MCM-GPU: Multi-Chip-Module GPUs for Continued Performance Scalability

Seminar-AToMSC

Abdullah Amawi.

August 16, 2022

University of Göttingen.

Authors & Affiliation

- Authors: Akhil Arunkumar, Evgeny Bolotin, Benjamin Cho, Ugljesa Milic, Eiman Ebrahimi, Oreste Villa, Aamer Jaleel, Carole-Jean Wu, David Nellans.

- Institutions: Arizona State University, NVIDIA, University of Texas at Austin, Barcelona Supercomputing Center / Universitat Politecnica de Catalunya.

Table of contents

- Introduction
  - Why GPU?
  - GPU vs CPU
  - Rise of GPU Computing
- MCM-GPU
  - MCM-GPU MCM-GPU Idea & Alternatives
  - MCM-GPU Architecture
- Optimized-MCM-GPU
  - Optimized MCM-GPU Cache Architecture
  - Optimized MCM-GPU Scheduling
  - Optimized MCM-GPU First Touch
- Evaluation & results
- Related works, conclusion & Opinion
- Questions & discussion

Introduction

Why GPU?

- GPUs and parallel applications(scientific computing, data analytics, machine learning).

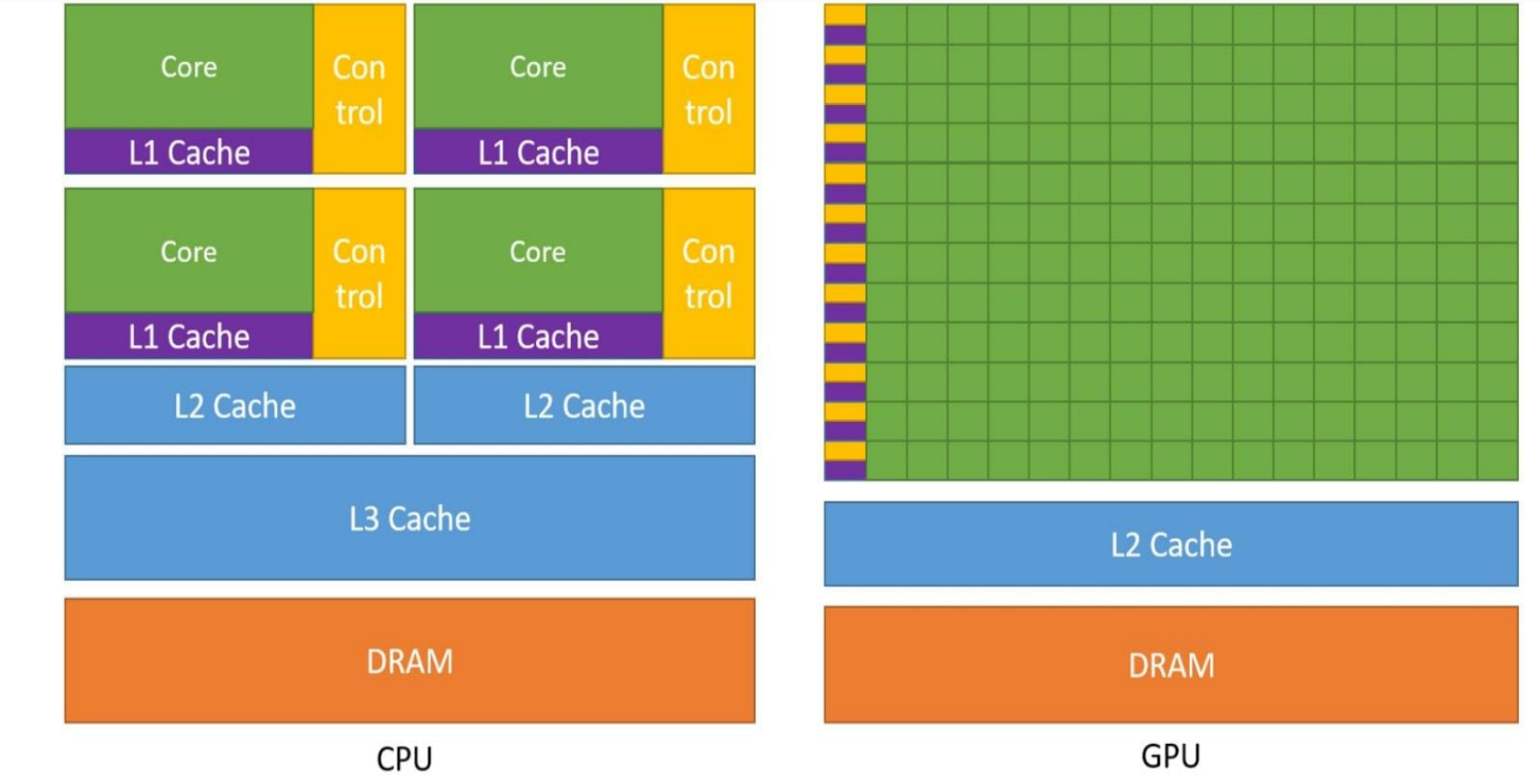

Figure 1: Why GPU?[2]

GPU vs CPU

- Comparing 32-Core AMD Threadripper to multiple NVIDIA GPUS.

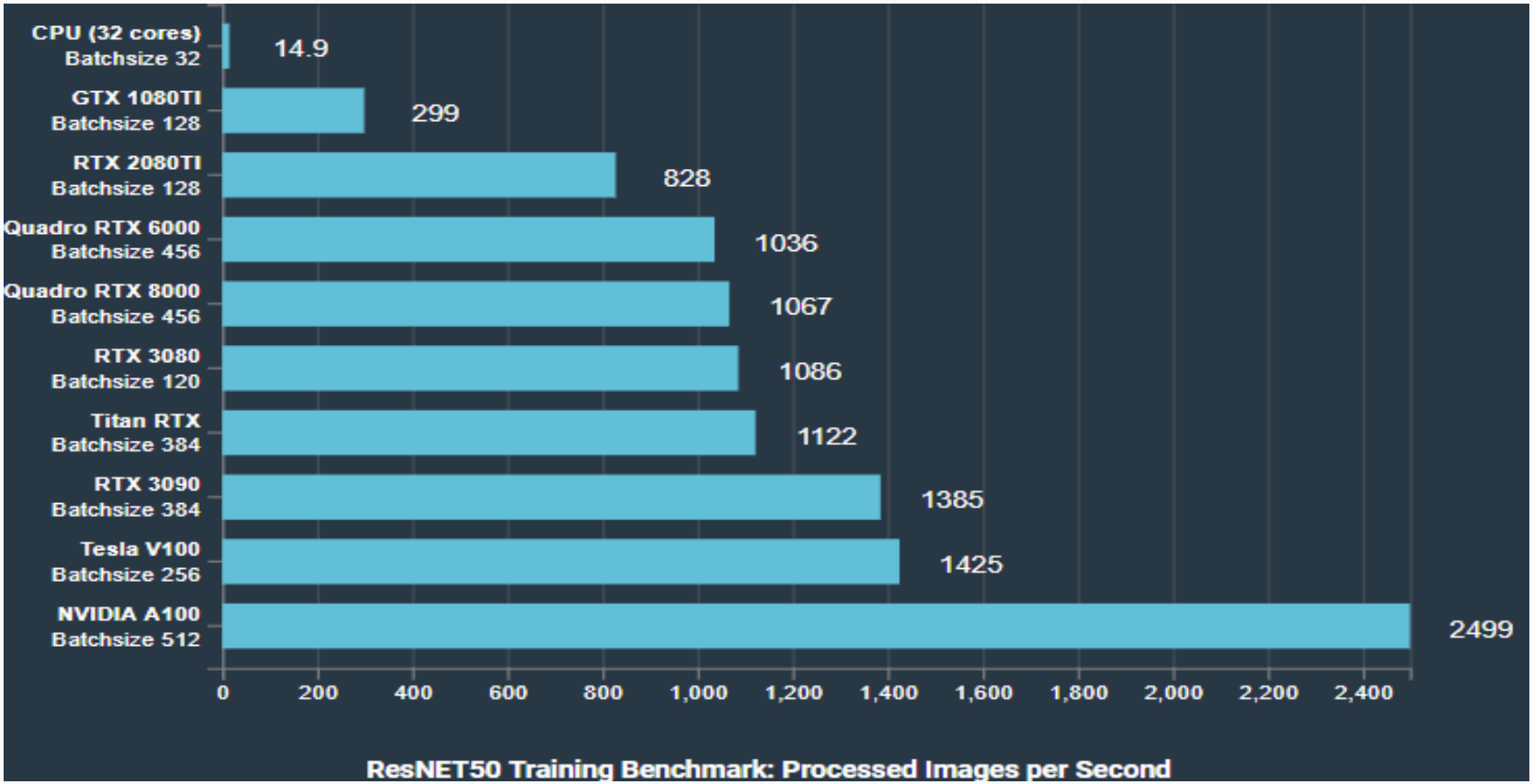

Figure 2: GPU Vs CPU [3]

Rise of GPU Computing

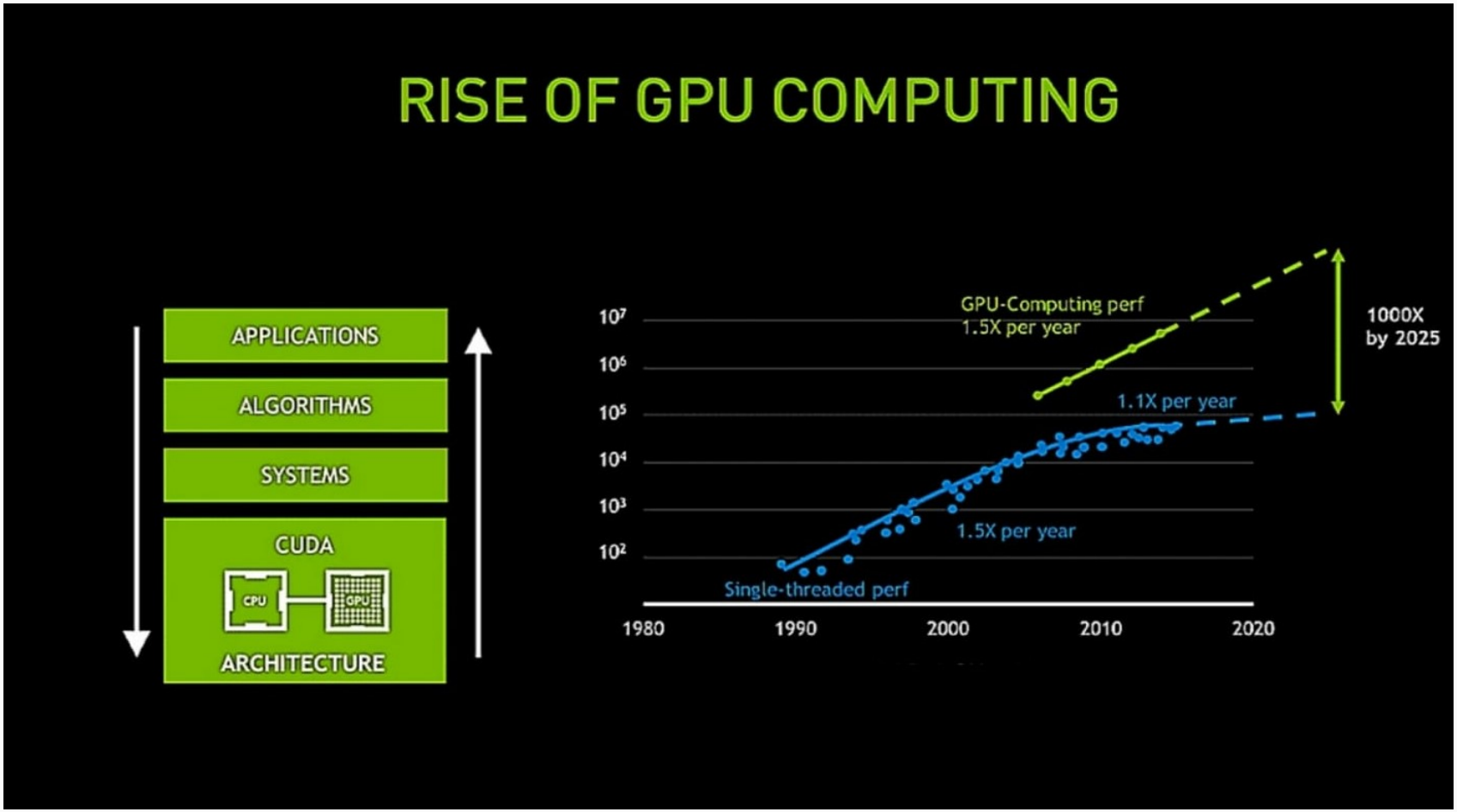

Figure 3: GPU computing and Moore law[1]

MCM-GPU.

MCM-GPU Idea & Alternatives

- Monolithic GPU and transistor scaling.
- Multi-GPU and drawbacks.
  - Partitioning.
  - Load balancing.
  - Synchronization.
- MCM-GPU and challenges.

MCM-GPU Architecture

- Eliminate hardware replications & enables resource sharing.
- Bigger, more capable GPUs & no additional programming effort.

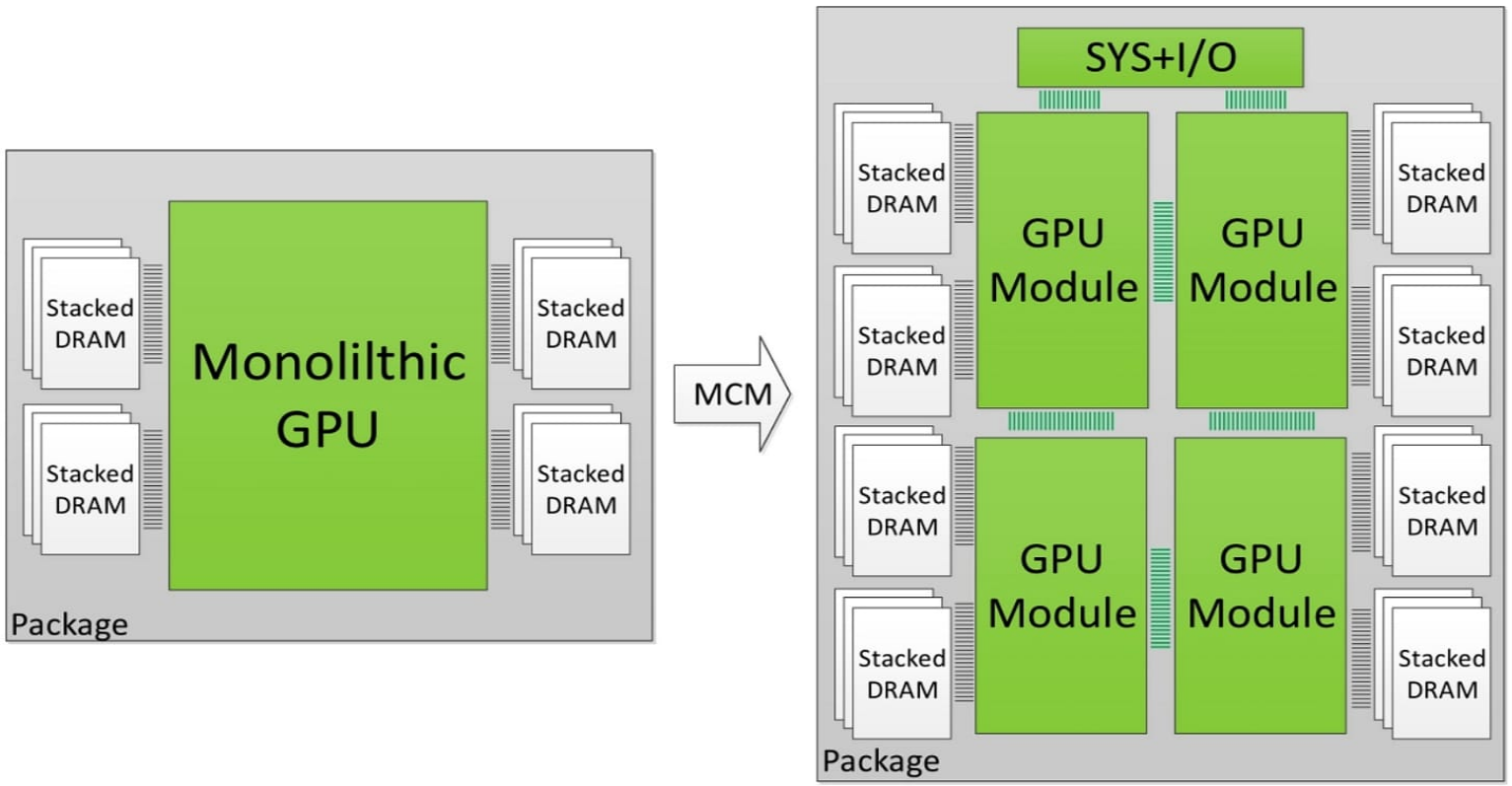

Figure 4: Monolithic GPU & MCM-GPU Architecture[5].

Optimized-MCM-GPU

Optimized-MCM-GPU Cache Architecture

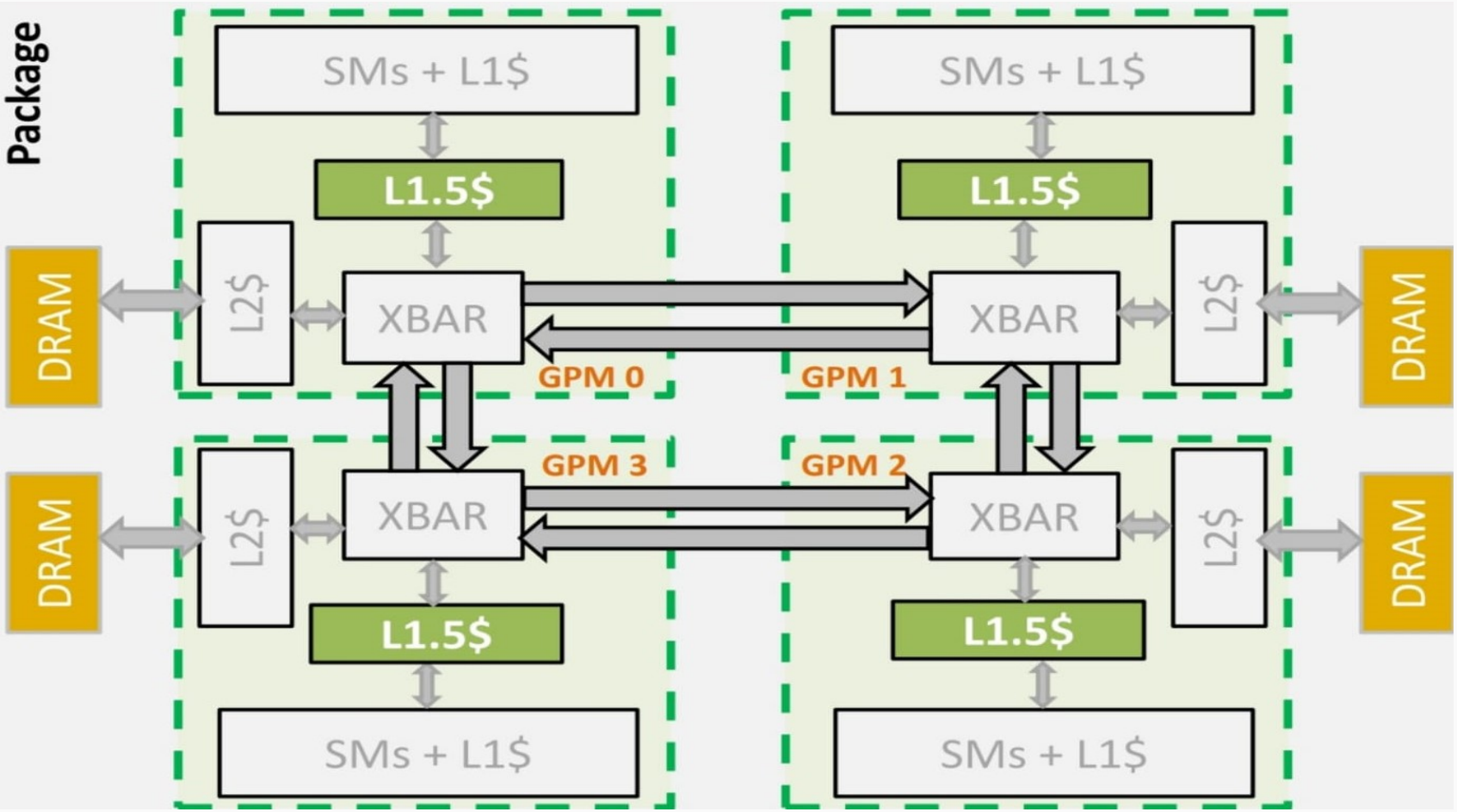

Figure 5: Optimized MCM-GPU Cache(first optimization)[5]

Optimized-MCM-GPU Distributed Scheduling

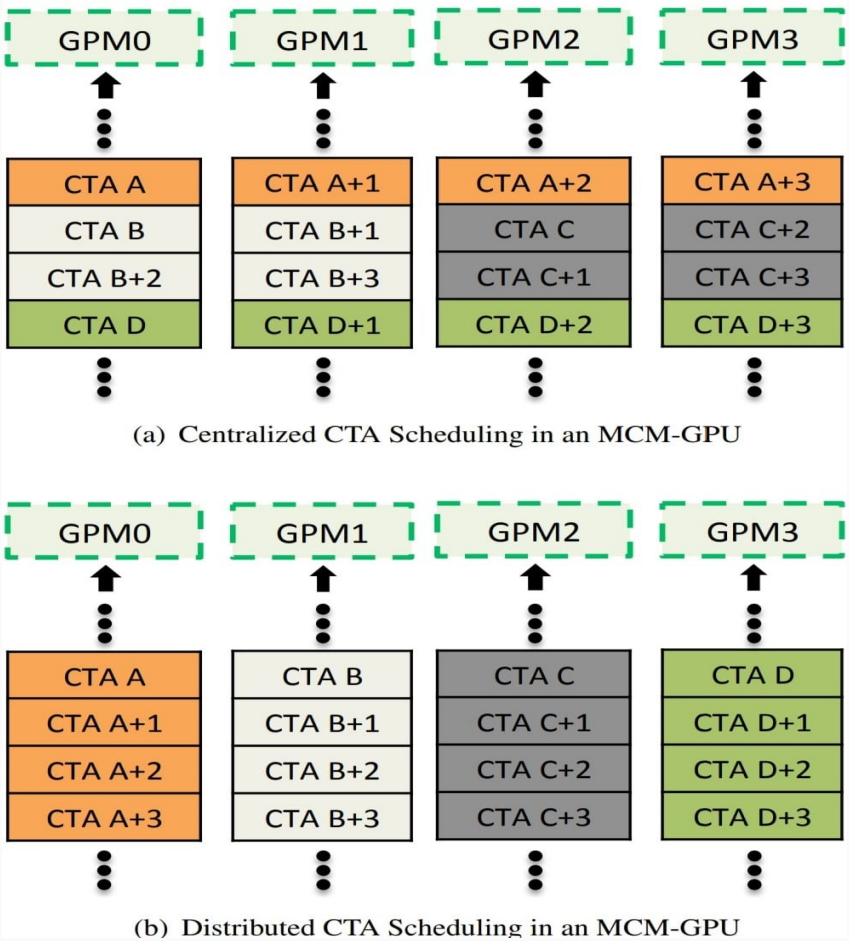

Figure 6: Optimized MCM-GPU Scheduler(2nd-optimization)[5]

Optimized-MCM-GPU Distributed Scheduling Performance

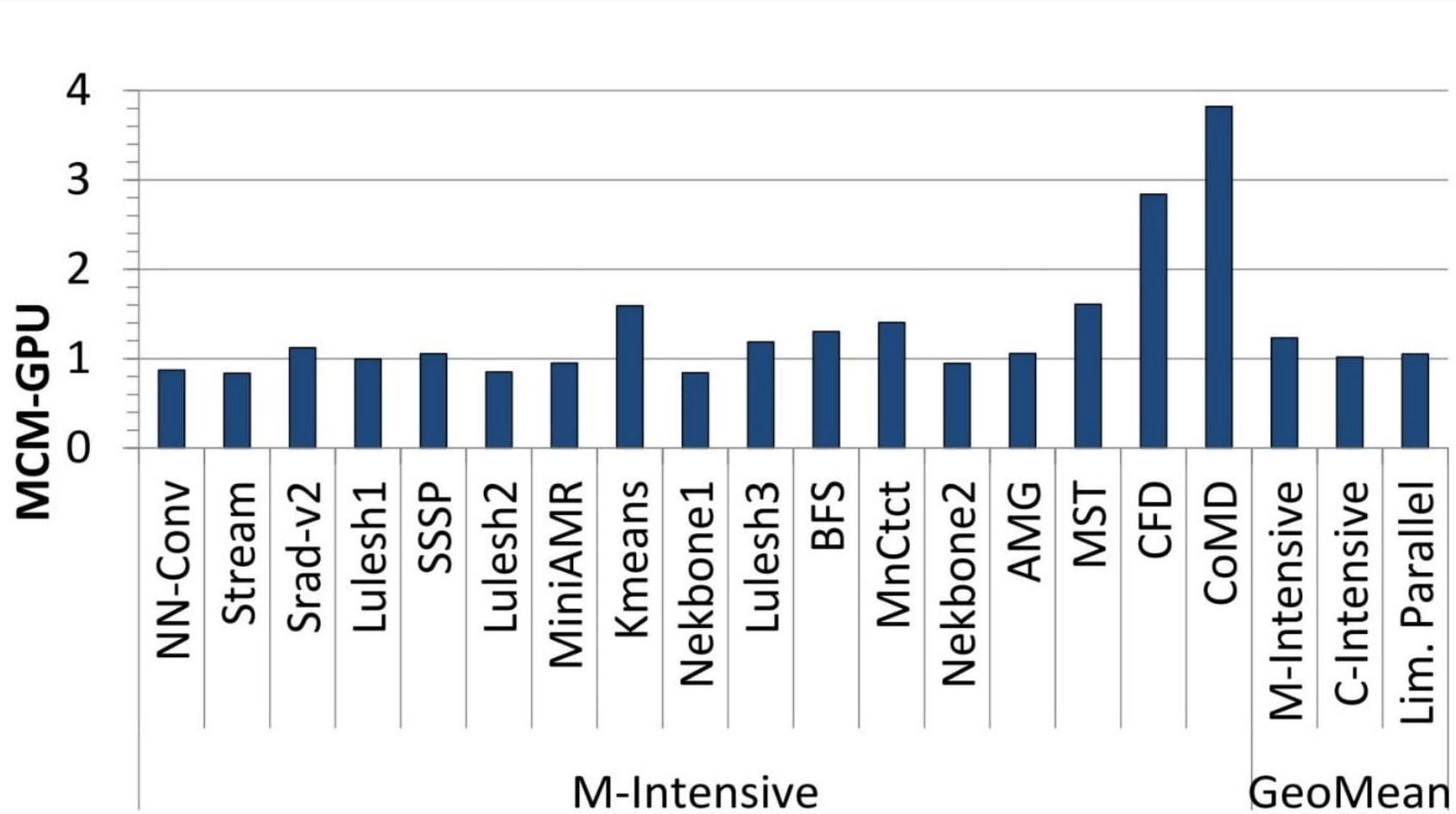

Figure 7: Optimized MCM-GPU Scheduler Performance[5]

Optimized-MCM-GPU First Touch

- Place Memory-Page in local Memory Partition of referenced GPM.
- Ex: Page 0 is accessed by CTA-X(on GPM0) > P0 on MP0.
- Maximises DRAM bandwidth utilization.

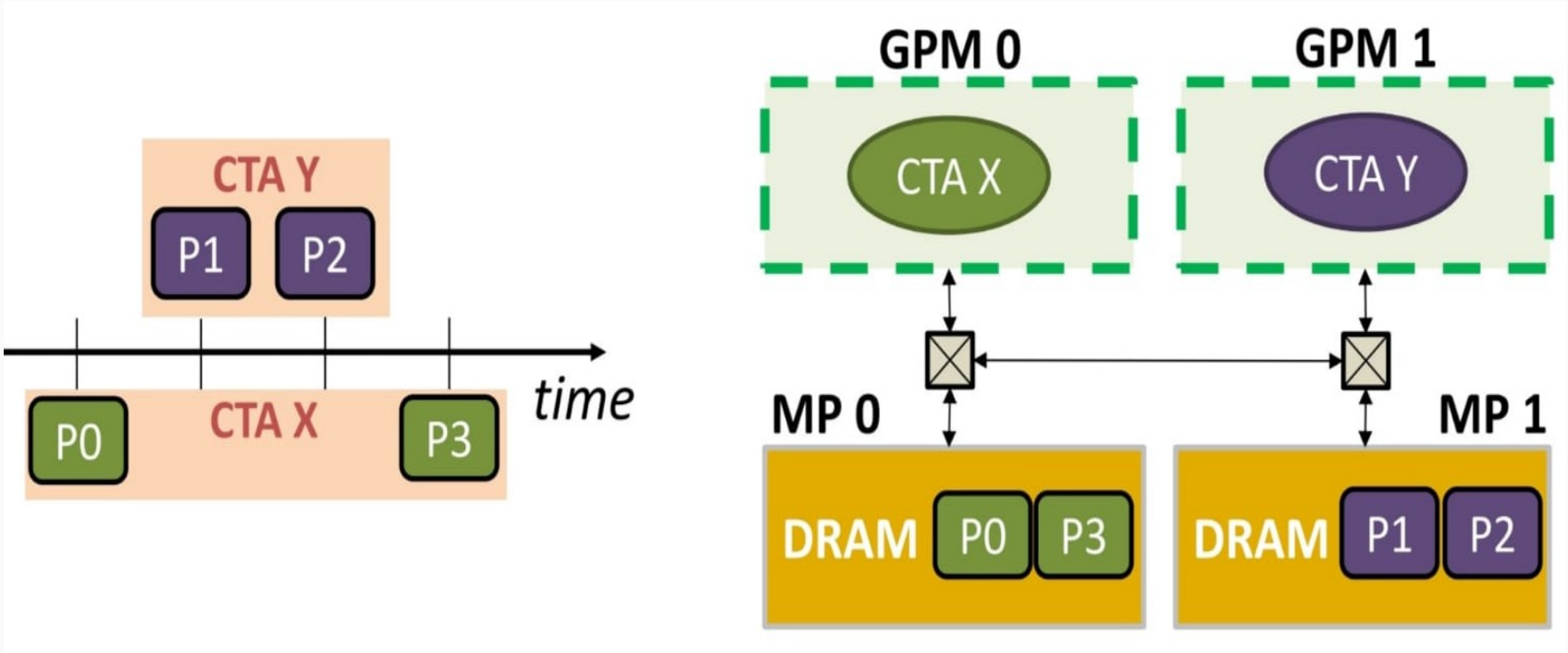

Figure 8: Optimized-MCM-GPU First-Touch Page Mapping(3rd-optimization)[5]

Optimized-MCM-GPU First Touch Results

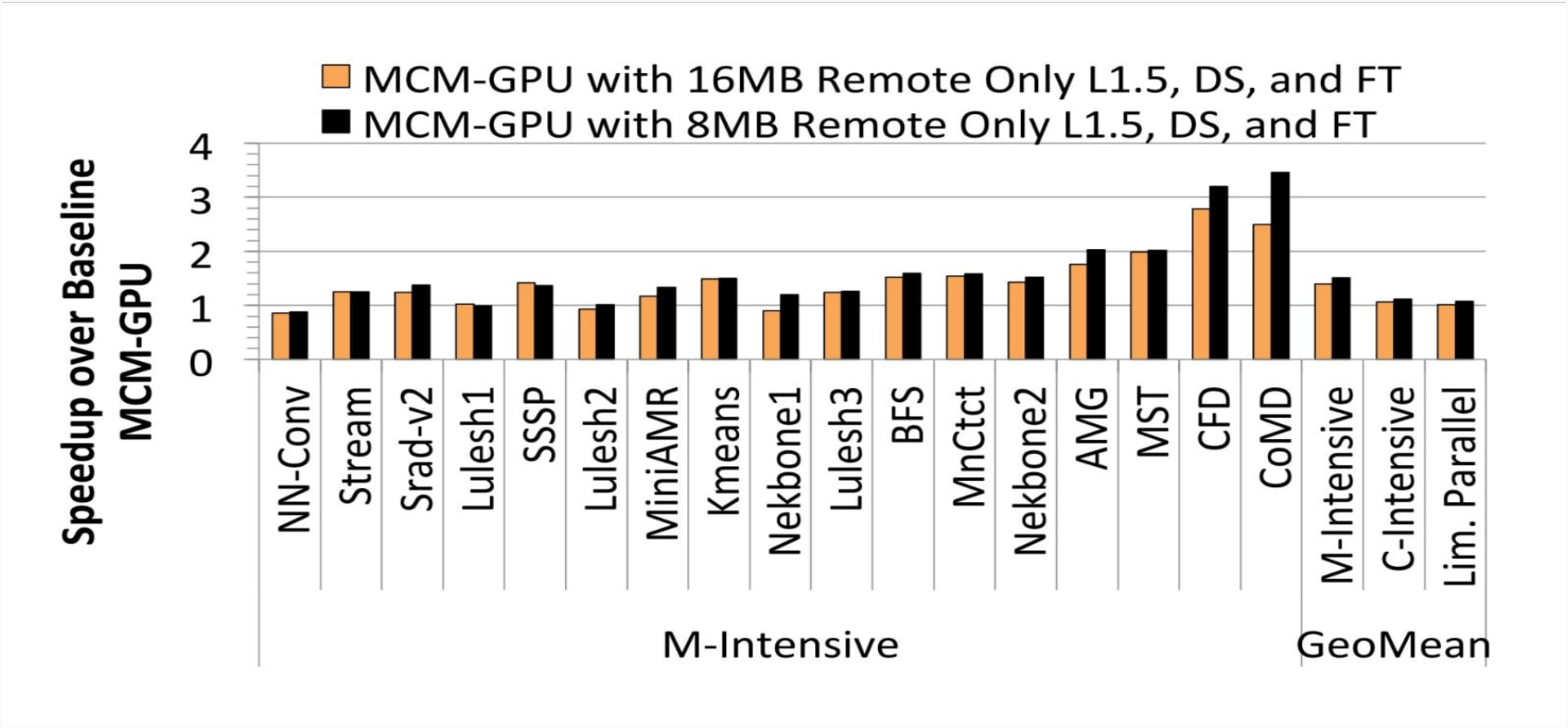

Figure 9: Optimized-MCM-GPU First Touch Results[5]

Evaluation & Results

Evaluation Methodology

- Use of an NVIDIA in-house simulator.
- Simulated GPU is similar to NVIDIA Pascal archeticture.
- SMs are modeled for parallelism.
- Evaluate High & Limited-parallelism (25=> or <= 25%).
- Evaluate Memory-Intensive and Compute-Intensive tasks.

Results.

- Baseline MCM-GPU with different optimizations results.

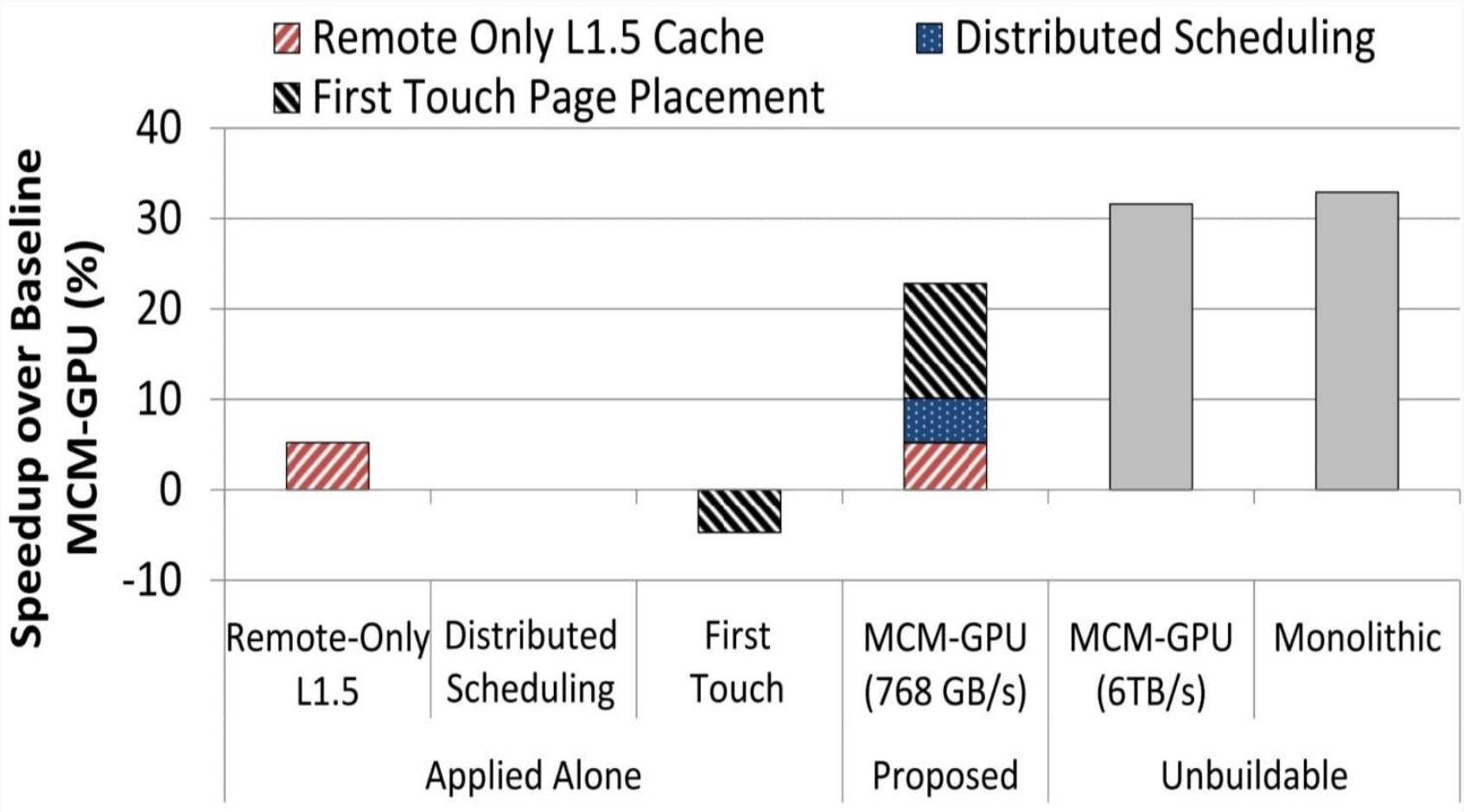

Figure 10: Optimized MCM-GPU results[5]

Related works & Conclusion

Related works.

Related Areas	Representative Article.	MCM Advantage.
MCM-Design.	Xenos: XBOX360 GPU[8]. The Xeon X5365[11]. IBM zEnterprise 196 Technical Guide[19]. AMD Server Solutions Playbook[4]. IBM Power Systems Deep Dive[10]. The Compute Architecture of Intel Processor Graphics Gen8[12].	Only applied to CPU. Combines CPU and GPU on chip
Multi-GPU-Systems.	Memory Access Patterns: The Missing Piece of the multi-GPU Puzzle[6]. Automatic Parallelization of Kernels in Shared-Memory Multi-GPU Nodes[7]. Achieving a Single Compute Device Image in OpenCL for Multiple GPUs[14]. Transparent CPU-GPU Collaboration for Data-parallel Kernels on Heterogeneous Systems[15].	Only work that is fully-suitable for MCM-GPUs. Only work that propose MCM-GPU-as a single logical GPU.
Signaling Tech	The 3rd generation of IBM's elastic interface on POWER6[9]. Enabling Interposer-based Disintegration of Multi-core Processors[13]. A scalable 0.128-to-1Tb/s 0.8-to-2.6pJ/b 64-lane parallel I/O in 32nm CMOS[17]. A 14-mW 6.25-Gb/s Transceiver in 90-nm CMOS[16]. Ground-Referenced Single-Ended Short-Reach Serial Link in 28 nm CMOS for Advanced Packaging Applications[18].	Operates at up to 3.2 Gbps Vs 20 Gbps Nvidia GRS(Ground-Referenced-Signaling)

Table 1: Related works comparison[5].

Conclusion.

- GPUs importance in compute-intensive fields such as AI.
- GPU growth importance and the need of MCM-GPUs to do so.
- The paper shows that MCM-GPUs are the future of GPU industry, but also demonstrates:
  - A 256 SMs MCM-GPU achieves 45.5% speedup over the largest possible monolithic GPU with 128 SMs.
  - It performs 26.8% better than an equally equipped discrete multi-GPU
  - Performance is within 10% of a monolithic GPU that cannot be built today.

Opinion about the Paper.

- On the positive side:
  - Novel idea that could be the GPU future.
  - Clear presentation of the idea and alternatives.
  - Great breakup of different GPU design alternations that they propose.
- On the negative side:
  - Totally ignores different additions by the competition.
  - Assumes the end of node technology prematurely.
  - Has a lot of biased false claims (Die size, SMs count, Cache).

Thank you!

Questions?

Additional resources

Terms

- CTA: Concurrent thread arrays.
- SMs: Stream multiprocessors.

References

References i

- https://blogs.nvidia.com/blog/2017/05/24/ai-revolution-eating-software/.
- https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html.
- https://www.aime.info/blog/deep-learning-gpu-benchmarks-2020/.
- AMD.
  Amd server solutions playbook.
  2012.

References ii


A. Arunkumar, E. Bolotin, B. Cho, UgljesaMilic, E. Ebrahimi, O. Villa, A. Jaleel, Carole-JeanWu, and D. Nellans.
Mcm-gpu: Multi-chip-module gpus for continued performance scalability.
2017.


T. Ben-Nun, E. Levy, A. Barak, and E. Rubin.
Memory access patterns: The missing piece of the multi-gpu puzzle.
2015.


J. Cabezas, L. Vilanova, I. Gelado, T. B. Jablin, N. Navarro, and W. mei W. Hwu.
Automatic parallelization of kernels in shared- memory multi-gpu nodes.
2015.

References iii


M. Doggett.
Xenos: Xbox360 gpu.
2005.


D. Dreps.
The 3rd generation of ibm's elastic interface on power6.
2007.


IBM.
Ibm power systems deep dive.
2012.


Intel.
The xeon x5365.
2007.

References iv


Intel.
The compute architecture of intel processor graphics gen8.
2015.


A. Kannan, N. E. Jerger, and G. H. Loh.
Enabling interposer-based disintegration of multi-core processors.
2015.


J. Kim, H. Kim, J. H. Lee, and J. Lee.
Achieving a single compute device image in opencl for multiple gpus.
2011.

References v

J. Lee, M. Samad, Y. Park, and S. Mahlke.
Transparent cpu-gpu collaboration for data-parallel kernels on heterogeneous systems.
2013.

M. Mansuri, J. E. Jaussi, J. T. Kennedy, T.-C. Hsueh, S. Shekhar, G. Balamurugan, F. O'Mahony, C. Roberts, R. Mooney, , and B. Casper.
A scalable 0.128-to-1tb/s 0.8-to-2.6pj/b 64-lane parallel i/o in 32nm cmos.
2013.

J. Poulton, R. Palmer, A. M. Fuller, T. Greer, J. Eyles, W. J. Dally, and M. Horowitz.
A 14-mw 6.25-gb/s transceiver in 90-nm cmos.
2007.


J. W. Poulton, W. J. Dally, X. Chen, J. G. Eyles, T. H. Greer, S. G. Tell, J. M. Wilson, and C. T. Gray.
Ground-referenced single-ended short-reach serial link in 28 nm cmos for advanced packaging applications.
2013.


B. White, E. Bakker, P. Hamid, O. Lascu, F. Nogal, F. Packheiser, V. R. Jr., K.-E. Stenfors, E. Ufacik, and C. Zhu.
Ibm zenterprise 196 technical guide.
2011.