Rethinking Memory System Design (along with Interconnects)

Onur Mutlu

Carnegie Mellon University

http://www.ece.cmu.edu/∼omutlu

Abstract

The memory system is a fundamental performance and energy bottleneck in almost all computing systems. Recent system design, application, and technology trends that require more capacity, bandwidth, efficiency, and predictability out of the memory system make it an even more important system bottleneck [27, 28]. At the same time, DRAM technology is experiencing difficult circuit and device scaling challenges that make the maintenance and enhancement of its capacity, energy-efficiency, and reliability significantly more costly with conventional techniques (see, for example [7, 8, 11, 12, 15, 17, 18, 22, 23, 32]).

In this talk, we examine some promising research and design directions to overcome challenges posed by memory scaling. Specifically, we discuss three key solution directions: 1) enabling new memory architectures, functions, interfaces, and better integration of the memory and the rest of the system, including interconnects (e.g., [1, 2, 19, 20, 34–36]), 2) designing a memory system that intelligently employs multiple memory technologies and coordinates memory and storage management using non-volatile memory technologies (e.g., [16–18, 24, 25, 32, 33, 40–42]), 3) providing predictable performance and QoS to applications sharing the memory system (e.g., [3, 9, 10, 13, 14, 26, 29, 37–39]). As we discuss challenges and solution directions in memory, we will point out research opportunities in interconnects and memory-interconnect co-design (e.g., [2, 4–6, 19, 21, 30, 31]).

Categories and Subject Descriptors B.3.1 [*Memory Structures*]: General; C.0 [*Computer Systems Organization*]: General
General Terms Algorithms, Design, Performance
Keywords Memory systems, DRAM, processing in memory, multicore, interconnects, quality of service, persistent memory

References

[1] J. Ahn et al. PIM-Enabled Instructions: A Low-Overhead, Locality-Aware Processing-in-Memory Architecture. In *ISCA*, 2015.

[2] J. Ahn et al. A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing. In *ISCA*, 2015.

[3] R. Ausavarungnirun et al. Staged memory scheduling: Achieving high performance and scalability in heterogeneous systems. In *ISCA*, 2012.

[4] R. Das et al. Application-aware prioritization mechanisms for on-chip networks. In *MICRO*, 2009.

[5] R. Das et al. Aergia: Exploiting packet latency slack in on-chip networks. In *ISCA*, 2010.

[6] R. Das et al. Application-to-core mapping policies to reduce memory system interference in multi-core systems. In *HPCA*, 2013.

[7] H. David et al. Memory power management via dynamic voltage/frequency scaling. In *ICAC*, 2011.

[8] Q. Deng et al. Memscale: active low-power modes for main memory. In *ASPLOS*, 2011.

[9] E. Ebrahimi et al. Fairness via source throttling: a configurable and high-performance fairness substrate for multi-core memory systems. In *ASPLOS*, 2010.

[10] E. Ebrahimi et al. Prefetch-aware shared-resource management for multi-core systems. In *ISCA*, 2011.

[11] U. Kang et al. Co-architecting controllers and DRAM to enhance DRAM process scaling. In *The Memory Forum*, 2014.

[12] S. Khan et al. The efficacy of error mitigation techniques for DRAM retention failures: A comparative experimental study. In *SIGMETRICS*, 2014.

[13] H. Kim et al. Bounding memory interference delay in COTS-based multi-core systems. In *RTAS*, 2014.

[14] Y. Kim et al. Thread cluster memory scheduling: Exploiting differences in memory access behavior. In *MICRO*, 2010.

[15] Y. Kim et al. Flipping bits in memory without accessing them: An experimental study of DRAM disturbance errors. In *ISCA*, 2014.

[16] E. Kultursay et al. Evaluating STT-RAM as an energy-efficient main memory alternative. In *ISPASS*, 2013.

[17] B. C. Lee et al. Architecting phase change memory as a scalable DRAM alternative. In *ISCA*, 2009.

[18] B. C. Lee et al. Phase change memory architecture and the quest for scalability. *CACM*, 53(7), 2010.

[19] D. Lee et al. Tiered-latency DRAM: A low latency and low cost DRAM architecture. In *HPCA*, 2013.

[20] D. Lee et al. Adaptive-latency DRAM: Optimizing DRAM timing for the common-case. In *HPCA*, 2015.

[21] D. Lee et al. Decoupled Direct Memory Access: Isolating CPU and IO Traffic by Leveraging a Dual-Data-Port DRAM. In *PACT*, 2015.

[22] J. Liu et al. RAIDR: Retention-aware intelligent DRAM refresh. In *ISCA*, 2012.

[23] J. Liu et al. An experimental study of data retention behavior in modern DRAM devices: Implications for retention time profiling mechanisms. In *ISCA*, 2013.

[24] Y. Lu et al. Loose-ordering consistency for persistent memory. In *ICCD*, 2014.

[25] J. Meza et al. A case for efficient hardware-software cooperative management of storage and memory. In *WEED*, 2013.

[26] S. Muralidhara et al. Reducing memory interference in multi-core systems via application-aware memory channel partitioning. In *MICRO*, 2011.

[27] O. Mutlu. Memory scaling: A systems architecture perspective. In *IMW*, 2013.

[28] O. Mutlu and L. Subramanian. Research problems and opportunities in memory systems. *SUPERFRI*, 2014.

[29] O. Mutlu et al. Parallelism-aware batch scheduling: Enhancing both performance and fairness of shared DRAM systems. In *ISCA*, 2008.

[30] G. Nychis et al. Next generation on-chip networks: What kind of congestion control do we need? In *HotNets*, 2010.

[31] G. Pekhimenko et al. Toggle-Aware Compression for GPUs. *IEEE Comp. Arch. Letters*, 2015.

[32] M. K. Qureshi et al. Scalable high performance main memory system using phase-change memory technology. In *ISCA*, 2009.

[33] J. Ren et al. Dual-scheme checkpointing: A software-transparent mechanism for supporting crash consistency in persistent memory systems. In *MICRO*, 2015.

[34] V. Seshadri et al. RowClone: Fast and efficient In-DRAM copy and initialization of bulk data. In *MICRO*, 2013.

[35] V. Seshadri et al. Gather-Scatter DRAM: In-DRAM Address Translation to Improve the Spatial Locality of Non-unit Strided Accesses. In *MICRO*, 2015.

[36] V. Seshadri et al. Fast Bulk Bitwise AND and OR in DRAM. *IEEE Comp. Arch. Letters*, 2015.

[37] L. Subramanian, D. Lee, V. Seshadri, H. Rastogi, and O. Mutlu. The blacklisting memory scheduler: Achieving high performance and fairness at low cost. In *ICCD*, 2014.

[38] L. Subramanian et al. MISE: Providing performance predictability and improving fairness in shared main memory systems. In *HPCA*, 2013.

[39] L. Subramanian et al. The application slowdown model: Quantifying and controlling the impact of inter-application interference at shared caches and main memory. In *MICRO*, 2015.

[40] H. Yoon et al. Row buffer locality aware caching policies for hybrid memories. In *ICCD*, 2012.

[41] H. Yoon et al. Efficient data mapping and buffering techniques for multi-level cell phase-change memories. *TACO*, 2014.

[42] J. Zhao et al. FIRM: Fair and high-performance memory control for persistent memory systems. In *MICRO*, 2014.

NoCArc'15, December 5, 2015, Waikiki, Hawaii, USA.
ACM .