Analytically Modeling NVM Design Trade-Offs

Lillian Pentecost*, Marco Donato*, Akshitha Sriraman+, Gu-Yeon Wei*, David Brooks*
Harvard University*, University of Michigan+

Abstract

As considerable ongoing research is put towards developing emerging non-volatile memories, there is a growing interest in evaluating the potential of these technologies for enabling efficient compute in a variety of application spaces. While simulation tools for characterizing different NVMs at the array level exist, the increasingly complex design space introduced by emerging NVMs together with the need to assess the impact of application-specific performance and energy requirements demands the ability to conduct extensive and efficient system-level analysis. Our proposed, generalized framework provides an opportunity for analyzing the performance, power, and area implications of leveraging a particular proposed NVM technology under a specific use case.

1 Introduction

We propose an analytical framework for quantifying the potential benefits and studying design trade-offs in terms of overall performance, power, and area characteristics across different NVM technologies Our analytical model defines program behavior by the frequency and size of read and write accesses. We design a flexible, customizable interface to NVSim [2] to define memory cell definitions, explore a multitude of configuration settings (e.g., capacity, access characteristics, process node), and extract and synthesize memory array characterization results in order to evaluate an end-to-end analytical model. This model is additionally parameterized to take into account technology properties under active research, such as write retention, write endurance, and the characteristics of different programming strategies (e.g., multi-level cell programming).

There are many potential use cases for this generic framework, such as: (1) identifying program behavior characteristics that maximize potential benefits for a given NVM technology (for example, what is the more energy-efficient memory for an application that is dominated by frequent writes?); (2) determining which set of technology choices is best suited for a given program based on its behavior and other application-level constraints, such as latency requirements or power budget; or (3) conducting expansive design sweeps to maximize performance or lifetime of a memory solution in light of write characteristics.

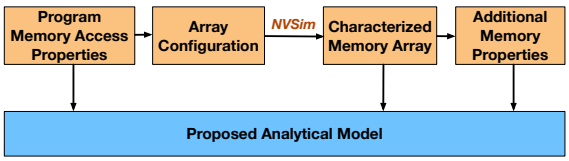

Figure 1: Access patterns are defined and passed to the array configuration, which has a technology-specific cell definition and additional parameters. This configuration is characterized using NVSim, and those outputs are in turn leveraged to inform other properties and evaluate the memory solution using our analytical model.

2 Analytical Framework

Figure 1 gives an overview of the inputs to our analytical framework for evaluating NVM benefits across different technology proposals.

Program Memory Access Properties include the size and frequency of reads and writes, in addition to the working set size of the program. These properties can be varied for limit studies that indicate what types of program characteristics and access patterns are most suitable for a given NVM array configuration, and additional program properties (e.g., different phases of execution or program duration) can also be taken into account.

Array Configurations identify which memory technology to evaluate (via a flexible interface to use existing or modified NVSim cell definitions), in addition to specifying the chosen array capacity, word width, process node, optimization target, and access characteristics. In this way, our framework provides an interface to perform exhaustive design space exploration across array configurations and memory technologies using NVSim.

Characterized Memory Array results are pro-

duced using a minimally-modified version of NVSim [2, 4], and our framework provides an additional interface for post-processing results to serve as inputs to the unified analytical model, in which array characteristics are evaluated in light of the specific access pattern.

Additional Memory Properties such as write endurance and write retention characteristics, as well as maximum operating frequency, can be either directly be extracted using NVSim results or separately extracted from relevant publications. These parameters provide an opportunity for additional design space exploration. For example, this framework can answer the question of what write endurance would be required by a given NVM in order to avoid degradation for a certain period of time for a specific application write pattern.

Proposed Analytical Model is our final evaluation of the overall power and performance of a particular memory array when utilized according to the defined memory access pattern. This model leverages NVSim results for the dynamic energy and latency per read or write access in order to have a first-order model of the total operating power, the lifetime of the memory, the bandwidth utilization, and other important metrics. Our evaluation framework also allows for simple integration of NVM fault models as defined in [4] and [1], which is implemented using Ares [5], an open-source, application-level fault injection tool. We are actively developing the model to additionally include the impact of write buffering, but, as we discuss next, even the simplest version of this framework provides a powerful interface for design space explorations and technology comparisons.

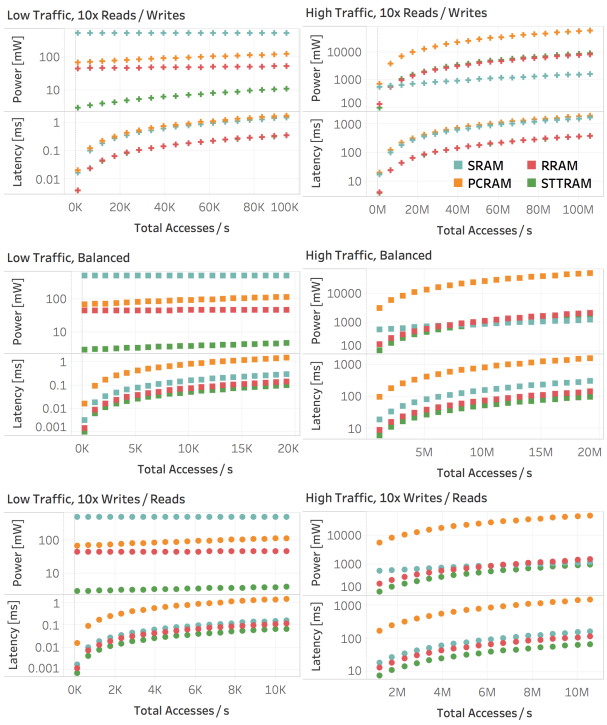

Figure 2: Varying access patterns between low (left) and high (right) memory traffic and the ratio between the number of reads and writes (10x more reads than write, balanced traffic, or 10x more writes than reads, from top to bottom). Optimal memory choice in terms of both power and latency differs depending on access frequency and read vs. write ratio.

3 Case Study

For a sampling of NVM solutions and SRAM, we characterize memory arrays with equivalent capacity (16MB), word width (64 bits), and optimization target (Read Energy-Delay-Product). Next, we use our proposed framework to study how these arrays compare in terms of total power (leakage and dynamic power) and total latency incurred due to memory accesses for a variety of memory access patterns.

The results of our study are shown in Figure 2 for different sets of access properties including when there are 10× more reads than writes (top, as is common when reading feature vectors for ML workloads [3]) or there are 10× more writes than reads (bottom, as is common for logging user requests in client-facing frontend datacenter services [6]). On the left, we observe the behavior of these memories under relatively low number of memory accesses per second, varying the frequency of reads vs. writes. Independent of the ratio between reads and writes, we observe that NVM solutions maintain consistent power savings relative to SRAM under low traffic while maintaining performance, with the exception of PCRAM. In fact, we observe that RRAM and STTRAM exhibit significantly higher performance benefits compared to SRAM only when there are 10× more reads than writes due to their longer write latency.

When we observe the same trends for higher traffic (Figure 2, right), we observe a traffic threshold above which NVM solutions no longer offer a power advantage (about 6 million accesses for the case in which there are 10× more reads than writes). Interestingly, though the power consumption for all memories characterized increases as the access pattern becomes more write-dominated, NVMs are still able to maintain an advantage up to a similar threshold as in the read-dominated case.

Thus, the trade-offs between power and latency

across different program characteristics are non-trivial and result in different optimal NVM choices. This limited study is just one example of the type of exploration that is made possible to easily evaluate and reason about because of our proposed framework.

4 Conclusion & Potential Impact

Our analytical model can serve as a first-order design guide for researchers at different levels of the computing stack. Application experts eager to determine whether NVM storage solutions may offer performance, power, or area benefits can easily do so, and perhaps even get guidance as to which technologies are most promising or what application-level changes could further increase benefits. Architects and systems experts can examine trends in performance and power trade-offs under different assumptions of array capacity, configuration, and NVM technologies to inform and guide deeper exploration of NVM storage solutions. Device researchers can utilize this high-level model as a proxy to estimate the potential benefits of a proposed technology change (e.g., optimistic scaling assumptions or a new programming technique to improve endurance) before increasing investment or performing more rigorous studies. Furthermore, having a unifying analytical model like the one we propose in this work would enable co-design optimizations by helping designers identify cross-layer limitations and opportunities.

In short, our proposed framework allows the entire design space of emerging NVM technologies to be more navigable and approachable, and we provide an efficient first-order approximation of the performance, power, and area of NVM-based solutions by building atop NVSim array characterization results in conjunction with information about program behavior and other technology properties.

Acknowledgments

This work was supported in part by Applications Driving Architectures (ADA), a Semiconductor Research Corporation (SRC) JUMP center.

References

[1] Marco Donato, Brandon Reagen, Lillian Pentecost, Udit Gupta, David Brooks, and Gu-Yeon Wei. On-chip deep neural network storage with multi-level envm. In *2018 The 55th Annual Design Automation Conference (DAC)*, June 2018.

[2] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi. Nvsim: A circuit-level performance, energy, and area model for emerging nonvolatile memory. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, July 2012.

[3] K. Hazelwood, S. Bird, D. Brooks, S. Chintala, U. Diril, D. Dzhulgakov, M. Fawzy, B. Jia, Y. Jia, A. Kalro, J. Law, K. Lee, J. Lu, P. Noordhuis, M. Smelyanskiy, L. Xiong, and X. Wang. Applied machine learning at facebook: A datacenter infrastructure perspective. In *HPCA 2018*, Feb 2018.

[4] L. Pentecost, M. Donato, B. Reagen, U. Gupta, S. Ma, G. Wei, and D. Brooks. Maxnvm: Maximizing dnn storage density and inference efficiency with sparse encoding and error mitigation. *MICRO 2019*, October 2019.

[5] Brandon Reagen, Lillian Pentecost, Udit Gupta, Paul Whatmough, Sae Kyu Lee, Niamh Mulholland, David Brooks, and Gu-Yeon Wei. Ares: A framework for quantifying the resilience of deep neural networks. In *2018 The 55th Annual Design Automation Conference (DAC)*, June 2018.

[6] Rahul Soni. *Nginx*. Springer, 2016.