Work in Progress - Exploring Performance and Power of CXL Memory and PCIe 5.0 NVMe for Memory-Hungry Workloads

MARIA BAHNA, The University of Edinburgh, United Kingdom
DAVID FITZSIMMONS, The University of Edinburgh, United Kingdom
ANTONIO BARBALACE, The University of Edinburgh, United Kingdom

The recent convergence of storage and memory, enabled by PCIe 5.0 NVMe SSDs and CXL 2.0 memory expanders, offers promising solutions to memory bottlenecks of memory-hungry workloads. Both technologies can extend main memory (DDR RAM) at least for capacity. We analyse the performance and energy efficiency of these technologies on an example memory-hungry workload: quantum classical simulation [2]. We believe it is worth exploring how such new storage and memory technologies – PCIe 5.0 NVMe SSDs and CXL 2.0 memory expanders, actually improve the execution of memory-hungry workloads [5], which is especially important today with the global memory market price inflation of memory chips [1].

In this work, we focus on quantum classical simulations – specifically, state vector, because of the simple structure of the problem: an arbitrarily large vector of complex numbers that is continuously multiplied by relatively small and sparse square matrices of complex numbers. It is indeed memory-bound, as the required memory increases exponentially, $2^n \times 16$ bytes for a n size system [2].

CXL Type 3.0 devices are recognised as memory by Linux (namely, CPU-less NUMA – Non-Uniform memory Access, nodes). Hence, blocks of memory can be allocated from CXL rather than DDR RAM (directly attached). Because CXL access latency is longer, Linux introduced weighted interleaving [3]. For each requested block of virtual memory, physical CXL and DDR pages are allocated one after the other with a certain ratio. We find that the optimal page interleaving strategy is a 3:1 ratio, as it provides a bandwidth closest to the DDR RAM bandwidth $93GB/s$. Basically, for the nodes holding DDR we set a weight of 3 while CXL nodes have a weight of 1; for every three pages sent to the DDR, we send one page to the CXL.

NVMe SSDs are introduced by both naïvely using OS swap subsystem and by building new functionalities atop existing solution [4], featuring the techniques from the paper: triple-buffered pipeline, asynchronous I/O, three-step swap algorithm, and partitioning strategy. Our SSDs are partitioned using the OS *RAID 0* software to automate load balancing of I/O. The latter is used in the following analysis as it outperforms the naïve option.

We disclose the following **key observations** by comparing 3 configurations: CXL with OS-enabled weighted interleaving, NVMe SSD with rewritten application to use it, and the machine with doubled DDR RAM. (1) CXL is competitive with double DDR RAM in both performance and energy. Average execution time difference between the two configurations across varying thread counts stays under 30%, while energy consumption under 35%. When all DDR RAM slots are full, CXL is the only solution to expand memory. (2) Using half CPU cores, and directly connected memory, the SSD's performance and energy usage is similar to the CXL configuration; while using all CPU cores is more power hungry than using SSDs. (3) The performance gap becomes a worthwhile tradeoff as the SSD version allows significant simulation size increases.

Authors' Contact Information: Maria Bahna, maria.bahna@ed.ac.uk, The University of Edinburgh, Edinburgh, Scotland, United Kingdom; David Fitzsimmons, dfitzsim@ed.ac.uk, The University of Edinburgh, Edinburgh, Scotland, United Kingdom; Antonio Barbalace, antonio.barbalace@ed.ac.uk, The University of Edinburgh, Edinburgh, Scotland, United Kingdom.

References

[1] Jason England. 2025. RAM Prices Are Exploding: Here's Why and Everything You Need to Know About Surviving RAMageddon. https://www.tomsguide.com/computing/ram-prices-are-exploding-heres-why-and-everything-you-need-to-know-about-surviving-ramageddon. Accessed: 2026-02-14.

[2] Tyson Jones, Bálint Koczor, and Simon C. Benjamin. 2023. Distributed Simulation of Statevectors and Density Matrices. arXiv:2311.01512 [quant-ph] https://arxiv.org/abs/2311.01512

[3] MemVerge. 2023. *Introducing Weighted Interleaving in Linux for Enhanced Memory Bandwidth Management.* https://memverge.ai/introducing-weighted-interleaving-in-linux-for-enhanced-memory-bandwidth-management/ Accessed: 2026-02-13.

[4] Daeyoung Park, Heehoon Kim, Jinpyo Kim, Taehyun Kim, and Jaejin Lee. 2022. SnuQS: scaling quantum circuit simulation using storage devices. In *Proceedings of the 36th ACM International Conference on Supercomputing* (Virtual Event) *(ICS '22)*. Association for Computing Machinery, New York, NY, USA, Article 6, 13 pages. doi:10.1145/3524059.3532375

[5] Xi Wang, Jie Liu, Jianbo Wu, Shuangyan Yang, Jie Ren, Bhanu Shankar, and Dong Li. 2025. Performance Characterization of CXL Memory and Its Use Cases. In *2025 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 1048–1061. doi:10.1109/IPDPS64566.2025.00097