beijing)

08:00 - 08:15PM ABC-DIMM: Alleviating the Bottleneck of Communication in DIMM-Based Near-Memory Processing with Inter-DIMM Broadcast

Weiyi Sun, Zhaoshi Li, Shouyi Yin, Shaojun Wei, Leibo Liu (Tsinghua)

Abstract:

Near-Memory Processing (NMP) systems that integrate accelerators within DIMM (Dual-Inline Memory Module) buffer chips potentially provide high performance with relatively low design and manufacturing costs. However, an inevitable communication bottleneck arises when considering the main memory bus among peer DIMMs and the host CPU. This communication bottleneck roots in the bus-based nature and the limited point-to-point communication pattern of the main memory system. The aggregated memory bandwidth of DIMM-based NMP scales with the number of DIMMs. When the number of DIMMs in a channel scales up, the per-DIMM point-to-point communication bandwidth scales down, whereas the computation resources and local memory bandwidth per DIMM stay the same. For many important sparse data-intensive workloads like graph applications and sparse tensor algebra, we identify that communication among DIMMs and the host CPU easily dominates their processing procedure in previous DIMM-based NMP systems, which severely bottlenecks their performance.

To tackle this challenge, we propose that inter-DIMM broadcast should be implemented and utilized in the main memory system of DIMM-based NMP. On the hardware side, the main memory bus naturally scales out with broadcast, where per-DIMM effective bandwidth of broadcast remains the same as the number of DIMMs grows. On the software side, many sparse applications can be implemented in a form such that broadcasts dominate their communication. Based on these ideas, we design ABC-DIMM, which Alleviates the Bottleneck of Communication in DIMM-based NMP, consisting of integral broadcast mechanisms and Broadcast-Process programming framework, with minimized modifications to commodity software-hardware stack. Our evaluation shows that ABC-DIMM offers an 8.33× geo-mean speedup over a 16-core CPU baseline, and outperforms two NMP baselines by 2.59× and 2.93× on average.

08:15 - 08:30PM Sieve: Scalable In-Situ DRAM-Based Accelerator Designs for Massively Parallel k-mer Matching

Lingxi Wu, Rasool Sharifi, Marzieh Lenjani, Kevin Skadron, Ashish Venkat (Virginia)

Abstract

The rapid influx of biosequence data, coupled with the stagnation of the processing power of modern computing systems, highlights the critical need for exploring high-performance accelerators that can meet the ever-increasing throughput demands of modern bioinformatics applications. This work argues that processing in memory (PIM) is an effective solution to enhance the performance of k-mer matching, a critical bottleneck stage in standard bioinformatics pipelines, that is characterized by random access patterns and low computational intensity.

This work proposes three DRAM-based in-situ k-mer matching accelerator designs (one optimized for area, one optimized for throughput, and one that strikes a balance between hardware cost and performance), dubbed Sieve, that leverage a novel data mapping scheme to allow for simultaneous comparisons of millions of DNA base pairs, lightweight matching circuitry for fast pattern matching, and an early termination mechanism that prunes unnecessary DRAM row activation to reduce latency and save energy. Evaluation of Sieve using state-of-the-art workloads with real-world datasets shows that the most aggressive design provides an average of 326x/32x speedup and 74X/48x energy savings over multi-core-CPU/GPU baselines for k-mer matching.

08:30 - 08:45PM FORMS: Fine-Grained Polarized ReRAM-Based In-Situ Computation for Mixed-Signal DNN Accelerator

Geng Yuan (Northeastern); Payman Behnam (Georgia Tech); Zhengang Li (Northeastern); Ali Shafiee (Samsung); Sheng Lin, Xiaolong Ma (Northeastern); Hang Liu (Stevens); Xuehai Qian (USC); Mahdi Nazm Bojnordi (Utah); Yanzhi Wang (Northeastern); Caiwen Ding (UConn)

Abstract:

Recent work demonstrated the promise of using resistive random access memory (ReRAM) as an emerging technology to perform inherently parallel analog domain in-situ matrix-vector multiplication---the intensive and key computation in deep neural networks (DNNs).

One key problem is the weights that are signed values.

However, in a ReRAM crossbar, weights are stored as conductance of the crossbar cells, and the in-situ computation assumes all cells on each crossbar column are of the same sign.

The current architectures either use two ReRAM crossbars for positive and negative weights (PRIME), or add an offset to weights so that all values become positive (ISAAC).

Neither solution is ideal: they either double the cost of crossbars, or incur extra offset circuity. To better address this problem, we propose FORMS, a fine-grained ReRAM-based DNN accelerator with algorithm/hardware co-design.

Instead of trying to represent the positive/negative weights, our key design principle is to enforce exactly

what is assumed in the in-situ computation---ensuring that all weights in the same column of a crossbar have the same sign.

It naturally avoids the cost of an additional crossbar.

Such polarized weights can be nicely generated using alternating direction method of multipliers (ADMM) regularized optimization during the DNN training, which can exactly enforce certain patterns in DNN weights.

To achieve high accuracy, we divide the crossbar into logical sub-arrays and only enforce this property within the fine-grained sub-array columns. Crucially, the small sub-arrays provides a unique opportunity for input zero-skipping, which can significantly avoid unnecessary computations and reduce computation time.

At the same time, it also makes the hardware much easier to implement and is less susceptible to non-idealities and noise than coarse-grained architectures.

Putting all together, with the same optimized DNN models, FORMS achieves 1.50Xand 1.93X throughput improvement in terms of GOPs/s x mm^2 and GOPs/W compared to ISAAC, and 1.12X-2.4X speed up in terms of frame per second over optimized ISAAC with almost the same power/area cost.

Interestingly, FORMS optimization framework can even speed up the original ISAAC from 10.7X up to 377.9Z, reflecting the importance of software/hardware co-design optimizations.

08:45 - 09:00PM BOSS: Bandwidth-Optimized Search Accelerator for Storage-Class Memory

Jun Heo, Seungyul Lee, Sunhong Min, Yeonhong Park, Sung Jun Jung, Tae Jun Ham, Jae W. Lee (SNU)

Abstract:

Search is one of the most popular and important web services. The inverted index is the standard data structure adopted by most full-text search engines. Recently, custom hardware accelerators for inverted index search have emerged to demonstrate much higher throughput than the conventional CPU or GPU. However, less attention has been paid to addressing the memory capacity pressure with inverted index. The conventional DDRx DRAM memory system significantly increases the system cost to make a terabyte-scale main memory. Instead, a shared memory pool composed of storage-class memory (SCM) devices is a promising alternative for scaling memory capacity at a much lower cost. However, this SCM-based pooled memory poses new challenges caused by the limited bandwidth of both SCM devices and the shared interconnect to the host CPU. Thus, we propose BOSS, the first near-data processing (NDP) architecture for inverted index search on SCM-based pooled memory, which maintains high throughput of query processing in this bandwidthconstrained environment. BOSS mitigates the impact of low bandwidth of SCM devices by employing early-termination search algorithms, reducing the footprint of intermediate data, and introducing a programmable decompression module that can select the best compression scheme for a given inverted index. Furthermore, BOSS includes a top-k selection module in hardware to substantially reduce the host-accelerator bandwidth consumption. Compared to Apache Lucene, a production-grade search engine library, running on 8 CPU cores, BOSS achieves a geomean speedup of 8.1× on various complex query types, while reducing the average energy consumption by 189×.

Session 4B: 08:00 - 09:00PM Session 4B. Data Center

08:00 PM – 09:00 PM (EDT/New York)

05:00 PM (PDT/San Francisco), Tue 02:00 (CEST/Brussels), Tue 08:00 (CST/Beijing)

08:00 - 08:15PM SATORI: Efficient and Fair Resource Partitioning by Sacrificing Short-Term Benefits for Long-Term Gains

Rohan Basu Roy, Tirthak Patel, Devesh Tiwari (Northeastern)

Abstract:

Multi-core architectures have enabled data centers to increasingly co-locate multiple jobs to improve resource utilization and lower the operational cost. Unfortunately, naively co-locating multiple jobs may lead to only a modest increase in system throughput. Worse, some users may observe proportionally higher performance degradation compared to other users co-located on the same physical multi-core system. SATORI is a novel strategy to partition multi-core architectural resources to achieve two conflicting goals simultaneously: increasing system throughput and achieving fairness among the co-located jobs.

08:15 - 08:30PM Confidential Serverless Made Efficient with Plug-In Enclaves

Mingyu Li, Yubin Xia, Haibo Chen (Shanghai Jiao Tong)

Abstract:

Serverless computing has become a fact of life on modern clouds. A serverless function may process sensitive data from clients. Protecting such a function against untrusted clouds using hardware enclave is attractive for user privacy. In this work, we run existing serverless applications in SGX enclave, and observe that the performance degradation can be as high as 5.6xto even 422.6x. Our investigation identifies these slowdowns are related to architectural features, mainly from page-wise enclave initialization. Leveraging insights from our overhead analysis, we revisit SGX hardware design and make minimal modification to its enclave model. We extend SGX with a new primitive—region-wise plugin enclaves that can be mapped into existing enclaves to reuse attested common states amongst functions. By remapping plugin enclaves, an enclave allows in-situ processing to avoid expensive data movement in a function chain. Experiments show that our design reduces the enclave function latency by 94.74- 99.57%, and boosts the autoscaling throughput by 19-179x.

08:30 - 08:45PM Flex: High-Availability Datacenters with Zero Reserved Power

Chaojie Zhang (Chicago); Alok Gautam Kumbhare (Microsoft Research); Ioannis Manousakis (Microsoft); Deli Zhang, Pulkit Misra (Microsoft Research); Rod Assis, Kyle Woolcock, Nithish Mahalingam, Brijesh Warrier, David Gauthier, Lalu Kunnath, Steve Solomon, Osvaldo Morales, Marcus Fontoura (Microsoft); Ricardo Bianchini (Microsoft Research)

Abstract:

Cloud providers, like AWS and Azure Amazon and Microsoft, must guarantee high availability for a large fraction of their workloads. For this reason, they build datacenters with redundant infrastructures for power delivery and cooling. Typically, the redundant resources are reserved for use only during infrastructure failure or maintenance events, so that workload performance and availability do not suffer. Unfortunately, the reserved resources also produce lower power utilization and, consequently, require more datacenters to be built. To address these problems, in this paper we propose "zero-reserved-power" datacenters and the Flex system to ensure that workloads still receive their desired performance and availability. Flex leverages the existence of software-redundant workloads that can tolerate lower infrastructure availability, while imposing minimal (if any) performance degradation for those that require high infrastructure availability. Flex mainly comprises (1) a new offline workload placement policy that reduces stranded power while ensuring safety during failure or maintenance events, and (2) a distributed system that monitors for failures and quickly reduces the power draw while respecting the workloads' requirements, when it detects a failure. Our evaluation shows that Flex produces less than 5% stranded power and increases the number of deployed servers by up to 33%, which translates to hundreds of millions of dollars in construction cost savings per datacenter site. We end the paper with lessons from our experience bringing Flex to production in Microsoft's datacenters.

08:45 - 09:00PM BlockMaestro: Enabling Programmer-Transparent Task-Based Execution in GPU Systems

Amirali Abdolrashidi, Hodjat Asghari Esfeden, Ali Jahanshahi, Kaustubh Singh, Nael Abu-Ghazaleh, Daniel Wong (UC Riverside)

Abstract:

As modern GPU workloads grow in size and complexity, there is an ever-increasing demand for GPU computational power. Emerging workloads contain hundreds or thousands of GPU kernel launches, which incur high overheads, and exhibit data-dependent behavior between kernels, which requires synchronization, leading to GPU under-utilization. Task-based execution models have been proposed to solve these issues, but they require significant programmer effort to port applications to proprietary task-based programming models in order to specify tasks and task dependencies. To address this need, we propose BlockMaestro, a software-hardware solution that combines command queue reordering, kernel-launch-time static analysis, and runtime hardware support to dynamically identify and resolve thread-block level data dependencies between kernels. Through static analysis of memory access patterns at kernel-launch-time, BlockMaestro can extract inter-kernel thread block-level data dependencies. BlockMaestro also introduces kernel pre-launching to reduce the kernel launch overheads experienced by multiple dependent kernels. Correctness is enforced by dynamically resolving thread block-level data dependency at runtime through hardware support. BlockMaestro achieves an average speedup of 51.76% (up to 2.92x) on data-dependent benchmarks, and requires minimal hardware overhead.

09:00 - 09:45PM Session 5A. Security I

09:00 PM — 09:45 PM (EDT/New York)

06:00 PM (PDT/San Francisco), Tue 03:00 (CEST/Brussels), Tue 09:00 (CST/Beijing)

09:00 - 09:15PM Opening Pandora's Box: A Systematic Study of New Ways Microarchitecture Can Leak Private Data

Jose Rodrigo Sanchez Vicarte, Pradyumna Shome, Nandeeka Nayak (UIUC); Caroline Trippel (Stanford); Adam Morrison (Tel Aviv University); David Kohlbrenner (Washington); Christopher Fletcher (UIUC)

Abstract

Microarchitectural attacks have plunged Computer Architecture into a security crisis. Yet, as the slowing of Moore's law justifies the use of ever more exotic microarchitecture, it is likely we have only seen the tip of the iceberg.

To better anticipate this security crisis, this paper performs a systematic security-centric analysis of the Computer Architecture literature. Our rationale is that when implementing current and

future processors, microarchitects will (quite reasonably) look to previously-proposed ideas. Our study uncovers seven classes of microarchitectural optimization with novel security implications,

proposes a conceptual framework through which to study them and demonstrates several proofs-of-concept to show their efficacy. The optimizations we study range from those that leak as much

privacy as Spectre/Meltdown (but without exploiting speculative execution) to those that otherwise undermine security-critical programs in a variety of ways. Many have storied histories—ranging from industry patents to media/3rd party speculation regarding current implementation status to recent renewed interest in the academic community. This paper's goal is to perform an early (hopefully not too late) analysis to inform their development moving forward.

09:15 - 09:30PM I See Dead μops: Leaking Secrets via Intel/AMD Micro-Op Caches

Xida Ren, Logan G. Moody (Virginia); Mohammadkazem Taram (UCSD); Matthew Jordan (Virginia); Dean M. Tullsen (UCSD); Ashish Venkat (Virginia)

Abstract:

Modern Intel, AMD, and ARM processors translate complex instructions into simpler internal micro-ops that are then cached in a dedicated on-chip structure called the microop cache. This work presents an in-depth

characterization study of the micro-op cache, reverse-engineering many undocumented features, and further describes attacks that exploit the microop cache as a timing channel to transmit secret information.

In particular, this paper describes three attacks – (1) a same thread cross-domain attack that leaks secrets across the userkernel boundary, (2) a cross-SMT thread attack that transmits secrets across two SMT threads via the micro-op cache, and (3) transient execution attacks that have the ability to leak an unauthorized secret accessed along a misspeculated path, even before the transient instruction is dispatched to execution, breaking several existing invisible speculation and fencing-based solutions that mitigate Spectre.

09:30 - 09:45PM TimeCache: Using Time to Eliminate Cache Side Channels when Sharing Software

Divya Ojha, Sandhya Dwarkadas (Rochester)

Abstract

Timing side channels have been used to extract cryptographic keys and sensitive documents even from trusted enclaves. Specifically, cache side channels created by reuse of shared code or data in the memory hierarchy have been exploited by several known attacks, e.g., evict+reload for recovering an RSA key and Spectre variants for leaking speculatively loaded data.

In this paper, we present TimeCache, a cache design that incorporates knowledge of prior cache line access to eliminate cache side channels due to reuse of shared software (code and data). Our goal is to retain the benefits of a shared cache of allowing each process access to the entire cache and of cache occupancy by a single copy of shared software. We achieve our goal by implementing per-process cache line visibility so that the processes do not benefit from cached data brought in by another process until they have incurred a corresponding miss penalty. Our design achieves low overhead by using a novel combination of timestamps and a hardware design to allow efficient parallel comparisons of the timestamps. The solution works at all the cache levels without the need to limit the number of security domains, and defends against an attacker process running on the same core, on a another hyperthread, or on another core.

Our implementation in the gem5 simulator demonstrates that the system is able to defend against RSA key extraction. We evaluate performance using SPEC2006 and PARSEC and observe the overhead of TimeCache to be 1.13% on average. Delay due to first access misses adds the majority of the overhead, with the security context bookkeeping incurred at the time of a context switch contributing 0.02% of the 1.13%.

09:00 - 09:45PM Session 5B. Accelerators I

09:00 PM – 09:45 PM (EDT/New York)

06:00 PM (PDT/San Francisco), Tue 03:00 (CEST/Brussels), Tue 09:00 (CST/Beijing)

09:00 - 09:15PM Accelerated Seeding for Genome Sequence Alignment with Enumerated Radix Trees

Arun Subramaniyan, Jack Wadden, Kush Goliya, Nathan Ozog, Xiao Wu, Satish Narayanasamy, David Blaauw, Reetuparna Das (Michigan)

Abstract:

Read alignment is a time-consuming step in genome sequencing analysis. The most widely used software for read alignment, BWA-MEM, and the recently published faster version BWA-MEM2 are based on the seed-and-extend paradigm for read alignment. The seeding step of read alignment is a major bottleneck contributing ~40% to the overall execution time of BWA-MEM2 when aligning whole human genome reads from the Platinum Genomes dataset. This is because both BWA-MEM and BWA-MEM2 use a compressed index structure called the FMD-Index, which results in high bandwidth requirements, primarily due to its character-by-character processing of reads. For instance, to seed each read (101 DNA base-pairs stored in 37.8 bytes), the FMD-Index solution in BWA-MEM2 requires ~68.5 KB of index data.

We propose a novel indexing data structure named Enumerated Radix Tree (ERT) and design a custom seeding accelerator based on it. ERT improves bandwidth efficiency of BWA-MEM2 by 4.5x while guaranteeing 100% identical output to the original software, and still fitting in 64 GB DRAM. Overall, the proposed seeding accelerator implemented on AWS F1 FPGA (f1.4xlarge) improves seeding throughput of BWA-MEM2 by 3.3x. When combined with seed-extension accelerators, we observe a 2.1x improvement in overall read alignment through-

put over BWA-MEM2. The software implementation of ERT is integrated into BWA-MEM2 (ert branch: https://github.com/bwa-mem2/bwa-mem2/tree/ert) and is open sourced for the benefit of the research community.

09:15 - 09:30PM Aurochs: An Architecture for Dataflow Threads

Matt Vilim, Alexander Rucker, Kunle Olukotun (Stanford)

Abstract:

Data analytics pipelines increasingly rely on databases to select, filter, and pre-process reams of data.

These databases use data structures with irregular control flow like trees and hash tables which map poorly to existing database accelerators, leaving architects with a choice between CPUs---with stagnant performance---or accelerators that handle this complexity by relying on simpler but asymptotically sub-optimal algorithms.

To bridge this gap, we propose Aurochs: a reconfigurable dataflow accelerator (RDA) that matches a CPU asymptotically but outperforms it by over 100x on constant factors.

We introduce a threading model for vector dataflow accelerators that extracts massive parallelism from irregular data structures using lightweight thread contexts.

To implement this model, we add only a sparse scratchpad to an existing database accelerator---increasing area by 5%.

We reformulate common data structures using dataflow threads and evaluate Aurochs on ridesharing queries---outperforming a GPU by 8x.

09:30 - 09:45PM PipeZK: Accelerating Zero-Knowledge Proof with a Pipelined Architecture

Ye Zhang (Peking / Shanghai Tree-Graph Blockchain Research Institute); Shuo Wang (Peking); Xian Zhang (Microsoft Research); Jiangbin Dong (Xi'an Jiaotong University); Xingzhong Mao (Institute for Interdisciplinary Information Core Technology); Fan Long (Toronto); Cong Wang (Imo.vc); Dong Zhou, Mingyu Gao (Tsinghua); Guangyu Sun (Peking)

09:45 - 10:30PM Panel 2: Biological Computing

The biological and life sciences present a wealth of sophisticated and efficient computing substrates and, as a consequence, have been the source of inspiration for next-generation computing. This panel will cover emerging opportunities for research cross-pollination between the life sciences and computing technologies. Discussions will focus on topics ranging from machine learning and neural networks, brain computer interfaces, molecular & DNA computing, to the opportunities that they present for classical computing and acceleration, as well as emerging neuromorphic and quantum computing technologies

Tuesday, June 15th

10:00 - 11:00AM SIGARCH/TCCA Business Meeting

10:00 AM – 11:00 AM (EDT/New York)

07:00 AM (PDT/San Francisco), 16:00 (CEST/Brussels), 22:00 (CST/Beijing)

11:00 - 12:00AM Key Note by Monica Lam

11:00 AM – 12:00 PM (EDT/New York)

08:00 AM (PDT/San Francisco), 17:00 (CEST/Brussels), 23:00 (CST/Beijing)

Genie: An Open Privacy-Preserving Virtual Assistant with Deep Learning

Abstract:

Virtual assistants today provide a proprietary voice interface for over 100,000 skills and are built with a 100,000-strong workforce. This talk presents the Stanford open virtual assistant initiative that uses deep learn-

ing to lower the development cost, improve the scalability and robustness, and to add dialogue capabilities to enhance the user experience. The research results are encapsulated in the Genie toolset to make voice interfaces as easy to build as web interfaces, and can thus accelerate the growth of an open worldwide voice web. In addition, the open-source assistant is federated to protect user privacy; it is distributed with Home Assistant, an open-source local gateway for home IoTs with over 100,000 users.

Bio

Dr. Monica Lam has been a Professor of Computer Science at Stanford University since 1988, and is the Faculty Director of the Stanford Open Virtual Assistant Laboratory. She leads the Genie open virtual assistant project, which aims to advance and democratize voice assistant technology, keep the voice web open, and protect the privacy of consumers.

Prof. Lam is a member of the National Academy of Engineering and an ACM Fellow. She has won numerous best paper awards, and has published over 150 papers on many topics: natural language processing, machine learning, HCI, compilers, computer architecture, operating systems, and high-performance computing. She is a co-author of the "Dragon Book", the definitive text on compiler technology. She received a B.Sc. from University of British Columbia (1980) and a Ph.D. from Carnegie Mellon University (1987).

12:00AM - 01:00PM Session 6A. Compilers

12:00 PM – 1:00 PM (EDT/New York)

09:00 AM (PDT/San Francisco), 18:00 (CEST/Brussels), Tue 00:00 (CST/Beijing)

12:00 - 12:15PM Taming the Zoo: The Unified GraphIt Compiler Framework for Novel Architectures

Ajay Brahmakshatriya (MIT); Emily Furst (Washington); Victor A. Ying, Claire Hsu, Changwan Hong (MIT); Max Ruttenberg (Washington); Yunming Zhang (MIT); Dai Cheol Jung, Dustin Richmod, Michael Taylor (Washington); Julian Shun (MIT); Mark Oskin (Washington); Daniel Sanchez, Saman Amarasinghe (MIT)

Abstract:

We live in a new Cambrian Explosion of hardware devices. The end of conventional processor scaling has driven research and industry practice to explore a new generation of approaches. The old DNA of architecture design, including vectors, threads, shared or private memories, coherence or message passing, dataflow or von Neumann execution, are hybridized together in new and exciting ways. Each new architecture exposes a unique hardware-level API. Performance and energy efficiency are critically dependent on how well programs can use these APIs. One approach is to implement custom libraries for each new hardware architecture and application domain. A more scalable approach is to utilize a portable compiler infrastructure tailored to the application domain that makes it easy to generate efficient code for a diverse set of architectures with minimal porting effort. We propose the Unified GraphIt Compiler framework (UGC), which does exactly this for graph applications. UGC achieves portability with reasonable effort by decoupling the architecture independent algorithm from the architecture-specific schedules and backends. We introduce a new domain-specific intermediate representation, GraphIR, that is key to this decoupling. GraphIR encodes high-level algorithm and optimization information needed for hardware-specific code generation, making it easy to develop different backends (GraphVMs) for diverse architectures, including CPUs, GPUs, and next-generation hardware such as Swarm and the HammerBlade manycore. We also build scheduling language extensions that make it easy to expose optimization decisions like load balancing strategies, blocking for locality, and other data structure choices. We evaluate UGC on five algorithms and 10 input graphs on these 4 distinct architectures and show that UGC enables implementing optimizations that can provide up to 53× speedup over programmer-generated straightforward implementations

12:15 - 12:30PM Supporting Legacy Libraries on Non-Volatile Memory: A User-Transparent Approach

Chencheng Ye (HUST); Yuanchao Xu, Xipeng Shen (NCSU); XIAOFEI LIAO, Hai Jin (HUST); Yan Solihin (UCF)

Abstract

Leveraging sparsity in deep neural network (DNN) models is promising for accelerating model inference. Yet existing GPUs can only leverage the sparsity from weights but not activations, which are dynamic, unpredictable, and hence challenging to exploit. In this work, we propose a novel architecture to efficiently harness the

dual-side sparsity (i.e., weight and activation sparsity). We take a systematic approach to understand the (dis) advantages of previous sparsity-related architectures and propose a novel, unexplored paradigm that combines outer-product computation primitive and bitmap-based encoding format. We demonstrate the feasibility of our design with minimal changes to the existing production-scale inner-product-based Tensor Core. We propose a set of novel ISA extensions and co-design the matrix-matrix multiplication and convolution algorithms, which are the two dominant computation patterns in today's DNN models, to exploit our new dual-side sparse Tensor Core. Our evaluation shows that our design can fully unleash the dual-side DNN sparsity and improve the performance by up to one order of magnitude with small hardware overhead.

12:30 - 12:45PM Execution Dependence Extension (EDE): ISA Support for Eliminating Fences

Thomas Shull (Oracle Labs); Ilias Vougioukas, Nikos Nikoleris, Wendy Elsasser (Arm Research); Josep Torrellas (UIUC)

Abstract:

Fence instructions are a coarse-grained mechanism to enforce the order of instruction execution in an out-of-order pipeline. They are an overkill for cases when only one instruction must wait for the completion of one other instruction. For example, this is the case when performing undo logging in Non-Volatile Memory (NVM) systems: while the update of a variable needs to wait until the corresponding undo log entry is persisted, all other instructions can be reordered. Unfortunately, current ISAs do not provide a way to describe such an execution dependence between two instructions that have no register or memory dependences. As a result, programmers must place fences, which unnecessarily serialize many unrelated instructions.

To remedy this limitation, we propose an ISA extension capable of describing these execution dependences. We call the proposal Execution Dependence Extension (EDE), and add it to Arm's AArch64 ISA. We also present two hardware realizations of EDE that enforce execution dependences at different stages of the pipeline: one in the issue queue (IQ) and another in the write buffer (WB). We implement IQ and WB in a simulator and test them with several NVM applications. Overall, by using EDE with IQ and WB rather than fences, we attain average workload speedups of 18% and 26%, respectively.

12:45 - 01:00PM Hetero-ViTAL: A Virtualization Stack for Heterogeneous FPGA Clusters

Yue Zha, Jing Li (UPenn)

Abstract:

With field-programmable gate arrays (FPGAs) being widely deployed into data centers, an efficient virtualization support is required to fully unleash the potential of cloud FPGAs. Nevertheless, existing FPGA virtualization solutions only support a homogeneous FPGA cluster comprising identical FPGA devices. Representative work such as ViTAL provides sufficient system support for scale-out acceleration and improves the overall resource utilization through a fine-grained spatial sharing. While these existing solutions (including ViTAL) can efficiently virtualize a homogeneous cluster, it is hard to extend them to virtualizing a heterogeneous cluster which comprises multiple types of FPGAs. We expect the future cloud FPGAs are likely to be more heterogeneous due to hardware rolling upgrade.

In this paper, we rethink FPGA virtualization from ground up and propose Hetero-ViTAL to virtualize heterogeneous FPGA clusters. We identify the conflicting requirements of runtime management and offline compilation when designing the abstraction for a heterogeneous cluster, which is also the fundamental reason why the single-level abstraction as proposed in ViTAL (and other prior works) cannot be trivially extended to the heterogeneous case. To decouple these conflicting requirements, we provide a two-level system abstraction in Hetero-ViTAL. Specifically, the high-level abstraction is FPGA-agnostic and provides a simple and homogeneous view of the FPGA resources to simplify the runtime management. On the contrary, the low-level abstraction is FPGA-specific and exposes sufficient spatial resource constraints to the compilation framework to ensure the mapping quality. Rather than simply adding a layer on top of the single-level abstraction as proposed in ViTAL and other prior work, we judiciously determine how much hardware details should be exposed at each level to balance the management complexity, mapping quality and compilation cost. We then develop a compilation framework to map applications onto this two-level abstraction with several optimization techniques to further improve the mapping quality. We also provide a runtime management policy to alleviate the fragmentation issue, which becomes more severe in a heterogeneous cluster due to the distinct resource capacities of diverse FPGAs.

We evaluate Hetero-ViTAL on a custom-built FPGA cluster and demonstrate its effectiveness using machine learning and image processing applications. Results show that Hetero-ViTAL reduces the average response

time (a critical metric for QoS) by 79.2% for a heterogeneous cluster compared to the non-virtualized baseline. When virtualizing a homogeneous cluster, Hetero-ViTAL also reduces the average response time by 42.0% compared with ViTAL due to a better system design.

12:00AM - 01:00PM Session 6B. Memory II

12:00 PM – 01:00 PM (EDT/New York)

09:00 AM (PDT/San Francisco), 18:00 (CEST/Brussels), Tue 00:00 (CST/Beijing)

12:00 - 12:15PM CODIC: A Low-Cost Substrate for Enabling Custom In-DRAM Functionalities and Optimizations

Lois Orosa (ETH Zurich); Yaohua Wang (NUDT); Mohammad Sadrosadati (IPM); Jeremie Kim, Haocong Luo, Kaveh Razavi, Juan Gómez Luna, Ivan Puddu, Hasan Hassan, Minesh Patel, Nika Mansouri Ghiasi (ETH Zurich); Saugata Ghose (UIUC); Onur Mutlu (ETH Zurich)

12:15 - 12:30PM NVOverlay: Enabling Efficient and Scalable High-Frequency Snapshotting to NVM

Ziqi Wang (CMU); Chul-Hwan Choo (Samsung); Michael A. Kozuch (Intel Labs / CMU); Todd C. Mowry (CMU); Gennady Pekhimenko (Toronto); Vivek Seshadri (Microsoft Research India); Dimitrios Skarlatos (CMU)

Abstract:

The ability to capture frequent (per millisecond) persistent snapshots to NVM would enable a number of compelling use cases. Unfortunately, existing NVM snapshotting techniques suffer from a combination of persistence barrier stalls, write amplification to NVM, and/or lack of scalability beyond a single socket. In this paper, we present NVOverlay, which is a scalable and efficient technique for capturing frequent persistent snapshots to NVM such that they can be randomly accessed later. NVOverlay uses Coherent Snapshot Tracking to efficiently track changes to memory (since the previous snapshot) across multisocket parallel systems, and it uses Multi-snapshot NVM Mapping to store these snapshots to NVM while avoiding excessive write amplification. Our experiments demonstrate that NVOverlay successfully hides the overhead of capturing these snapshots while reducing write amplification by 29%-47% compared with state of-the-art logging-based snapshotting techniques.

12:30 - 12:45PM Rebooting Virtual Memory with Midgard

Siddharth Gupta, Atri Bhattacharyya, Yunho Oh (EPFL); Abhishek Bhattacharjee (Yale); Babak Falsafi, Mathias Payer (EPFL)

Abstract:

Computer systems designers are building cache hierarchies with higher capacity to capture the ever-increasing working sets of modern workloads. Cache hierarchies with higher capacity improve system performance but shift the performance bottleneck to address translation. We propose Midgard, an intermediate address space between the virtual and the physical address spaces, to mitigate address translation overheads without program-level changes.

Midgard leverages the operating system concept of virtual memory areas (VMAs) to realize a single Midgard address space where VMAs of all processes can be uniquely mapped. The Midgard address space serves as the namespace for all data in a coherence domain and the cache hierarchy. Because real-world workloads use far fewer VMAs than pages to represent their virtual address space, virtual to Midgard translation is achieved with hardware structures that are much smaller than TLB hierarchies. Costlier Midgard to physical address translations are needed only on LLC misses, which become much less frequent with larger caches. As a consequence, Midgard shows that instead of amplifying address translation overheads, memory hierarchies with large caches can reduce address translation overheads.

Our evaluation shows that Midgard achieves only 5% higher address translation overhead as compared to traditional TLB hierarchies for 4KB pages when using a 16MB aggregate LLC. Midgard also breaks even with traditional TLB hierarchies for 2MB pages when using a 256MB aggregate LLC. For cache hierarchies with higher capacity, Midgard's address translation overhead drops to near zero as secondary and tertiary data working sets fit in the LLC, while traditional TLBs suffer even higher degrees of address translation overhead.

12:45 - 01:00PM Dve: Improving DRAM Reliability and Performance On-Demand via Coherent Replication

Adarsh Patil, Vijay Nagarajan (Edinburgh); Rajeev Balasubramonian (Utah); Nicolai Oswald (Edinburgh)

01:00 - 01:45PM Session 7A. Accelerators II

01:00 PM – 01:45 PM (EDT/New York)

10:00 AM (PDT/San Francisco), 19:00 (CEST/Brussels), Tue 01:00 (CST/Beijing)

01:00 - 01:15PM Enabling Compute-Communication Overlap in Distributed Deep Learning Training Platforms

Saeed Rashidi, Matthew Denton (Georgia Tech); Srinivas Sridharan (Facebook); Sudarshan Srinivasan (Intel); Amoghavarsha Suresh (Stony Brook); Jade Nie (Facebook); Tushar Krishna (Georgia Tech)

Abstract:

Deep Learning (DL) training platforms are built by interconnecting multiple DL accelerators (e.g., GPU/TPU) via fast, customized interconnects with 100s of gigabytes (GBs) of bandwidth. However, as we identify in this work, driving this bandwidth is quite challenging. This is because there is a pernicious balance between using the accelerator's compute and memory for both DL computations and communication. This work makes two key contributions. First, via real system measurements and detailed modeling, we provide an understanding of compute and memory bandwidth demands for DL compute and comms. Second, we propose a novel DL collective communication accelerator called Accelerator Collectives Engine (ACE) that sits alongside the compute and networking engines at the accelerator endpoint. ACE frees up the endpoint's compute and memory resources for DL compute, which in turn reduces the required memory BW by 3.5× on average to drive the same network BW compared to state-of-the-art baselines. For modern DL workloads and different network sizes, ACE, on average, increases the effective network bandwidth utilization by 1.44× (up to 2.67×), resulting in an average of 1.41× (up to 1.51×), 1.12× (up to 1.17×), and 1.13× (up to 1.19×) speedup in iteration time for ResNet-50, GNMT and DLRM when compared to the best baseline configuration, respectively.

01:15 - 01:30PM CoSA: Scheduling by Constrained Optimization for Spatial Accelerators

Qijing Huang, Minwoo Kang, Grace Dinh, Thomas Norell (Berkeley); Aravind Kalaiah (Facebook); James Demmel, John Wawrzynek, Yakun Sophia Shao (Berkeley)

Abstract:

Recent advances in Deep Neural Networks (DNNs) have led to active development of specialized DNN accelerators, many of which feature a large number of processing elements laid out spatially, together with a multi-level memory hierarchy and flexible interconnect. While DNN accelerators can take advantage of data reuse and achieve high peak throughput, they also expose a large number of runtime parameters to the programmers who need to explicitly manage how computation is scheduled both spatially and temporally. In fact, different scheduling choices can lead to wide variations in performance and efficiency, motivating the need for a fast and efficient search strategy to navigate the vast scheduling space.

To address this challenge, we present CoSA, a constrained-optimization-based approach for scheduling DNN accelerators. As opposed to existing approaches that either rely on designers' heuristics or iterative methods to navigate the search space, CoSA expresses scheduling decisions as a constrained-optimization problem that can be deterministically solved using mathematical optimization techniques. Specifically, CoSA leverages the regularities in DNN operators and hardware to formulate the DNN scheduling space into a mixed-integer programming (MIP) problem with algorithmic and architectural constraints, which can be solved to automatically generate a highly efficient schedule in one shot. We demonstrate that CoSA-generated schedules significantly outperform state-of-the-art approaches by a geometric mean of up to 2.5x across a wide range of DNN networks while improving the time-to-solution by 90x.

01:30 - 01:45PM η-LSTM: Co-Designing Highly-Efficient Large LSTM Training via Exploiting Memory-Saving and Architectural Design Opportunities

Xingyao Zhang (Washington / Houston); Haojun Xia, Donglin Zhuang, Hao Sun (Sydney); Xin Fu (Houston); Michael Taylor (Washington); Shuaiwen Leon Song (Sydney)

Abstract:

Recently, the recurrent neural network, or its most popular type---the Long Short Term Memory (LSTM) network---has achieved great success in a broad spectrum of real-world application domains, such as autonomous driving, natural language processing, sentiment analysis, and epidemiology. Due to the complex features of the real-world tasks, current LSTM models become increasingly bigger and more complicated for enhancing the learning ability and prediction accuracy. However, through our in-depth characterization on the state-of-the-art general-purpose deep-learning accelerators, we observe that the LSTM training execution grows inefficient in terms of storage, performance, and energy consumption, under an increasing model size. With further algorithmic and architectural analysis, we identify the root cause for large LSTM training inefficiency: massive intermediate variables. To enable a highly-efficient LSTM training solution for the ever-growing model size, we exploit some unique memory-saving and performance improvement opportunities from the LSTM training procedure, and leverage them to propose the first cross-stack training solution, eta-LSTM, for large LSTM models. eta-LSTM comprises both software-level and hardware-level innovations that effectively lower the memory footprint upper-bound and excessive data movements during large LSTM training, while also drastically improving training performance and energy efficiency. Experimental results on six real-world large LSTM training benchmarks demonstrate that eta-LSTM reduces the required memory footprint by an average of 57.5% (up to 75.8%) and brings down the data movements for weight matrices, activation data, and intermediate variables by 40.9%, 32.9%, and 80.0%, respectively. Furthermore, it outperforms the state-of-the-art GPU implementation for LSTM training by an average of 3.99x (up to 5.73x) on performance and 2.75x (up to 4.25x) on energy. We hope this work can shed some light on how to design high logic utilization for future NPUs.

01:00 - 01:45PM Session 7B. Graph Processing

01:00 PM – 01:45 PM (EDT/New York)

10:00 AM (PDT/San Francisco), 19:00 (CEST/Brussels), Tue 01:00 (CST/Beijing)

01:00 - 01:15PM FlexMiner: A Pattern-Aware Accelerator for Graph Pattern Mining

Xuhao Chen, Tianhao Huang, Shuotao Xu, Thomas Bourgeat, Chanwoo Chung, Arvind (MIT)

Abstract:

Graph pattern mining (GPM) is a class of algorithms widely used in many real-world applications in bio-medicine, ecommerce, security, social sciences, etc. GPM is a computationally intensive problem with an enormous amount of coarse-grain parallelism and therefore, attractive for hardware acceleration. Unfortunately, existing GPM accelerators have not used the best known algorithms and optimizations, and thus offer questionable benefits over software implementations.

We present FlexMiner, a software/hardware co-designed GPM accelerator that improves the efficiency without compromising the generality or productivity of state-of-the-art software GPM frameworks. FlexMiner exploits massive amount of coarse-grain parallelism in GPM by deploying a large number of specialized processing elements. For efficient searches, the FlexMiner hardware accepts pattern-specific execution plans, which are generated automatically by the FlexMiner compiler from the given pattern(s). To avoid repetitive computation on neighborhood connectivity, we provide dedicated on-chip storage to memoize reusable connectivity information in a connectivity map (c-map) which is implemented with low-cost yet high-throughput hardware. The on-chip memories in FlexMiner are managed dynamically using heuristics derived by the compiler, and thus are fully utilized. We have evaluated FlexMiner with 4 GPM applications on a wide range of real-world graphs. Our cycle-accurate simulation shows that FlexMiner with 64 PEs achieves 10.6× speedup on average over the state-of-the-art software system executing 20 threads on a 10-core Intel CPU.

01:15 - 01:30PM PolyGraph: Exposing the Value of Flexibility for Graph Processing Accelerators

Vidushi Dadu, Sihao Liu, Tony Nowatzki (UCLA)

Abstract:

Because of the importance of graph workloads and the limitations of CPUs/GPUs, many graph processing accelerators have been proposed. The basic approach of prior accelerators is to focus on a single graph algorithm variant (eg. Bulk-synchronous + slicing). While helpful for specialization, this leaves performance potential

from flexibility on the table and complicates understanding the relationship between graph types, workloads, algorithms, and specialization. In this work, we explore the value of flexibility in graph processing accelerators. First, we identify a taxonomy of key algorithm variants. Then we develop a template architecture (PolyGraph) that is flexible across these variants while being able to modularly integrate specialization features for each. Overall, we find that flexibility in graph acceleration is critical. If only one variant can be supported, asynchronous updates/priority-vertex-scheduling/graph-slicing is the best design, achieving 1.93x speedup over the best-performing accelerator, GraphPulse. However, static flexibility per-workload can further improve performance by 2.71x. With dynamic flexibility per-phase, performance further improves by up to 50%.

01:30 - 01:45PM Large-Scale Graph Processing on FPGAs with Caches for Thousands of Simultaneous Misses

Mikhail Asiatici, Paolo Ienne (EPFL)

01:45 - 02:30PM Panel 3: Quantum Computing

1:45 PM – 2:30 PM (EDT/New York)

10:45 AM (PDT/San Francisco), 19:45 (CEST/Brussels), Tue 01:45 (CST/Beijing)

Quantum computers promise to solve a class of commercial and scientifically important problems that are beyond the abilities of classical computers. Computing, whether it is conventional or quantum, is ultimately a series of transformations, ranging from algorithms at the top, to devices at the bottom. Over the past three decades, there has been significant progress in the field of quantum algorithms (which relies on mathematical properties of quantum states) and quantum devices (which relies on physical properties of materials), however, the role of computer systems (which transforms mathematics into physics) has only recently started to gain prominence.

This panel will discuss the role and the challenges for the architecture and compiler community in making quantum computing practical.

08:00 - 09:00PM Session 8A. Low Temperature / Low Energy Computing

08:00 PM – 09:00 PM (EDT/New York)

05:00 PM (PDT/San Francisco), Tue 02:00 (CEST/Brussels), Tue 08:00 (CST/Beijing)

08:00 - 08:15PM Cost-Efficient Overclocking in Immersion-Cooled Datacenters

Majid Jalili (UT Austin); Ioannis Manousakis, Íñigo Goiri, Pulkit Misra, Ashish Raniwala, Husam Alissa, Bharath Ramakrishnan (Microsoft); Phillip Tuma (3M); Christian Belady, Marcus Fontoura, Ricardo Bianchini (Microsoft)

Abstract

Cloud providers typically use air-based solutions for cooling servers in datacenters. However, increasing transistor counts and the end of Dennard scaling will result in chips with thermal design power that exceeds the capabilities of air cooling in the near future. Consequently, providers have started to explore liquid cooling solutions (e.g., cold plates, immersion cooling) for the most power-hungry workloads. By keeping the servers cooler, these new solutions enable providers to operate server components beyond the normal frequency range (i.e.,overclocking them) all the time. Still, providers must tradeoff the increase in performance via overclocking with its higher power draw and any component reliability implications.

In this paper, we argue that two-phase immersion cooling (2PIC) is the most promising technology, and build three prototype 2PIC tanks. Given the benefits of 2PIC, we characterize the impact of overclocking on performance, power, and reliability. Moreover, we propose several new scenarios for taking advantage of overclocking in cloud platforms, including oversubscribing servers and virtual machine (VM) auto-scaling. For the autoscaling scenario, we build a system that leverages overclocking for either hiding the latency of VM creation or postponing the VM creations in the hopes of not needing them. Using realistic cloud workloads running on a tank prototype, we show that overclocking can improve performance by 20%, increase VM packing density by 20%, and improve tail latency in auto-scaling scenarios by 54%. The combination of 2PIC and overclocking can reduce platform cost by up to 13% compared to air cooling.

08:15 - 08:30PM CryoGuard: A Near Refresh-Free Robust DRAM Design for Cryogenic Computing

Gyu-Hyeon Lee, Seongmin Na, Il-Kwon Byun, Dongmoon Min, Jangwoo Kim (SNU)

Abstract

Cryogenic computing, which runs a computer device at an extremely low temperature, is highly promising thanks to the significant reduction of the wire latency and leakage current. A recently proposed cryogenic DRAM design achieved the promising performance improvement, but it also reveals that it must reduce the DRAM's dynamic power to overcome the huge cooling cost at 77K. Therefore, researchers now target to reduce the cryogenic DRAM's refresh power by utilizing its significantly increased retention time driven by the reduced leakage current. To achieve the goal, however, architects should first answer many fundamental questions regarding the reliability and then design a refresh-free, but still robust cryogenic DRAM by utilizing the analysis result.

In this work, we propose a near refresh-free, but robust cryogenic DRAM (NRFC-DRAM), which can almost eliminate its refresh overhead while ensuring reliable operations at 77K. For the purpose, we first evaluate various DRAM samples of multiple vendors by conducting a thorough analysis to accurately estimate the cryogenic DRAM's retention time and reliability. Our analysis identifies a new critical challenge such that reducing DRAM's refresh rate can make the memory highly unreliable because normal memory operations can now appear as row-hammer attacks at 77K. Therefore, NRFC-DRAM requires a cost-effective, cryogenic-friendly protection mechanism against the new row-hammer-like ``faults'' at 77K.

To resolve the challenge, we present CryoGuard, our cryogenic-friendly row-hammer protection method to ensure the NRFC-DRAM's reliable operations at 77K. With CryoGuard applied, NRFC-DRAM reduces the overall power consumption by 25.9% even with its cooling cost included, whereas the existing cryogenic DRAM fails to reduce the power consumption.

08:30 - 08:45PM Superconducting Computing with Alternating Logic Elements

Georgios Tzimpragos (UCSB); Jennifer Volk (UCSB); Alexander Wynn (MIT Lincoln Lab); James Smith; Tim Sherwood (UCSB)

Abstract

Although superconducting single flux quantum (SFQ) technologies offer the potential for low-latency operation with energy dissipation of the order of attojoules per gate, their inherently pulse-driven nature and stateful cells have led to designs in which every logic gate is clocked. This means that clocked buffers must be added to equalize logic path lengths, and every gate becomes a pipeline stage. We propose a different approach, where gates are clock-free and synchronous designs have a conventional look-and-feel. Despite being clock-free, however, the gates are state machines by nature. To properly manage these state machines, the logical clock cycle is composed of two synchronous alternating phases: the first of which implements the desired function, and the second of which returns the state machines to the ground state. Moreover, to address the challenges associated with the asynchronous implementation of Boolean NOT operations in pulse-based systems, values are represented as unordered binary codes -- in particular, dual-rail codes. With unordered codes, AND and OR operations are functionally complete.

We demonstrate that our new approach, xSFQ, with its dual-rail construction and alternating clock phases, along with ``double-pumped'' logical latches and a timing optimization through latch decomposition, is capable of implementing arbitrary digital designs without gate-level pipelining and the overheads that come with it. We evaluate energy-delay trade-offs enabled by this approach through a mix of detailed analog circuit modeling, pulse-level discrete-event simulation, and high-level pipeline efficiency analysis. The resulting systems are shown to deliver energy-delay product (EDP) gains over conventional SFQ even with pipeline hazard ratios (HR) below 1%. For hazard ratios equal to 15% and 20% and a design resembling a RISC-V RV32I core (excluding the cost of interlock logic), xSFQ achieves 22x and 31x EDP savings, respectively.

08:45 - 09:00PM Failure Sentinels: Ubiquitous Just-in-Time Intermittent Computation via Low-Cost Hardware Support for Voltage Monitoring

Harrison Williams, Michael Moukarzel, Matthew Hicks (Virginia Tech)

08:00 - 09:00PM Session 8B. Machine Learning II

08:00 PM – 9:00 PM (EDT/New York)

05:00 PM (PDT/San Francisco), Tue 02:00 (CEST/Brussels), Tue 08:00 (CST/Beijing)

08:00 - 08:15PM SPACE: Locality-Aware Processing in Heterogeneous Memory for Personalized Recommendations

Hongju Kal, Seokmin Lee, Gun Ko, Won Woo Ro (Yonsei)

Abstract:

Personalized recommendation systems have become a major AI application in modern data centers.

The main challenges in processing personalized recommendation inferences are the large memory footprint and high bandwidth requirement of embedding layers.

To overcome the capacity limit and bandwidth congestion of on-chip memory, near memory processing (NMP) can be a promising solution.

Recent work on accelerating personalized recommendations proposes a DIMM based NMP design to solve the bandwidth problem and increases memory capacity.

The performance of NMP is determined by the internal bandwidth and the prior DIMM-based approach utilizes more DIMMs to achieve higher operation throughput.

However, extending the number of DIMMs could eventually lead to significant power consumption due to inefficient scaling.

We propose SPACE, a novel heterogeneous memory architecture, which is efficient in terms of performance and energy.

SPACE exploits a compute-capable 3D-stacked DRAM with DIMMs for personalized recommendations.

Prior to designing the proposed system, we give a quantitative analysis of the user/item interactions and define the two localities: gather locality and reduction locality.

In gather operations, we find only a small proportion of items are highly-accessed by users, and we call this gather locality.

Also, we define reduction locality as the reusability of the gathered items in reduction operations.

Based on the gather locality, SPACE allocates highly-accessed embedding items to the 3D-stacked DRAM to achieve the maximum bandwidth.

Subsequently, by exploiting reduction locality, we utilize the remaining space of the 3D-stacked DRAM to store and reuse repeated partial sums, thereby minimizing the required number of element-wise reduction operations.

As a result, the evaluation shows that SPACE achieves 3.2× performance improvement and 56% energy saving over the previous DIMM-based NMPs leveraging 3D-stacked DRAM with a 1/8 size of DIMMs.

Also, compared to the state-of-the-art DRAM cache designs with the same NMP configuration, SPACE achieves an average 32.7% of performance improvement.

08:15 - 08:30PM ELSA: Hardware-Software Co-Design for Efficient, Lightweight Self-Attention Mechanism in Neural Networks

Tae Jun Ham, Yejin Lee, Seong Hoon Seo, Soosung Kim, Hyunji Choi, Sung Jun Jung, Jae W. Lee (SNU)

Abstract:

The self-attention mechanism is rapidly emerging as one of the most important key primitives in neural networks (NNs) for its ability to identify the relations within input entities. The self-attention-oriented NN models such as Google Transformer and its variants have established the state-of-the-art on a very wide range of natural language processing tasks, and many other self-attention-oriented models are achieving competitive results in computer vision and recommender systems as well. Unfortunately, despite its great benefits, the self-attention mechanism is an expensive operation whose cost increases quadratically with the number of input entities that it processes, and thus accounts for a significant portion of the inference runtime. Thus, this paper presents ELSA (Efficient, Lightweight Self-Attention), a hardware-software co-designed solution to substantially reduce the runtime as well as energy spent on the self-attention mechanism. Specifically, based on the intuition that not all relations are equal, we devise a novel approximation scheme that significantly reduces the amount of computation by efficiently filtering out relations that are unlikely to affect the final output. With the specialized hardware for this approximate self-attention mechanism, ELSA achieves a geomean speedup of 58.1× as well as over three orders of magnitude improvements in energy efficiency compared to GPU on self-attention computation in modern NN models while maintaining less than 1% loss in the accuracy metric.

08:30 - 08:45 Cambricon-Q: A Hybrid Architecture for Efficient Training

Yongwei Zhao, Chang Liu, Zidong Du, Qi Guo, Xing Hu, Yimin Zhuang, Zhenxing Zhang, Xinkai Song, Wei Li, Xishan Zhang (ICT, CAS); Ling Li (Institute of Software, CAS); Zhiwei Xu (ICT, CAS), Tianshi Chen (Cambricon)

08:45 - 09:00PM TENET: A Framework for Modeling Tensor Dataflow Based on Relation-Centric Notation

Liqiang Lu (Peking); Naiqing Guan (Toronto); Yuyue Wang, Liancheng Jia, Zizhang Luo (Peking); Jieming Yin (Lehigh); Jason Cong (UCLA); Yun Liang (Peking)

09:00 - 09.45PM Session 9A. Memory III

9:00 PM – 9:45 PM (EDT/New York)

06:00 PM (PDT/San Francisco), Tue 03:00 (CEST/Brussels), Tue 09:00 (CST/Beijing)

09:00 - 09:15PM Ripple: Profile-Guided Instruction Cache Replacement for Data Center Applications

Tanvir Ahmed Khan (Michigan); Dexin Zhang (USTC); Akshitha Sriraman (Michigan); Joseph Devietti (UPenn); Gilles A. Pokam (Intel); Heiner Litz (UCSC); Baris Kasikci (Michigan)

09:15 - 09:30PM Quantifying Server Memory Frequency Margin and Using It to Improve Performance in HPC Systems

Da Zhang, Gagandeep Panwar (Virginia Tech); Jagadish Kotra (AMD Research); Nathan DeBardeleben, Sean Blanchard (Los Alamos); Xun Jian (Virginia Tech)

Abstract

Leveraging sparsity in deep neural network (DNN) models is promising for accelerating model inference. Yet existing GPUs can only leverage the sparsity from weights but not activations, which are dynamic, unpredictable, and hence challenging to exploit. In this work, we propose a novel architecture to efficiently harness the dual-side sparsity (i.e., weight and activation sparsity). We take a systematic approach to understand the (dis) advantages of previous sparsity-related architectures and propose a novel, unexplored paradigm that combines outer-product computation primitive and bitmap-based encoding format. We demonstrate the feasibility of our design with minimal changes to the existing production-scale inner-product-based Tensor Core. We propose a set of novel ISA extensions and co-design the matrix-matrix multiplication and convolution algorithms, which are the two dominant computation patterns in today's DNN models, to exploit our new dual-side sparse Tensor Core. Our evaluation shows that our design can fully unleash the dual-side DNN sparsity and improve the performance by up to one order of magnitude with small hardware overhead.

09:30- 09:45PM Revamping Storage Class Memory With Hardware Automated Memory-Over-Storage Solution

Jie Zhang, Miryeong Kwon, Donghyun Gouk, Sungjoon Koh (KAIST); Nam Sung Kim (UIUC); Mahmut Taylan Kandemir (Penn State); Myoungsoo Jung (KAIST)

Abstract:

Large persistent memories such as NVDIMM have been perceived as a disruptive memory technology, because they can maintain the state of a system even after a power failure and allow the system to recover quickly. However, overheads incurred by a heavy software-stack intervention seriously negate the benefits of such memories. First, to significantly reduce the software stack overheads, we propose HAMS, a hardware automated Memory-over-Storage (MoS) solution. Specifically, HAMS aggregates the capacity of NVDIMM and ultra-low latency flash archives (ULL-Flash) into a single large memory space, which can be used as a working or persistent memory expansion, in an OS-transparent manner. HAMS resides in the memory controller hub and manages its MoS address pool over conventional DDR and NVMe interfaces; it employs a simple hardware cache to serve all the memory requests from the host MMU after mapping the storage space of

ULL-Flash to the memory space of NVDIMM. Second, to make HAMS more energy-efficient and reliable, we propose an "advanced HAMS" which removes unnecessary data transfers between NVDIMM and ULL-Flash after optimizing the datapath and hardware modules of HAMS. This approach unleashes the ULL-Flash and its NVMe controller from the storage box and directly connects the HAMS datapath to NVDIMM over the conventional DDR4 interface. Our evaluations show that HAMS and advanced HAMS can offer 97% and 119% higher system performance than a software-based hybrid NVDIMM design, while consuming 41% and 45% lower system energy, respectively.

09:00 - 09:45PM Session 9B. Network Storage Acceleration

09:00 PM – 09:45 PM (EDT/New York)

06:00 PM (PDT/San Francisco), Tue 03:00 (CEST/Brussels), Tue 09:00 (CST/Beijing)

09:00 - 09:15PM NASGuard: A Novel Accelerator Architecture for Robust Neural Architecture Search (NAS) Networks

Xingbin Wang (State Key Laboratory of Information Security, Institute of Information Engineering, CAS); Boyan Zhao and Rui Hou (Institute of Information Engineering, CAS); Amro Awad (NCSU / UCF); Zhihong Tian (Guangzhou); Dan Meng (Institute of Information Engineering, CAS)

09:15 - 09:30PM NASA: Accelerating Neural Network Design with a NAS Processor

Xiaohan Ma, Chang Si, Ying Wang (ICT, CAS); Cheng Liu (State Key Laboratory of Computer Architecture, ICT, CAS); Lei Zhang (ICT, CAS)

Abstract:

Neural network search (NAS) projects a promising direction to automate the design process of efficient and powerful neural network architectures. Nevertheless, the NAS techniques have to dynamically generate a large number of candidate neural networks, and iteratively train and evaluate these online generated network architectures, thus they are extremely time-consuming even when it is deployed on large GPU clusters, which dramatically hinders the adoption of NAS. Though recently there are many specialized architectures proposed to accelerate the training or inference of neural networks, we observe that existing neural network accelerators are typically targeted at static neural network architectures, and they are not suitable to accelerate the evaluation of the dynamical neural network candidates evolving during the NAS process, which cannot be deployed onto current accelerators via the off-line compilation.

To enable rapid and energy-efficient NAS in compact singlechip solutions, we propose NASA, a specialized architecture for one-shot based NAS acceleration. It is able to generate, schedule, and evaluate the candidate neural network architectures for the target machine learning workload with high speed, significantly alleviating the processing bottleneck of one-shot NAS. Motivated by the observation that there are considerable computation sharing opportunities among the different neural network candidates generated in one-shot NAS, NASA is equipped with an on-chip network fusion unit to remove the redundant computation during the network mapping stage. In addition, the NASA accelerator can partition and re-schedule the candidate neural network architectures at fine-granularity to maximize the chance of data reuse and improve the utilization of the accelerator arrays integrated to accelerate network evaluation. According to our experiments on multiple one-shot NAS tasks, NASA achieves 33.52× performance speedup and 214.33× energy consumption reduction on average when compared to a CPU-GPU system.

09:30 - 09:45PM PMNet: In-Network Data Persistence

Korakit Seemakhupt, Sihang Liu, Yasas Senevirathne (Virginia); Muhammad Shahbaz (Stanford); Samira Khan (Virginia)

Abstract

To guarantee data persistence, storage workloads (such as key-value stores and databases) typically use a synchronous protocol that places the network and server stack latency on the critical path of request processing. The use of the fast and byte-addressable persistent memory (PM) has helped mitigate the storage overhead of the server stack; yet, networking is still a dominant factor in the end-to-end latency of request processing. Emerging programmable network devices can reduce network latency by moving parts of the applications' compute into the network (e.g., caching results for read requests); however, for update requests, the client still has to stall on the server to commit the updates, persistently.

In this work, we introduce in-network data persistence that extends the data-persistence domain from servers to the network, and present PMNet, a programmable data plane (e.g., switch or NIC) with PM for persisting data in the network. PMNet logs incoming update requests and acknowledges clients directly without having them wait on the server to commit the request. In case of a failure, the logged requests act as redo logs for the server to recover. We implement PMNet on an FPGA and evaluate its performance using common PM workloads, including key-value stores and PM-backed applications. Our evaluation shows that PMNet can improve the throughput of update requests by 4.31x on average, and the 99th-percentile tail latency by 3.23x.

09:45 - 10:30PM Panel 4: Reserach Methodology

The new era of computer architecture heavily focuses on cross-stack system design with heterogeneous accelerators and new memory and storage systems. This change brings the opportunity and excitement of innovating new systems, but also introduces the challenge of building tools and system components necessary to evaluate radically new designs. The question we will discuss on this panel is the following: how should the architecture community propose, validate, and prototype ideas in this new era of computer architecture to maximize the impact?

Wednesday, June 16th

10:00 - 11:00AM Key Note by Pradeep Dubey

10:00 AM – 11:00 AM (EDT/New York): Keynote by Pradeep Dubey

07:00 AM (PDT/San Francisco), 16:00 (CEST/Brussels), 22:00 (CST/Beijing)

The Era of Ubiquitous AI: A Call-to- Arms for Architects

Abstract:

Artificial intelligence (AI) is touching, if not transforming, every aspect of our lives. Fast-evolving AI algorithms are driving demand for general-purpose computing that cannot be met by "business as usual" engineering. At the same time, programmers are often data scientists, not computer scientists; expecting programmers to figure out increasingly complex hardware on their own just doesn't work. Architects are therefore needed more than ever – chip architects to create new processors, systems architects to design new data centers, software architects to design new frameworks, and AI architects to churn out new models and new algorithms. Are we up to the task? Or do we need to augment human architects with AI to meet the challenge?

Bio

Pradeep K. Dubey is an Intel Senior Fellow and director of the Parallel Computing Lab, a part of the Intel Labs organization at Intel Corporation. He leads a team of top researchers focused on state-of-the-art research in parallel computing. Dubey and his team are responsible for defining computer architectures that can efficiently handle emerging machine learning/artificial intelligence, traditional HPC applications for data-centric computing environments, and deriving product differentiation opportunities for Intel's CPU and GPU processing platforms. Dubey previously worked at IBM's T.J. Watson Research Center. Dubey has made significant contributions to the design, architecture and application performance of various microprocessors, including the IBM Power PC, the Intel386™, Intel486™, Intel® Pentium®, and Intel Xeon® processors. He holds 36 patents and has published more than 100 peer-reviewed technical papers. In 2012, Dubey was honored with an Intel Achievement Award for breakthroughs in parallel computing research, and was honored with the Outstanding Electrical and Computer Engineer Award from Purdue University in 2014. Dubey holds a Ph.D. in electrical engineering from Purdue University. He is also a Fellow of IEEE

11:00 - 12:00AM Awards Ceremony

11:00 AM – 12:00 PM (EDT/New York)

08:00 AM (PDT/San Francisco), 17:00 (CEST/Brussels), 23:00 (CST/Beijing)

12:00 - 13:00PM Session 10A. Quantum / Photonics

12:00 PM – 1:00 PM (EDT/New York)

09:00 AM (PDT/San Francisco), 18:00 (CEST/Brussels), Tue 00:00 (CST/Beijing)

12:00 - 12:15PM Exploiting Long Distance Interactions and Tolerating Atom Loss in Neutral Atom Quantum Architectures

Jonathan M. Baker, Andrew Litteken, Casey Duckering, Henry Hoffmann, Hannes Bernien, Fred Chong (Chicago)

Abstract

Quantum technologies currently struggle to scale beyond moderate scale prototypes and are unable to execute even reasonably sized programs due to prohibitive gate error rates or coherence times. Many software approaches rely on heavy compiler optimization to squeeze extra value from noisy machines but are fundamentally limited by hardware. Alone, these software approaches help to maximize the use of available hardware but cannot overcome the inherent limitations posed by the underlying technology.

An alternative approach is to explore the use of new, though potentially less developed, technology as a path towards scalability. In this work we evaluate the advantages and disadvantages of a Neutral Atom (NA) architecture. NA systems offer several promising advantages such as long range interactions and native multiqubit gates which reduce communication overhead, overall gate count, and depth for compiled programs. Long range interactions, however, impede parallelism with restriction zones surrounding interacting qubit pairs. We extend current compiler methods to maximize the benefit of these advantages and minimize the cost.

Furthermore, atoms in an NA device have the possibility to randomly be lost over the course of program execution which is extremely detrimental to total program execution time as atom arrays are slow to load. When the compiled program is no longer compatible with the underlying topology, we need a fast and efficient coping mechanism. We propose hardware and compiler methods to increase system resilience to atom loss dramatically reducing total computation time by circumventing complete reloads or full recompilation every cycle.

12:15 - 12:30PM Software-Hardware Co-Optimization for Computational Chemistry on Superconducting Quantum Processors

Gushu Li (UCSB); Yunong Shi (Chicago); Ali Javadi-Abhari (IBM)

Abstract:

Computational chemistry is the leading application to demonstrate the advantage of quantum computing in the near term. However, large-scale simulation of chemical systems on quantum computers is currently hindered due to a mismatch between the computational resource needs of the program and those available in today's technology. In this paper we argue that significant new optimizations can be discovered by co-designing the application, compiler, and hardware. We show that multiple optimization objectives can be coordinated through the key abstraction layer of Pauli strings, which are the basic building blocks of computational chemistry programs. In particular, we leverage Pauli strings to identify critical program components that can be used to compress program size with minimal loss of accuracy. We also leverage the structure of Pauli string simulation circuits to tailor a novel hardware architecture and compiler, leading to significant execution overhead reduction by up to 99%. While exploiting the high-level domain knowledge reveals significant optimization opportunities, our hardware/software framework is not tied to a particular program instance and can accommodate the full family of computational chemistry problems with such structure. We believe the co-design lessons of this study can be extended to other domains and hardware technologies to hasten the onset of quantum advantage.

12:30 - 12:45PM Designing Calibration and Expressivity-Efficient Instruction Sets for Quantum Computing

Lingling Lao (University College London); Prakash Murali, Margaret R. Martonosi (Princeton); Dan Browne (University College London)

Abstract:

Near-term quantum computing (QC) systems have limited qubit counts, high gate (instruction) error rates, and typically support a minimal instruction set having one type of two-qubit gate (2Q).

To reduce program instruction counts and improve application expressivity, vendors have proposed, and shown proof-of-concept demonstrations of richer instruction sets such as XY gates (Rigetti) and fSim gates (Google).

These instruction sets comprise of families of 2Q gate types parameterized by continuous qubit rotation angles. That is, it allows a large set of different physical operations to be realized on the qubits, based on the input angles.

However, having such a large number of gate types is problematic because each gate type has to be calibrated periodically, across the full system, to obtain high fidelity implementations. This results in substantial recurring calibration overheads even on current systems which use only a few gate types.

Our work aims to navigate this tradeoff between application expressivity and calibration overhead, and identify what instructions vendors should implement to get the best expressivity with acceptable calibration time.

Studying this tradeoff is challenging because of the diversity in QC application requirements, the need to optimize applications for widely different hardware gate types and noise variations across gate types. Therefore, our work develops NuOp, a flexible compilation pass based on numerical optimization, to efficiently decompose application operations into arbitrary hardware gate types. Using NuOp and four important quantum applications,

we study the instruction set proposals of Rigetti and Google, with realistic noise simulations and a calibration model. Our experiments show that implementing 4-8 types of 2Q gates is sufficient to attain nearly the same expressivity as a full continuous gate family, while reducing the calibration overhead by two orders of magnitude. With several vendors proposing rich gate families as means to higher fidelity, our work has potential to provide valuable instruction set design guidance for near-term QC systems.

12:45 - 01:00PM Albireo: Energy-Efficient Acceleration of Convolutional Neural Networks via Silicon Photonics

Kyle Shiflett, Avinash Karanth (Ohio University); Razvan Bunescu (UNC Charlotte); Ahmed Louri (GWU)

Abstract

With the end of Dennard scaling, highly-parallel and specialized hardware accelerators have been proposed to improve the throughput and energy-efficiency of deep neural network (DNN) models for various applications. However, collective data movement primitives such as multicast and broadcast that are required for multiply-and-accumulate (MAC) computation in DNN models are expensive, and require excessive energy and latency when implemented with electrical networks. This consequently limits the scalability and performance of electronic hardware accelerators. Emerging technology such as silicon photonics can inherently provide efficient implementation of multicast and broadcast operations, making photonics more amenable to exploit parallelism within DNN models. Moreover, when coupled with other unique features such as low energy consumption, high channel capacity with wavelength-division multiplexing (WDM), and high speed, silicon photonics could potentially provide a viable technology for scaling DNN acceleration.

In this paper, we propose Albireo, an analog photonic architecture for scaling DNN acceleration. By characterizing photonic devices such as microring resonators (MRRs) and Mach-Zehnder modulators (MZM) using photonic simulators, we develop realistic device models and outline their capability for system level acceleration. Using the device models, we develop an efficient broadcast combined with multicast data distribution by leveraging parameter sharing through unique WDM dot product processing. We evaluate the energy and throughput performance of Albireo on DNN models such as ResNet18, MobileNet and VGG16. When compared to current state-of-the-art electronic accelerators, Albireo increases throughput by 110 X, and improves energy-delay product (EDP) by an average of 74 X with current photonic devices. Furthermore, by considering moderate and aggressive photonic scaling, the proposed Albireo design shows that EDP can be reduced by at least 229 X.

12:00 -01.00PM Session 10B. Security II

12:00 PM – 01:00 PM (EDT/New York)

09:00 AM (PDT/San Francisco), 18:00 (CEST/Brussels), Tue 00:00 (CST/Beijing)

12:00 - 12:15PM IntroSpectre: A Pre-Silicon Framework for Discovery and Analysis of Transient Execution Vulnerabilities

Moein Ghaniyoun, Kristin Barber, Yinqian Zhang, Radu Teodorescu (Ohio State)

Abstract:

Transient execution vulnerabilities originate in the extensive speculation implemented in modern high-performance microprocessors. Identifying all possible vulnerabilities in complex designs is very challenging. One of the challenges stems from the lack of visibility into the transient micro-architectural state of the processor. Prior work has used covert channels to identify data leakage from transient state, which limits the systematic discovery of all potential leakage sources.

This paper presents IntroSpectre, a pre-silicon framework for early discovery of transient execution vulnerabilities. IntroSpectre addresses the lack of visibility into the micro-architectural processor state by integrating into the register transfer level (RTL) design flow, gaining full access to the internal state of the processor. Full visibility into the processor state enables IntroSpectre to perform a systematic leakage analysis that includes all micro-architectural structures, allowing it to identify potential leakage that may not be reachable with known side channels. We implement IntroSpectre on an RTL simulator and use it to perform transient leakage analysis on the RISC-V BOOM processor. We identify multiple transient leakage scenarios, most of which had not been highlighted on this processor design before.

12:15- 12:30PM Maya: Using Formal Control to Obfuscate Power Side Channels

Raghavendra Pradyumna Pothukuchi, Sweta Yamini Pothukuchi, Petros G Voulgaris, Alex Schwing, Josep Torrellas (UIUC)

Abstract:

The security of computers is at risk because of information leaking through their power consumption. Attackers can use advanced signal measurement and analysis to recover sensitive data from this side channel. To address this problem, this paper presents Maya, a simple and effective defense against power side channels. The idea is to use formal control to re-shape the power dissipated by a computer in an application-transparent manner—preventing attackers from learning any information about the applications that are running. With formal control, a controller can reliably keep power close to a desired target function even when runtime conditions change unpredictably. By selecting the target function intelligently, the controller can make power to follow any desired shape, appearing to carry activity information which, in reality, is unrelated to the application. Maya can be implemented in privileged software, firmware, or simple hardware. In this paper, we implement Maya on three machines using privileged threads only, and show its effectiveness and ease of deployment. Maya has already thwarted a newly-developed remote power attack.

12:30 - 12:45PM Demystifying the System Vulnerability Stack: Transient Fault Effects Across the Layers

George Papadimitriou, Dimitris Gizopoulos (University of Athens)

Abstract

In this paper, we revisit the system vulnerability stack for transient faults. We reveal severe pitfalls in widely used vulnerability measurement approaches, which separate the hardware and the software layers. We rely on microarchitecture level fault injection to derive very tight full-system vulnerability measurements. For our architectural and microarchitectural measurements, we employ GeFIN, a state-of-the-art fault injector built on top of the gem5 simulator, while for software level measurements we employ the LLFI fault injector. Analyzing two different Arm ISAs and two different microarchitectures for each ISA, we quantify the sources and the magnitude of error of architecture and software level vulnerability evaluation methods, which aim to reproduce the effects of hardware faults. We show that widely applied methodologies for system resilience evaluation fail to capture important fault manifestation and propagation aspects and lead to misleading findings, which report opposite vulnerability results than a comprehensive cross-layer analysis. To justify the validity of our findings we employ a state-of-the-art software-based fault tolerance technique and evaluate its impact at all layers through a case study. Our evaluation shows that although higher-level methods can report significant vulnerability improvements (up to 3.8x vulnerability reduction), the actual cross-layer vulnerability of the protected system can be degraded (increased) by up to 30% for the selected benchmarks. Our analysis firmly suggests that only accurate methodologies for full-system vulnerability evaluation of a microprocessor can guide informed transient faults protection decisions either at the hardware or at the software layer.

12:45 - 01:00PM No-FAT: Architectural Support for Low Overhead Memory Safety Checks

Mohamed Tarek Ibn Ziad, Miguel Arroyo, Evgeny Manzhosov, Ryan Piersma (Columbia); Simha Sethumadhavan (Columbia / Chip Scan)

Abstract:

Memory safety continues to be a significant software reliability and security problem, and low overhead and low complexity hardware solutions have eluded computer designers.

In this paper, we explore a pathway to deployable memory safety defenses. Our technique builds on a recent trend in software: the usage of binning memory allocators. We observe that if memory allocation sizes (e.g., malloc sizes) are made an architectural feature, then it is possible to overcome many of the thorny issues with traditional approaches to memory safety such as compatibility with unsecured software and significant performance degradation. We show that our architecture, No-FAT, incurs an overhead of 8% on SPEC CPU2017 benchmarks, and our VLSI measurements show low power and area overheads. Finally, as No-FAT's hardware is aware of the memory allocation sizes, it effectively mitigates certain speculative attacks (e.g., Spectre-V1) with no additional cost.

When our solution is used for pre-deployment fuzz testing it can improve fuzz testing bandwidth by an order of magnitude compared to state-of-the-art approaches.

01:00 - 01:45PM Session 11A. DRAM / IO / Network

01:00 PM – 01:45 PM (EDT/New York)

10:00 AM (PDT/San Francisco), 19:00 (CEST/Brussels), Tue 01:00 (CST/Beijing)

01:00 - 01:15PM Ghost Routing to Enable Oblivious Computation on Memory-Centric Networks

Yeonju Ro, Seongwook Jin, Jaehyuk Huh, John Kim (KAIST)

01:15 - 01:30PM QUAC-TRNG: High-Throughput True Random Number Generation Using Quadruple Row Activation in Commodity DRAM Chips

Ataberk Olgun (TOBB); Abdullah Giray Yaglikci, Minesh Patel, Jeremie Kim (ETH Zurich); Fatma Nisa Bostanci (TOBB); Haocong Luo (ETH Zurich); Nandita Vijaykumar (Toronto); Oguz Ergin (TOBB); Onur Mutlu (ETH Zurich)

Abstract

True random number generators (TRNG) sample random physical processes to create large amounts of random numbers for various use cases, including security-critical cryptographic primitives, scientific simulations, machine learning applications, and even recreational entertainment. Unfortunately, not every computing system is equipped with dedicated TRNG hardware, limiting the application space and security guarantees for such systems. To open the application space and enable security guarantees for the overwhelming majority of computing systems that do not necessarily have dedicated TRNG hardware (e.g., processing-in-memory systems), we develop QUAC-TRNG, a new high-throughput TRNG that can be fully implemented in commodity DRAM chips, which are key components in most modern systems.

QUAC-TRNG exploits the new observation that a carefully-engineered sequence of DRAM commands activates four consecutive DRAM rows in rapid succession. This QUadruple ACtivation (QUAC) causes the bitline sense amplifiers to non-deterministically converge to random values when we activate four rows that store conflicting data because the net deviation in hotline voltage fails to meet reliable sensing margins.

We experimentally demonstrate that QUAC reliably generates random values across 136 commodity DDR4 DRAM chips from one major DRAM manufacturer. We describe how to develop an effective TRNG (QUAC-TRNG) based on QUAC. We evaluate the quality of our TRNG using the commonly-used NIST statistical test suite for randomness andfind that QUAC-TRNG successfully passes each test. Our experimental evaluations show that QUAC-TRNG reliably generates true random numbers with a throughput of 3.44 Gb/s (per DRAM channel), outperforming the state-of-the-art DRAM-based TRNG by 15.08X and 1.41X for basic and throughput-optimized versions, respectively. We show that QUAC-TRNG utilizes DRAM bandwidth better than the state-of-the-art, achieving up to 2.03X the throughput of a throughput-optimized baseline when scaling bus frequencies to 12 GT/s.

01:30 - 01:45PM A RISC-V In-Network Accelerator for Flexible High-Performance Low-Power Packet Processing

Salvatore Di Girolamo, Andreas Kurth, Alexandru Calotoiu, Thomas Benz, Timo Schneider (ETH Zurich); Jakub Beránek (Technical University of Ostrava); Luca Benini, Torsten Hoefler (ETH Zurich)

Abstract:

The capacity of offloading data and control tasks to the network is becoming increasingly important, especially if we consider the faster growth of network speed when compared to CPU frequencies. In-network compute alleviates the host CPU load by running tasks directly in the network, enabling additional computation/communication overlap and potentially improving overall application performance. However, sustaining bandwidths provided by next-generation networks, e.g., 400 Gbit/s, can become a challenge. sPIN is a programming model for in-NIC compute, where users specify handler functions that are executed on the NIC, for each incoming packet belonging to a given message or flow. It enables a CUDA-like acceleration, where the NIC is equipped with lightweight processing elements that process network packets in parallel. We investigate the architectural specialties that a sPIN NIC should provide to enable high-performance, low-power, and flexible packet processing. We introduce PsPIN, a first open-source sPIN implementation, based on a multi-cluster RISC-V architecture and designed according to the identified architectural specialties. We investigate the performance of PsPIN with cycle-accurate simulations, showing that it can process packets at 400 Gbit/s for several use cases, introducing minimal latencies (26 ns for 64 B packets) and occupying a total area of 18.5 mm2 (22 nm FDSOI).

01:00 - 01:45PM Session 11B. Security III

01:00 PM – 01:45 PM (EDT/New York)

10:00 AM (PDT/San Francisco), 19:00 (CEST/Brussels), Tue 1:00 (CST/Beijing)

01:00 - 01:15PM Leaky Buddies: Cross-Component Covert Channels on Integrated CPU-GPU Systems

Sankha Baran Dutta (UC Riverside); Hoda Naghibijouybari (Binghamton); Nael Abu-Ghazaleh (UC Riverside); Andres Marquez, Kevin Barker (Pacific Northwest National Lab)

Abstract:

Graphics Processing Units (GPUs) are ubiquitous components used across the range of today's computing platforms, from phones and tablets, through personal computers, to high-end server class platforms. With the increasing importance of graphics and video workloads, recent processors are shipped with GPU devices that are integrated on the same chip. Integrated GPUs share some resources with the CPU and as a result, there is a potential for microarchitectural attacks from the GPU to the CPU or vice versa. We consider the potential for covert channel attacks that arise either from shared microarchitectural components (such as caches) or through shared contention domains (e.g., shared buses). We illustrate these two types of channels by developing two reliable covert channel attacks.

The first covert channel uses the shared LLC cache in Intel's integrated GPU architectures. The second is a contention based channel targeting the ring bus connecting the CPU and GPU to the LLC. This is the first demonstrated microarchitectural attack crossing the component boundary (GPU to CPU or vice versa). Cross component channels introduce a number of new challenges that we had to overcome since they occur across heterogeneous components that use different computation models and are interconnected using asymmetric memory hierarchies.

We also exploit GPU parallelism to increase the bandwidth of the communication, even without relying on a common clock. The LLC based channel achieves a bandwidth of 120 kbps with a low error rate of 2%, while the contention based channel delivers up to 400 kbps with a 0.8% error rate. We also demonstrate a proofof-concept prime-and-probe side channel attack that probes the full LLC from the GPU.

01:15 - 01:30PM IChannels: Exploiting Current Management Mechanisms to Create Covert Channels in Modern Processors

Jawad Haj-Yahya, Jeremie Kim, Ivan Puddu, Abdullah Giray Yaglikci, Mohammed Alser, Lois Orosa, Juan Gómez Luna, Onur Mutlu (ETH Zurich)

01:30 - 01:45PM ZeRØ: Zero-Overhead Resilient Operation Under Pointer Integrity Attacks

Mohamed Tarek Ibn Ziad, Miguel Arroyo, Evgeny Manzhosov (Columbia); Simha Sethumadhavan (Columbia / Chip Scan)

Abstract:

A large class of today's systems require high levels of availability and security. Unfortunately, state-of-the-art security solutions tend to induce crashes and raise exceptions when under attack, trading off availability for security. In this work, we propose ZeRØ, a pointer integrity mechanism that can continue program execution even when under attack. ZeRØ proposes unique memory instructions and a novel metadata encoding scheme to protect code and data pointers. The combination of instructions and metadata allows ZeRØ to avoid explicitly tagging every word in memory, eliminating performance overheads. Moreover, ZeRØ is a deterministic security primitive that requires minor microarchitectural changes. We show that ZeRØ is better than commercially available state-of-the-art hardware primitives, e.g., ARM's Pointer Authentication (PAC), by a significant margin. ZeRØ incurs zero performance overheads on the SPEC CPU2017 benchmarks, and our VLSI measurements show low power and area overheads.

01:45 - 02:30PM Panel 5: Microprocessor at 50: Societal Challenges

01:45 PM – 02:30 PM (EDT/New York)

10:45 AM (PDT/San Francisco), 19:45 (CEST/Brussels), Tue 01:45 (CST/Beijing)

The 50 years of microprocessor technologies have tremendously advanced all aspects of our lives. This panel provides us the space to examine the powerful technologies we are developing, responsibilities and societal impacts we must keep in mind when developing the technologies. This panel will discuss the societal challenges brought by digital technologies — the ever-increasing carbon emissions from computing, bias and fairness issues facing AI technologies, and the disparate social justice. What underinvested research directions should the community focus on in order to build environmentally-sustainable, socially-responsible technologies for the next decades to come?

08:00 - 09:00PM Session 12A. Accelerators III

08:00 PM – 09:00 PM (EDT/New York)

05:00 PM (PDT/San Francisco), Tue 02:00 (CEST/Brussels), Tue 08:00 (CST/Beijing)

08:00 - 08:15PM NN-Baton: DNN Workload Orchestration and Chiplet Granularity Exploration for Multichip Accelerators

Zhanhong Tan, Hongyu Cai (Tsinghua); Runpei Dong (Xi'an Jiaotong University); Kaisheng Ma (Tsinghua)

Abstract:

The revolution of machine learning poses an unprecedented demand for computation resources, urging more transistors on a single monolithic chip, which is not sustainable in the Post-Moore era. The multichip integration with small functional dies, called chiplets, can reduce the manufacturing cost, improve the fabrication yield, and achieve die-level reuse for different system scales. DNN workload mapping and hardware design space exploration on such multichip systems are critical, but missing in the current stage.

This work provides a hierarchical and analytical framework to describe the DNN mapping on a multichip accelerator and analyze the communication overhead. Based on this framework,

we propose an automatic tool called NN-Baton with a pre-design flow and a post-design flow. The pre-design flow aims to guide the chiplet granularity exploration with given area and performance budgets for the target workload. The post-design flow focuses on the workload orchestration on different computation levels - package, chiplet, and core - in the hierarchy. Compared to Simba, NN-Baton generates mapping strategies that save 22.5% 44% energy under the same computation and memory configurations. The architecture exploration demonstrates that area is a decisive factor for the chiplet granularity. For a 2048-MAC system under a 2 mm² chiplet area constraint, the 4-chiplet implementation with 4 cores and 16 lanes of 8-size vector-MAC

is always the top-pick computation allocation across several benchmarks. In contrast, the optimal memory allocation policy in the hierarchy typically depends on the neural network models..

08:15 - 08:30PM SNAFU: An Ultra-Low-Power, Energy-Minimal CGRA-Generation Framework and Architecture

Graham Gobieski, Oguz Atli, Ken Mai, Brandon Lucia, Nathan Beckmann (CMU)

Abstract

Ultra-low-power (ULP) devices are becoming pervasive, enabling many emerging sensing applications. Energy-efficiency is paramount in these applications, as efficiency determines device lifetime in battery-powered deployments and performance in energy-harvesting deployments. Unfortunately, existing designs fall short because ASICs' upfront costs are too high and prior ULP architectures are too inefficient or inflexible.

We present SNAFU, the first framework to flexibly generate ULP coarse-grain reconfigurable arrays (CGRAs). SNAFU provides a standard interface for processing elements (PE), making it easy to integrate new types of PEs for new applications. Unlike prior high-performance, high-power CGRAs, SNAFU is designed from the ground up to minimize energy consumption while maximizing flexibility. SNAFU saves energy by configuring PEs and routers for a single operation to minimize switching activity; by minimizing buffering within the fabric; by implementing a statically routed, bufferless, multi-hop network; and by executing operations in-order to avoid expensive tag-token matching.

We further present SNAFU-ARCH, a complete ULP system that integrates an instantiation of the SNAFU fabric alongside a scalar RISC-V core and memory. We implement SNAFU in RTL and evaluate it on an industrial sub-28 nm FinFET process across a suite of common sensing benchmarks. SNAFU-ARCH operates at <1mW, orders-of-magnitude less power than most prior CGRAs. SNAFU-ARCH uses 41% less energy and runs 4.4× faster than the prior state-of-the-art general-purpose ULP architecture. Moreover, we conduct three comprehensive case-studies to quantify the cost of programmability in SNAFU. We find that SNAFU-ARCH is close to ASIC designs built in the same technology, using just 2.6× more energy on average.

08:30 - 08:45PM SARA: Scaling a Reconfigurable Dataflow Accelerator

Yaqi Zhang, Nathan Zhang, Tian Zhao, Matt Vilim, Muhammad Shahbaz, Kunle Olukotun (Stanford)

Abstract:

The need for speed in modern data-intensive workloads and the rise of "dark silicon" in the semiconductor industry are pushing for larger, faster, and more energy and area-efficient architectures, such as Reconfigurable Dataflow Accelerators (RDAs). Nevertheless, challenges remain in developing mechanisms to effectively utilize the compute power of these large-scale RDAs.

To address these challenges, we present SARA, a compiler that employs a novel mapping strategy to efficiently utilize large-scale RDAs. Starting from a single-threaded imperative abstraction, SARA spatially maps a program onto RDA's distributed resources, exploiting dataflow parallelism within and across hyperblocks to saturate the compute throughput of an RDA. SARA introduces (a) compiler-managed memory consistency (CMMC), a control paradigm that hierarchically pipelines a nested and data-dependent control-flow graph onto a dataflow architecture, and (b) a compilation flow that decomposes the program graph across distributed heterogeneous resources to hide low-level RDA constraints from programmers. Our evaluation shows that SARA achieves close to perfect performance scaling on a recently proposed RDA---Plasticine. Over a mix of deep-learning, graph-processing, and streaming applications, SARA achieves a 1.9x geo-mean speedup over a Tesla V100 GPU using only 12% of the silicon area.

08:45 - 09:00PM HASCO: Towards Agile HArdware and Software CO-design for Tensor Computation

Qingcheng Xiao, Size Zheng, Bingzhe Wu (Center for Energy-Efficient Computing and Applications, Peking); Pengcheng Xu (Peking); Xuehai Qian (USC); Yun Liang (Center for Energy-Efficient Computing and Applications, Peking)

Abstract:

Tensor computations overwhelm traditional general-purpose computing devices due to the large amounts of data and operations of the computations. They call for a holistic solution composed of both hardware acceleration and software mapping. Hardware/software (HW/SW) co-design optimizes the hardware and software in concert and produces high-quality solutions. There are two main challenges in the co-design flow. First, multiple methods exist to partition tensor computation and have different impacts on performance and energy efficiency. Besides, the hardware part

must be implemented by the intrinsic functions of spatial accelerators. It is hard for programmers to identify and analyze the partitioning methods manually. Second, the overall design space composed of HW/SW partitioning, hardware optimization, and software optimization is huge. The design space needs to be efficiently explored.

To this end, we propose an agile co-design approach HASCO that provides an efficient HW/SW solution to dense tensor computation. We use tensor syntax trees as the unified IR, based on which we develop a two-step approach to identify partitioning methods. For each method, HASCO explores the hardware and software design spaces. We propose different algorithms for the explorations, as they have distinct objectives and evaluation costs. Concretely, we develop a multi-objective Bayesian optimization algorithm to explore hardware optimization. For software optimization, we use heuristic and Q-learning algorithms. Experiments demonstrate that HASCO achieves a 1.25X to 1.44X latency reduction through HW/SW co-design compared with developing the hardware and software separately.

08:00-09:00PM Session 12B. Sparse Processing

08:00 PM – 09:00 PM (EDT/New York)

05:00 PM (PDT/San Francisco), Tue 02:00 (CEST/Brussels), Tue 08:00 (CST/Beijing)

08:00 - 08:15PM SpZip: Architectural Support for Effective Data Compression In Irregular Applications

Yifan Yang (MIT); Joel Emer (MIT / NVIDIA); Daniel Sanchez (MIT)

Abstract:

Irregular applications, such as graph analytics and sparse linear algebra, exhibit frequent indirect, data-dependent accesses to single or short sequences of elements that cause high main memory traffic and limit performance. Data compression is a promising way to accelerate irregular applications by reducing memory traffic. However, software compression adds substantial overheads, and prior hardware compression techniques work poorly on the complex access patterns of irregular applications.

We present SpZip, an architectural approach that makes data compression practical for irregular algorithms. SpZip accelerates the traversal, decompression, and compression of the data structures used by irregular applications. In addition, these activities run in a decoupled fashion, hiding both memory access and decompression latencies. To support the wide range of access patterns in these applications, SpZip is programmable, and uses a novel Dataflow Configuration Language to specify programs that traverse and generate compressed data. Our SpZip implementation leverages dataflow execution and time-multiplexing to implement programmability cheaply. We evaluate SpZip on a simulated multicore system running a broad set of graph and linear algebra algorithms. SpZip outperforms prior state-of-the art software-only (hardware-accelerated) systems by gmean 3.0x (1.5x) and reduces memory traffic by 1.7x (1.4x). These benefits stem from both reducing data movement due to compression, and offloading expensive traversal and (de)compression operations.

08:15 - 08:30PM Dual-Side Sparse Tensor Core

Yang Wang (UESTC / Microsoft Research); Chen Zhang (Microsoft Research); Zhiqiang Xie (ShanghaiTech University); Cong Guo (Shanghai Jiao Tong); Yunxin Liu (Microsoft Research); Jingwen Leng (Shanghai Jiao Tong)

Abstract:

In the era of artificial intelligence, convolutional neural networks (CNNs) are emerging as a powerful technique for computational imaging. They have shown superior quality for reconstructing fine textures from badly-distorted images and have potential to bring next-generation cameras and displays to our daily life. However, CNNs demand intensive computing power for generating high-resolution videos and defy conventional sparsity techniques when rendering dense details. Therefore, finding new possibilities in regular sparsity is crucial to enable large-scale deployment of CNN-based computational imaging. In this paper, we consider a fundamental but yet well-explored approach--algebraic sparsity--for energy-efficient CNN acceleration. We propose to build CNN models based on ring algebra that defines multiplication, addition, and non-linearity for n-tuples properly. Then the essential sparsity will immediately follow, e.g. n-times reduction for the number of real-valued weights. We define and unify several variants of ring algebras into a modeling framework, RingCNN, and make comparisons in terms of image quality and hardware complexity. On top of that, we further devise a novel ring algebra which minimizes complexity with component-wise product and achieves the best quality using directional ReLU. Finally, we design an accelerator, eRingCNN, to accommodate to the

proposed ring algebra, in particular with regular ring-convolution arrays for efficient inference and on-the-fly directional ReLU blocks for fixed-point computation. We implement two configurations, n=2 and 4 (50% and 75% sparsity), with 40 nm technology to support advanced denoising and super-resolution at up to 4K UHD 30 fps. Layout results show that they can deliver equivalent 41 TOPS using 3.76 W and 2.22 W, respectively. Compared to the real-valued counterpart, our ring convolution engines for n=2 achieve 2.00x energy efficiency and 2.08x area efficiency with similar or even better image quality. With n=4, the efficiency gains of energy and area are further increased to 3.84x and 3.77x with only 0.11 dB drop of peak signal-to-noise ratio (PSNR). The results show that RingCNN exhibits great architectural advantages for providing near-maximum hardware efficiencies and graceful quality degradation simultaneously.

08:30 - 08:45PM RingCNN: Exploiting Algebraically-Sparse Ring Tensors for Energy-Efficient CNN-Based Computational Imaging

Chao-Tsung Huang (National Tsing Hua University)

Abstract:

In the era of artificial intelligence, convolutional neural networks (CNNs) are emerging as a powerful technique for computational imaging. They have shown superior quality for reconstructing fine textures from badly-distorted images and have potential to bring next-generation cameras and displays to our daily life. However, CNNs demand intensive computing power for generating high-resolution videos and defy conventional sparsity techniques when rendering dense details. Therefore, finding new possibilities in regular sparsity is crucial to enable large-scale deployment of CNN-based computational imaging. In this paper, we consider a fundamental but yet well-explored approach--algebraic sparsity--for energy-efficient CNN acceleration. We propose to build CNN models based on ring algebra that defines multiplication, addition, and non-linearity for n-tuples properly. Then the essential sparsity will immediately follow, e.g. n-times reduction for the number of real-valued weights. We define and unify several variants of ring algebras into a modeling framework, RingCNN, and make comparisons in terms of image quality and hardware complexity. On top of that, we further devise a novel ring algebra which minimizes complexity with component-wise product and achieves the best quality using directional ReLU. Finally, we design an accelerator, eRingCNN, to accommodate to the proposed ring algebra, in particular with regular ring-convolution arrays for efficient inference and on-the-fly directional ReLU blocks for fixed-point computation. We implement two configurations, n=2 and 4 (50% and 75% sparsity), with 40 nm technology to support advanced denoising and super-resolution at up to 4K UHD 30 fps. Layout results show that they can deliver equivalent 41 TOPS using 3.76 W and 2.22 W, respectively. Compared to the real-valued counterpart, our ring convolution engines for n=2 achieve 2.00x energy efficiency and 2.08x area efficiency with similar or even better image quality. With n=4, the efficiency gains of energy and area are further increased to 3.84x and 3.77x with only 0.11 dB drop of peak signal-to-noise ratio (PSNR). The results show that RingCNN exhibits great architectural advantages for providing near-maximum hardware efficiencies and graceful quality degradation simultaneously.

08:45 - 09:00PM GoSPA: An Energy-Efficient High-Performance Globally Optimized SParse Convolutional Neural Network Accelerator

Chunhua Deng, Yang Sui, Siyu Liao (Rutgers); Xuehai Qian (USC); Bo Yuan (Rutgers)

Abstract:

The co-existence of activation sparsity and model sparsity in convolutional neural network (CNN) models makes sparsity-aware CNN hardware designs very attractive. The existing sparse CNN accelerators utilize intersection operation to search and identify the key positions of the matched entries between two sparse vectors, and hence avoid unnecessary computations. However, these state-of-the-art designs still suffer from three major architecture-level drawbacks, including 1) hardware cost for the intersection operation is high; 2) frequent stalls of computation phase due to strong data dependency between intersection and computation phases; and 3) unnecessary data transfer incurred by the explicit intersection operation.

By leveraging the knowledge of the complete sparse 2-D convolution, this paper proposes two key ideas that overcome all of the three drawbacks. First, an implicit on-the-fly intersection is proposed to realize the optimal solution for intersection between one static stream and one dynamic stream, which is the case for sparse neural network inference. Second, by leveraging the global computation structure of 2-D convolution, we propose a specialized computation reordering to ensure that the activation is only transferred if necessary and only once.

Based on these two key ideas, we develop GoSPA, an energy-efficient high-performance Globally Optimized SParse CNN Accelerator. GoSPA is implemented with CMOS 28nm technology. Compared with the state-of-

the-art sparse CNN architecture, GoSPA achieves average 1.38X, 1.28X, 1.23X, 1.17X, 1.21X and 1.28X speedup on AlexNet, VGG, GoogLeNet, MobileNet, ResNet and ResNeXt workloads, respectively. Also, GoSPA achieves 5.38X, 4.96X, 4.79X, 5.02X, 4.86X and 2.06X energy efficiency improvement on AlexNet, VGG, GoogLeNet, MobileNet, ResNet and ResNeXt, respectively. In more comprehensive comparison including DRAM access, GoSPA also shows significant performance improvement over the existing designs.

09:00 - 10:30PM Panel 6: Research Funding Beyond Machine Learning and Quantum?

09:00 PM – 10:30 PM (EDT/New York)

06:00 PM (PDT/San Francisco), Tue 03:00 (CEST/Brussels), Tue 09:00 (CST/Beijing)

In the last decade, the computer architecture research community has grown dramatically. There are now many young faculty and graduate students in many academic institutions. With the seismic changes taking place in our technical field, we have an opportunity to use these creative minds to crack some of the many computer systems architecture challenges. To do so, researchers need to be aware of the opportunities available to them. This panel will include leaders from industry and funding agencies who will help the audience reflect on the existing funding and research opportunities, priorities and future trends.

Thursday, June 17th

09:00 - 03:00PM Workshop. CARRV (RISC-V)77

09:00AM – 03:00PM (EDT/New York)

09:00AM - 03:00PM Tutorial. AIBench

09:00 AM – 03:00PM (EDT/New York)

09:00AM - 04:00PM Workshop. WCAE

09:00AM – 03:00PM (EDT/New York)

09:00 - 12: 00AM Tutorial. AccelTraining

09:00AM – 12:00PM (EDT/New York)

Hardware Accelerators for Training Deep Neural Networks

10.00AM - 11.00AM Tutorial: SIGARCH CARES: Building Inclusive Research Environments

Cabanyal

10-11am EST

Please take this survey intended to gather your experiences of inclusion and exclusion, as well as ideas for build inclusion in our community. We are sending this survey to people attending ISCA 2021 and thus the focus will be on the Computer Architecture Community. (A similar survey is being sent to PLDI 2021 attendees.)

This survey is intended to generate reflection and discussion. We will open this discussion in the following two sessions (to be inclusive of people in time zones around the world) at ISCA 2021

12:00 - 03:00PM Tutorial. Voltage

12:00PM – 03:00PM (EDT/New York)

03.00PM - 10.00PM Workshop: CLEAR

Hemisferic

09:00AM – 04:00PM (EDT/New York)

04:00PM – 07:00PM Workshop. DRAMSec

04:00PM – 07:00PM (EDT/New York)

08.00PM - 09.00 PM Tutorial: SIGARCH CARES: Building Inclusive Research Environments

Cabanyal

08:00 – 09:00PM EST

Please take this survey intended to gather your experiences of inclusion and exclusion, as well as ideas for build inclusion in our community. We are sending this survey to people attending ISCA 2021 and thus the focus will be on the Computer Architecture Community. (A similar survey is being sent to PLDI 2021 attendees.)

This survey is intended to generate reflection and discussion. We will open this discussion in the following two sessions (to be inclusive of people in time zones around the world) at ISCA 2021

Friday, June 18th

09:00AM - 03:00PM Workshop. I2Q

09:00AM – 03:00PM (EDT/New York)

09:00AM - 03:00PM Tutorial. FireSim / Chipyard

09:00AM – 03:00PM (EDT/New York)

End-to-End Architecture Research with RISC-V SoC Generators, Agile Test Chips, and FPGA-Accelerated Simulation on Amazon EC2 F1

09:00AM - 03:00PM Workshop. uArch

09:00AM – 03:00PM (EDT/New York)

10:45 AM – 04:45 PM Workshop. SPSL

10:45 AM – 04:45 PM (EDT/New York)

09:00 -12:00AM Tutorial. DistDeep

09:00AM – 12:00PM (EDT/New York)

High Performance Distributed Deep Learning: A Beginner's Guide

09:00 -12:00AM Tutorial. ILLIXR

09:00 AM – 12:00PM (EDT/New York)

Illinois Extended Reality Testbed

12:00AM - 03:00PM Tutorial. MLPerf-Bench

12:00PM – 03:00PM (EDT/New York)

A Deep Dive into Deep Learning Benchmarking and Analysis

12:00 - 03:00PM Tutorial. MechaFlow

12:00PM – 03:00PM (EDT/New York)

Democratizing AI Hardware Landscape Evaluation and Exploration

Saturday, June 19th

09:00AM - 02:00PM Workshop. QRE

09:00AM – 02:00PM (EDT/New York)

09:00AM - 03:00PM Workshop. MLArchSys

09:00AM – 03:00PM (EDT/New York)

10:00AM - 02:00PM Workshop. AIDArc

9 AM – 3 PM (EDT/New York)

10:00AM - 02:00PM Tutorial. Xilinx HPC

9 AM – 3 PM (EDT/New York)

09:00 - 12:00AM Tutorial. SCALE-Sim

09:00AM – 12:00PM (EDT/New York)

Systolic CNN Accelerator Simulator

12:00AM - 15:00PM Tutorial. Sparse Tensor

12:00PM – 03:00 PM (EDT/New York)

Sparse Tensor Accelerators: Abstraction and Modeling