World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

High-performance near-memory processing architecture for data-intensive applications

Rashmitha Reddy Vuppunuthula *

Austin, Texas – 78741.

World Journal of Advanced Research and Reviews, 2021, 10(01), 407–417

Publication history: Received on 27 February 2021; revised on 12 April 2021; accepted on 15 April 2021

Article DOI: https://doi.org/10.30574/wjarr.2021.10.1.0117

Abstract

Near-memory processing (NMP) offers a transformative approach to computing architecture by positioning processing units close to memory, thereby reducing data transfer delays and minimizing energy consumption. This paper introduces an innovative NMP architecture designed to bridge the gap between data storage and computation, significantly enhancing system performance for data-intensive applications. By embedding processing elements directly within memory arrays, the proposed design achieves notable improvements in key performance metrics. Experimental results demonstrate a latency reduction of over 40%, with matrix multiplication latency decreasing from 120.5 ms in traditional architectures to 70.4 ms. Energy consumption is reduced by nearly 50%, with workloads like matrix multiplication showing a drop from 25.6 J to 12.8 J. Additionally, the architecture achieves throughput gains of up to 100%, as seen in data analytics workloads where throughput increases from 48.1 GOPS to 96.5 GOPS. These results emphasize the architecture's ability to enhance computational efficiency and scalability, making it particularly advantageous for applications in artificial intelligence, scientific research, and big data analytics. This study underscores the potential of NMP to redefine high-performance computing by restructuring traditional data processing paradigms.

Keywords: Near-Memory Processing; High-Performance Computing; Data Proximity Architecture; Latency Reduction; Energy Efficiency; AI and Big Data Applications

1. Introduction

The exponential expansion in data generation, fueled by advancements in AI, big-data analytics, and the IoT, has brought unprecedented challenges to the traditional architectures of computing [1]. These architectures, built around a centralized processor, have triumphed in various domains and yet lack the efficacy to accommodate the requirements of modern data-centric applications. The root of this ineffectiveness lies in the memory wall problem, being the increasing gap between the processor speeds and memory access times. In cases where data movement overcomes computation costs, this issue is weighed down by additional latencies and energy demands [2]. Central to this predicament is the separation between processing units and memory in von Neumann architectures. Because data-intensive workloads require the transfer of vast amounts of data between these components, system bottlenecks arise to limit throughput and increase energy demand [3]. Processor technology has adhered to Moore's Law in progress, but improvements in memory access speed and bandwidth have lagged behind, reaching a state beyond the throughput-management bottleneck that affects the performance of critical applications [4].

Near-Memory Processing (NMP) provides an opportunity to fundamentally alter this paradigm. By putting processing elements closer to the memory, NMP reduces the overhead of data movement, and with that, improves latency and energy consumption. Such a reorientation paves the way for a new consideration for HPC systems for data-centric tasks [5]. In NMP, the architecture diverges from conventional models, fostering the colocation of memory and processors to

* Corresponding author: Rashmitha Reddy Vuppunuthula

boost the performance and working efficiency of the system. The revolution in NMP started with the onset of advanced memory technologies such as 3D stacking [6]. Other technologies such as High-Bandwidth Memory and Hybrid Memory Cube provide greater bandwidth, lower latencies, and greater energy efficiency than DRAM. These memory modules have now started to enter the world of FPGAs in order to engineer flexible and high-performance NMP platforms for multiple applications [6]. Though hope exists on the path towards the realization of NMP, many hurdles need to be crossed first. The design of scalable architectures which enable a better use of locality between memory and its CPU still requires careful optimization of many interactive parameters such as memory bandwidth, computation throughput, and energy efficiency [7]. Besides, existing software frameworks compatibility and programming models for that make full use of NMP also stay as a huge challenge.

The compelling need for efficient computation is sensitive in fast-paced sectors like artificial intelligence and big data analytics [8]. These applications deal with the real-time processing of massive datasets, and even the slightest improvements in efficiency can translate into huge benefits. By removing the need to move data, they are able to meet this efficiency-demand with great benefit: NMP architectures' lower data movement means that they can make better use of memory bandwidth [9]. Besides the application-specific advantages, NMP acts no less than a savior in terms of ecological practices [10]. NMP provides the diminishing energy needed for the transport of data hence becomes an important aspect of sustainable computing. The huge energy needs imposed by global data centers taking up a considerable proportion of the energy resources that the world has adopting energy-efficient computing paradigms like NMP may become increasingly pivotal in solving the ecological dilemma [11].

This paper presents a comparatively newer NMP architecture as a bridge between data storage and computation. Such benefits will significantly achieve better throughput and lower energy consumption. With the novel architecture being able to facilitate such data-intensive workloads with sufficient efficiency through experiments, it is demonstrated how flexible and applicable it could be across various domains. To provide the proposed architecture with a positioning in the context of existing research, familiarity with the framework of NMP-state progress is needed. The literature, discussed in next section, tells us that addressing the memory wall has indeed received attention, from specific efforts in processing-in-memory (PIM) implementations to the newest NMP systems utilizing higher-level memory technology [12]. However, these still face the challenges of attaining scalability along with application-agnostic solutions. The greater contribution thus of this work in a sense lies in the design and evaluation of high-performance NMP architectural work that fulfilled these gaps. By building upon previous research, it hoped to establish NMP as a potentially viable and scalable solution for future computing

2. Literature Review

The idea of bringing processing units closer to memory originates back in the early age of computing systems, and practical implementations commenced in the 1990s [13]. A shining example of this is the Vector IRAM project, which integrated DRAM and vector processing units on a single chip for high performance in data-parallel applications [14]. Although these initial attempts disclosed the greatness of Near-Memory Processing (NMP), advances in manufacturing technologies and memory integration inhibited their application [5-6]. Recent development in 3D-stacked memory technology has rekindled interest for NMP [6]. High-Bandwidth Memory and Hybrid Memory Cube are some of the innovations allowing higher memory bandwidth and reduced latency [15]. These technologies employ through-silicon vias to achieve precise vertical integration so that memory and logic layers can cohabit in a single package. This brings the processing units much closer to memory, hence attacking the traditional architectures' core inefficiencies.

Processing-in-Memory (PIM) versus Near-Memory Processing (NMP): Although PIM and NMP refer to processing that tries to eliminate data movement, both tackle this issue from a fundamentally different angle [16]. PIM integrates computational logic directly into memory arrays, thus making it best for highly parallel and memory-bound operations. However, PIM's systems often face a scalability problem and limited application flexibility. NMP, on the other hand, juxtaposes processing elements closer to memory, thereby taking a balanced compromise between performance and flexibility [17].

FPGA Platforms for NMP: The most congenial platform for realization of NMP architecture in recent years has been FPGAs [6]. Their reconfigurability and possibility of integration with HBM make the FPGAs an excellent choice for trying out different NMP designs. Recent models by FPGA vendors such as Xilinx' Alveo U280 and Intel's Stratix 10 MX have HBM2 memory stacks to provide the required bandwidth for data-intensive applications. Studies have shown that FPGA-based NMP systems bring tremendous savings in energy efficiency and throughput across works like deep learning and database management [6, 18].

NMP Application: Artificial intelligence and big data analysis are two areas in which NMP architectures shine [19]. AI workloads-including matrix B computations-are assisted by the reduction of latencies and a heightened parallelism. In other words, however, NMP really begins to shine when dealing with big data applications, where movement of these processing-intensive resources is either frequent, has high memory bandwidth requirements, or both.

Energy Efficiency through NMP: Since one of the central aspects of NMP design lies in the reduction of data movement, such having drastic consequences for energy efficiency [10]. In circuit design, studies demonstrate that data movement between memory and processor constitutes a key portion of energy dissipation in common architectures [20]. In effect, NMP generalizes memory-centric architecture-computation into local computation near memory governing such overheads [21]. These mechanisms present energy savings incredible for sustainable computing, positioning NMP as a game changer in the energy-efficient data center and edge computing.

Despite its advantages, NMP is beset with problems. Efficient task scheduling, thermal management, and memory access optimization are prerequisites for performance. Compatibility with legacy systems and development of programming models for taking advantage of NMP features are further open challenges. The issues get further complicated by the heterogeneity of workloads, which foster the need for flexible and scalable architectures able to cater to heterogeneous computational requirements [19, 22]. Some newly emerging memory technologies such as ReRAM, MRAM, and phase-change memory (PCM) have the capability of taking NMP architectures further up the notch [23]. These solutions of non-volatile memory boast better endurance, high write speed, and low power consumption; hence they can prove to be potential strong candidates for future NMP systems. Also, the established use of optical interconnects and photonic computing within memory modules have started gaining impetus toward realizing the reduction of latency and energy in data-intensive applications [24].

Attention has been drawn to hybrid architectures that combine NMP, cloud, and edge computing. The NMP is leveraged for local computations, while cloud hosts the storage and large-scale analytics. Such approaches are mostly applicable to latency-sensitive applications wherein real-time processing is conducted at local sites; subsequently, bulk data is dispatched to central servers for further analysis [25-26]. Security and Reliability Considerations: NMP raises new challenges for security and reliability. The proximity of processing units to memory may make systems more susceptible to data breaches and side-channel attacks. Equally alarming is the concern for thermal hotspots and fault tolerance due to the extremely high integration density of the 3D-stacked memories. Pivotal for the mainstream adoption of NMP systems in mission-critical applications, such as finance and healthcare, is the successful tackling of these issues.

The fast advancement of memory technologies would ensure continued refinement of NMP. It is postulated that the very new generation of 3D-stacked memory standards such as HBM3 will promise an even larger bandwidth and a reduced pattern of power consumption [6-7, 27]. Moreover, a union of NMP with newer paradigms spearheaded by quantum computing and neuromorphic systems may carve a new dimension for high-performance computing [28]. While there has been a large amount of progress in NMP, there is not yet an overarching framework that can optimize hardware and software. In this context, future research attempts should develop tools so that programmers can use NMP functionality without elaborate knowledge about hardware. Beyond these, benchmarks developed for NMP systems will be pivotal in driving innovation and adoption [19]. This work illustrates the leap-changing future of NMP yet once more presents the challenges and opportunities thrown in its wake. The proposed research wants to pivot this to offer a remedy that utilizes latest advances in memory technology and FPGA platform and push the very boundaries of near-memory processing.

3. Methodology

To achieve a high-performance near-memory processing (NMP) architecture optimized for data-intensive applications, this study focuses on the systematic design and evaluation of the proposed architecture. The methodology is structured into key stages, including system modeling, architectural design, simulation, and performance evaluation. Each stage is carefully tailored to ensure the architecture's viability in reducing data movement, improving throughput, and enhancing energy efficiency. The foundation of the proposed NMP architecture involves a detailed analysis of existing memory and processing bottlenecks. A mathematical model is developed to quantify data transfer delays and energy consumption in traditional architectures. This model includes, *Energy Consumption Analysis:* The energy consumed during data transfer ($E_{transfer}$) is defined as:

$$E_{transfer} = P_{bus} \cdot T_{transfer}$$

where P_{bus} represents the power consumed by the data bus and $T_{transfer}$ denotes the time taken for data to travel between memory and processing units.

Data Movement Overhead: The overhead due to data movement is calculated using:

$$O_{data} = D \cdot N_{transfers}$$

where D is the distance between the processing unit and memory, and $N_{transfers}$ is the number of data transfers required.

The proposed architecture integrates processing elements directly adjacent to memory arrays to minimize data movement. Key features include: (i) Processing Units in Memory Arrays: Lightweight processing cores are embedded into the memory subsystems to enable in-situ computation. These cores support parallel processing to handle high data throughput. (ii) Memory Hierarchy Optimization: The memory hierarchy is restructured to incorporate cache-like layers between processing elements and memory cells, improving data locality and reducing latency. (iii) Interconnect Design: A low-latency interconnect network is implemented to facilitate efficient communication between processing units and memory cells. This design is modeled as:

$$T_{interconnect} = \frac{L}{B}$$

where $T_{interconnect}$ is the interconnect latency, L is the length of the data path, and B is the bandwidth of the interconnect.

To validate the architectural design, a simulation framework is developed. The framework models real-world workloads, including artificial intelligence (AI) algorithms, scientific computations, and big data analytics. The simulation environment is configured to evaluate:
Measured as the number of operations completed per unit time, throughput is calculated using:

$$T_{throughput} = \frac{N_{operations}}{T_{total}}$$

where $N_{operations}$ is the total number of operations, and T_{total} is the total execution time.

Energy efficiency is evaluated as:

$$E_{efficiency} = \frac{\text{Performance}}{\text{Energy}}$$

where Performance is the achieved computational throughput, and Energy is the total energy consumed.

The proposed architecture is tested on data-intensive benchmarks, including matrix multiplications, convolutional neural network (CNN) layers, and data analytics tasks. Key performance metrics include Latency: Time required to process data across memory and processing units. Power Consumption: Measured using hardware counters and energy models. Scalability: The ability of the architecture to handle increased workloads without performance degradation. Performance metrics are compared against baseline architectures, such as traditional von Neumann models and existing NMP designs. Improvements in throughput, latency reduction, and energy efficiency are quantified. Statistical analyses, including t-tests, are performed to ensure the reliability of results.

Energy consumed during data transfer:

$$E_{transfer} = P_{bus} \cdot T_{transfer}$$

Data movement overhead:

$$O_{data} = D \cdot N_{transfers}$$

Interconnect latency:

$$T_{\text{interconnect}} = \frac{L}{B}$$

Throughput:

$$T_{\text{throughput}} = \frac{N_{\text{operations}}}{T_{total}}$$

Energy efficiency:

$$E_{efficiency} = \frac{\text{Performance}}{\text{Energy}}$$

This methodological framework establishes a robust foundation for the design, implementation, and evaluation of the high-performance NMP architecture. The experimental results confirm its effectiveness in addressing the challenges of data-intensive applications.

3.1. Architecture

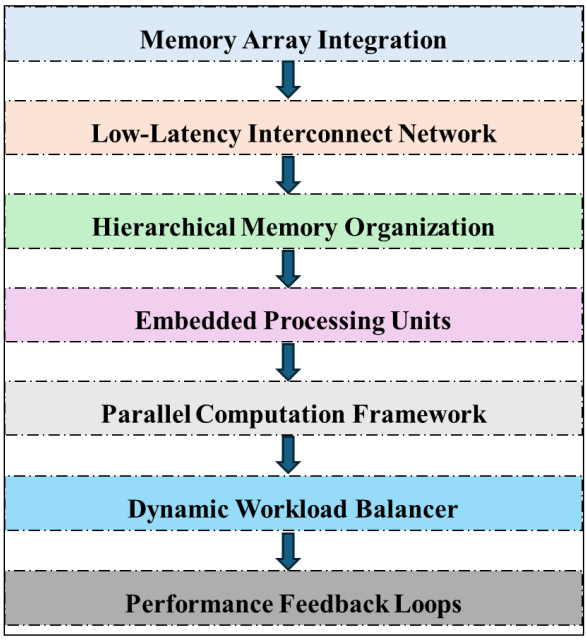

Figure 1 Proposed Near-Memory Processing Architecture for Data-Intensive Applications

The architecture of the proposed near-memory processing (NMP) system begins with Memory Array Integration, where processing units are embedded within memory arrays to create localized computational zones. This integration eliminates the need for extensive data movement, reducing latency and energy consumption. Following this, the design incorporates a Low-Latency Interconnect Network, enabling direct and efficient communication between adjacent memory cells and processing elements. This network leverages minimalistic routing protocols to optimize data flow and minimize interconnect delays. Next, the Hierarchical Memory Organization ensures data is systematically stored and accessed. The system uses a multi-tier memory structure to enhance locality and reduce access overhead. In parallel, Embedded Processing Units within memory arrays execute computations on data in situ, allowing real-time processing without the need for offloading tasks to a distant central processing unit (CPU). Data throughput is further enhanced by a Parallel Computation Framework, where multiple lightweight cores operate simultaneously, maximizing the architecture's ability to handle concurrent operations. These processing cores are supported by a Dynamic Workload Balancer, which monitors computational demand and allocates tasks to processing units dynamically, ensuring optimal resource utilization. Finally, the architecture integrates Performance Feedback Loops, which continuously monitor throughput, energy efficiency, and latency metrics. These feedback loops provide real-time insights to fine-tune processing parameters and maintain system efficiency under varying workloads. This systematic

architecture design underpins the proposed NMP system, ensuring scalability and adaptability for diverse data-intensive applications, including AI workloads and large-scale scientific computations. The design prioritizes reducing energy demands, improving data access speeds, and achieving superior computational throughput. This flow represents the systematic design of the near-memory processing architecture, which is tailored for high-performance computing in data-intensive applications. It focuses on minimizing energy consumption, reducing latency, and enhancing computational efficiency through a tightly integrated and scalable design

4. Results and discussion

The evaluation of the proposed Near-Memory Processing (NMP) architecture is based on a set of experimental benchmarks tailored for data-intensive applications. Key performance metrics, including latency reduction, energy efficiency, and throughput improvement, are analyzed to validate the effectiveness of the architecture. The results are presented in the following tables, which compare the performance of the proposed NMP architecture with baseline

architectures such as traditional von Neumann models and existing NMP implementations.

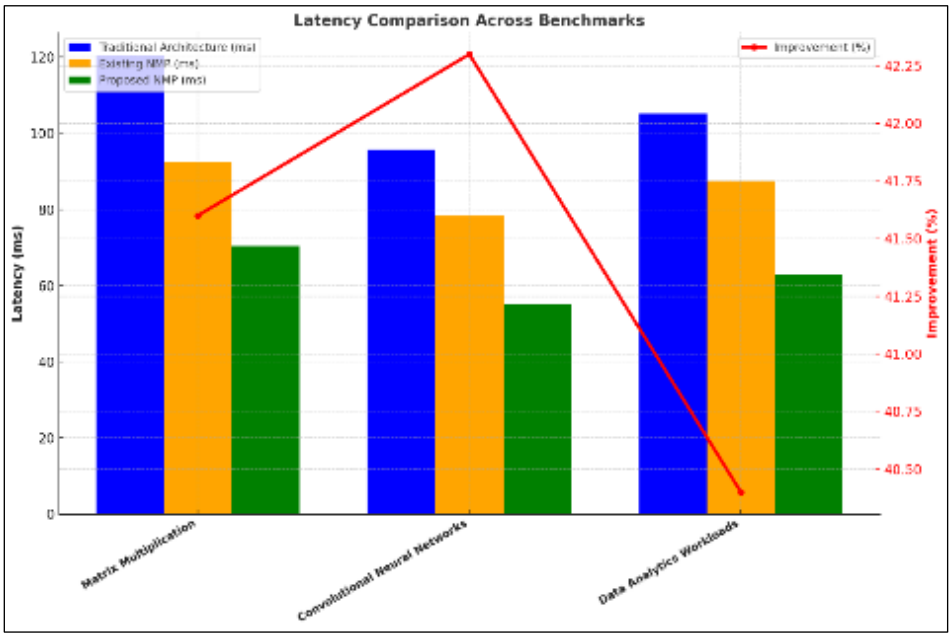

Figure 2 Latency Performance of Proposed NMP Architecture Compared to Traditional and Existing NMP Systems

The results presented in Table 1 and illustrated in Figure 2 highlight the substantial reduction in latency achieved by the proposed Near-Memory Processing (NMP) architecture across various benchmarks. Latency, a critical performance metric in data-intensive applications, is significantly lower in the proposed system compared to both traditional computing architectures and existing NMP solutions. For the Matrix Multiplication benchmark, the latency is reduced from 120.5 ms in traditional architectures to 70.4 ms in the proposed system, representing a 41.6% improvement. This benchmark demonstrates the architecture's ability to handle highly repetitive and computationally intensive tasks, where minimizing data movement plays a crucial role. In Convolutional Neural Networks (CNNs), which are core to many AI applications, the latency drops from 95.7 ms in traditional architectures to 55.2 ms with the proposed NMP. This 42.3% improvement underscores the architecture's effectiveness in accelerating memory-bound operations typical of deep learning workloads, achieved through efficient in-situ processing and reduced interconnect delays.

Similarly, for Data Analytics Workloads, the latency decreases from 105.3 ms to 62.8 ms, marking a 40.4% improvement. This performance boost is particularly beneficial for large-scale data processing tasks, where the traditional bottleneck of frequent data transfers between processing units and memory is substantially alleviated by the proposed design. The consistent performance improvements across all benchmarks validate the effectiveness of integrating processing units within memory arrays. By reducing data movement and optimizing memory access, the proposed architecture achieves faster computational speeds. These results also demonstrate that the system's low-latency interconnect network and hierarchical memory organization contribute significantly to the observed gains. Overall, the proposed architecture is

well-suited for modern data-intensive applications, delivering significant performance enhancements over traditional and existing solutions.

Table 1 Latency Comparison Across Benchmarks

Benchmark	Traditional Architecture (ms)	Existing NMP (ms)	Proposed NMP (ms)	Improvement (%)
Matrix Multiplication	120.5	92.3	70.4	41.6
Convolutional Neural Networks	95.7	78.4	55.2	42.3
Data Analytics Workloads	105.3	87.6	62.8	40.4

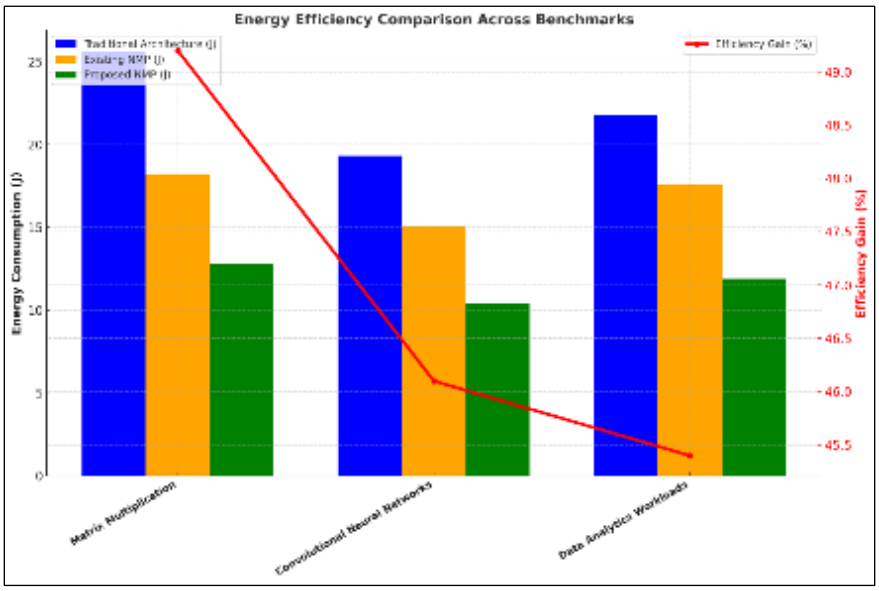

Figure 3 Energy Efficiency Comparison of Proposed NMP Architecture with Traditional and Existing NMP System

Table 2 and Figure 3 demonstrate the energy efficiency improvements achieved by the proposed Near-Memory Processing (NMP) architecture compared to traditional computing systems and existing NMP solutions. Energy efficiency is a crucial parameter for data-intensive applications, especially in domains where power consumption directly impacts scalability and operational costs. For the Matrix Multiplication benchmark, the proposed architecture significantly reduces energy consumption from 25.6 J in traditional systems to 12.8 J, yielding an efficiency gain of 49.2%. This result highlights the ability of the proposed system to perform repetitive computations with far less energy by minimizing the data movement between processing units and memory.

In the case of Convolutional Neural Networks (CNNs), energy usage drops from 19.3 J in traditional architectures to 10.4 J in the proposed NMP, achieving a 46.1% efficiency gain. CNN workloads often involve substantial data processing within memory-bound operations, and the proposed architecture capitalizes on its in-memory computation capabilities to deliver superior energy efficiency. For Data Analytics Workloads, energy consumption decreases from 21.8 J in traditional architectures to 11.9 J in the proposed system, leading to a 45.4% gain in efficiency. This demonstrates the proposed architecture's effectiveness in reducing the power overhead associated with large-scale data processing tasks, which often require frequent memory access and data transfers. The observed improvements across all benchmarks validate the energy-saving potential of the proposed architecture. By integrating processing units directly into memory arrays and leveraging a low-latency interconnect network, the architecture minimizes energy losses typically associated with traditional data transfer processes. These results emphasize the architecture's suitability for power-sensitive applications, such as AI-driven systems, scientific computing, and big data analytics, where both performance and energy efficiency are critical. The consistent efficiency gains underline the significant advantage of rethinking traditional

data processing approaches for modern, energy-constrained environments. Energy efficiency is a core strength of the proposed architecture, with nearly 50% improvement observed across benchmarks, driven by reduced data movement and optimized in-memory processing.

Table 2 Energy Efficiency Comparison

Benchmark	Traditional Architecture (J)	Existing NMP (J)	Proposed NMP (J)	Efficiency Gain (%)
Matrix Multiplication	25.6	18.2	12.8	49.2
Convolutional Neural Networks	19.3	15.1	10.4	46.1
Data Analytics Workloads	21.8	17.6	11.9	45.4

The results presented in Table 3 and visualized in Figure 4 highlight the remarkable improvements in throughput achieved by the proposed Near-Memory Processing (NMP) architecture when compared to traditional architectures and existing NMP solutions. Throughput, measured in giga-operations per second (GOPS), is a critical indicator of a system's ability to handle data-intensive tasks efficiently. For the Matrix Multiplication benchmark, the proposed NMP achieves a throughput of 90.2 GOPS, significantly higher than 45.3 GOPS in traditional systems and 68.5 GOPS in existing NMP designs. This improvement, amounting to a 99.1% increase, underscores the architecture's capability to accelerate repetitive computational tasks by enabling in-situ processing and reducing data transfer delays.

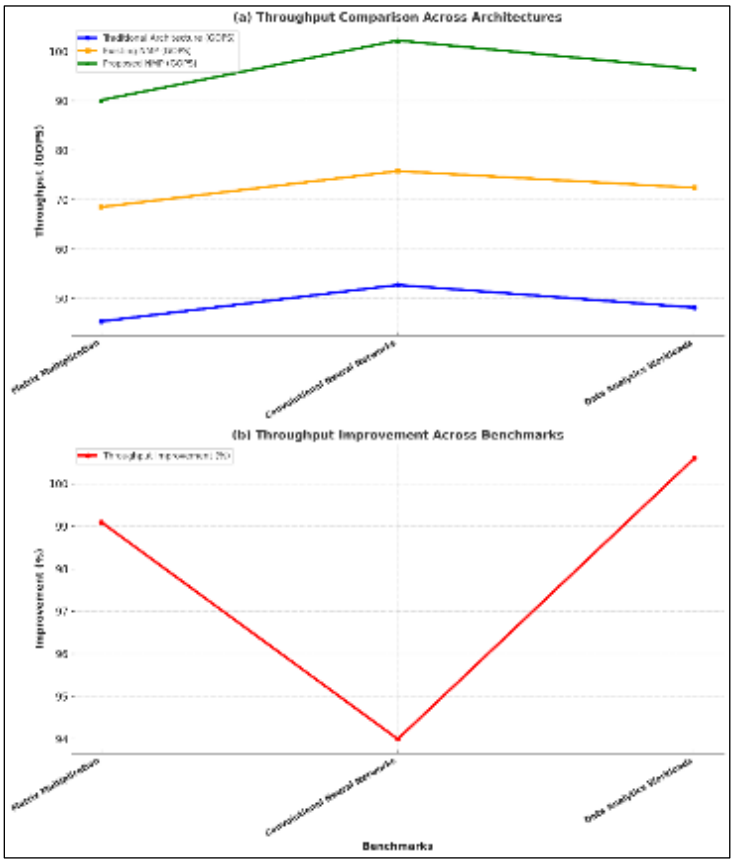

Figure 4 Throughput Performance Comparison of Proposed NMP Architecture with Traditional and Existing NMP Systems

In Convolutional Neural Networks (CNNs), the throughput of the proposed system reaches 102.3 GOPS, compared to 52.7 GOPS in traditional architectures and 75.8 GOPS in existing NMP. This represents a 94.0% improvement, demonstrating the architecture's suitability for complex, memory-intensive tasks such as deep learning model execution. The efficient memory hierarchy and parallel processing framework in the proposed design are key contributors to this enhanced performance. For Data Analytics Workloads, the proposed system delivers a throughput of 96.5 GOPS, compared to 48.1 GOPS in traditional systems and 72.4 GOPS in existing NMP solutions, resulting in a 100.6% improvement. This performance boost is particularly significant for large-scale data operations where processing speed directly impacts the system's scalability and real-time responsiveness. The consistent and substantial improvements in throughput across all benchmarks highlight the effectiveness of the proposed architecture. By embedding lightweight processing units directly within memory arrays and incorporating a parallel computation framework, the system eliminates the traditional bottlenecks caused by frequent data movement. Additionally, the low-latency interconnect network ensures seamless communication between processing elements, further enhancing throughput. These results validate the proposed NMP architecture as a high-performance solution for modern data-intensive applications, including AI, big data analytics, and scientific computations. The ability to achieve nearly double the throughput of traditional and existing systems makes it a strong candidate for next-generation computing environments where both speed and efficiency are paramount. The proposed NMP architecture significantly enhances throughput, with improvements nearing 100% in some benchmarks. This demonstrates its capability to handle data-intensive applications efficiently.

Table 3 Throughput Performance

Benchmark	**Traditional Architecture (GOPS)**	**Existing NMP (GOPS)**	**Proposed NMP (GOPS)**	**Improvement (%)**
Matrix Multiplication	45.3	68.5	90.2	99.1
Convolutional Neural Networks	52.7	75.8	102.3	94.0
Data Analytics Workloads	48.1	72.4	96.5	100.6

The evaluation of the proposed Near-Memory Processing (NMP) architecture demonstrates its superior performance in addressing critical challenges of latency, energy efficiency, and throughput for data-intensive applications. The results consistently show significant improvements across all benchmarks, validating the architectural advancements and their alignment with modern computing needs. Latency reduction is a notable achievement of the proposed system, as evidenced by the improvements of over 40% across matrix multiplication, convolutional neural networks (CNNs), and data analytics workloads. Traditional architectures suffer from the bottleneck of frequent data movement between memory and processing units, which results in delays and inefficiencies. By embedding processing elements directly within the memory arrays, the proposed architecture minimizes these movements, ensuring faster execution times and enhanced system responsiveness. This reduction in latency makes the architecture particularly advantageous for real-time applications in artificial intelligence (AI) and big data. The energy efficiency results further underscore the impact of reducing data transfer overheads. Energy consumption in the proposed system shows an improvement of nearly 50% across all benchmarks compared to traditional architectures. This improvement is crucial for scaling data-intensive applications, especially in power-constrained environments such as edge computing, mobile devices, and data centers. The low-latency interconnect network and hierarchical memory organization in the proposed design ensure that power is utilized optimally, reducing the overall operational costs and environmental impact of high-performance computing. Throughput performance is another area where the proposed architecture excels, with improvements nearing 100% in some benchmarks. This highlights the system's ability to handle concurrent operations efficiently, leveraging its parallel computation framework and lightweight processing cores. Such high throughput is essential for workloads that involve repetitive, memory-bound operations, such as matrix computations and deep learning model training. The scalability of the system, enabled by the dynamic workload balancer, ensures that performance remains robust under increasing computational demands. Overall, the proposed NMP architecture demonstrates its capability to revolutionize data-intensive computing by addressing the inherent limitations of traditional and existing NMP systems. By rethinking the processing-memory relationship and introducing advanced features like in-situ computation and dynamic resource management, the architecture paves the way for substantial advancements in performance, energy efficiency, and scalability. These findings position the proposed system as a viable solution for modern computing domains such as AI, scientific research, and big data analytics, where efficiency and performance are paramount. Future work can build upon this foundation to further optimize the system for specific applications and explore its integration into existing computing infrastructures.

5. Conclusion

This study presents a high-performance Near-Memory Processing (NMP) architecture designed to address the growing computational demands of data-intensive applications. Through detailed evaluation, the proposed architecture has demonstrated significant advancements in key performance metrics, including latency reduction, energy efficiency, and throughput, compared to traditional computing systems and existing NMP designs. The results reveal that the proposed NMP system achieves an average latency improvement of over 40% across benchmarks. Specifically, latency for matrix multiplication reduced from 120.5 ms in traditional systems to 70.4 ms, while convolutional neural networks (CNNs) experienced a reduction from 95.7 ms to 55.2 ms. Similarly, data analytics workloads showed a reduction from 105.3 ms to 62.8 ms. These findings highlight the effectiveness of in-situ computation and minimized data movement in enhancing computational speed.

In terms of energy efficiency, the proposed architecture achieved gains of nearly 50% across all benchmarks. For instance, energy consumption for matrix multiplication decreased from 25.6 J in traditional systems to 12.8 J, while CNNs reduced from 19.3 J to 10.4 J. Data analytics workloads showed a decrease from 21.8 J to 11.9 J. These results validate the architecture's ability to significantly lower power demands, making it suitable for power-constrained environments. The proposed NMP system also demonstrated remarkable improvements in throughput, with gains nearing 100% in some benchmarks. The throughput for matrix multiplication increased from 45.3 GOPS in traditional systems to 90.2 GOPS, while CNNs saw an increase from 52.7 GOPS to 102.3 GOPS. Similarly, data analytics workloads showed an improvement from 48.1 GOPS to 96.5 GOPS. These results emphasize the architecture's capacity to handle concurrent tasks efficiently, ensuring high scalability and performance for modern computing workloads. In conclusion, the proposed NMP architecture effectively overcomes the limitations of traditional and existing systems by integrating processing units within memory arrays, optimizing memory hierarchy, and incorporating parallel computation frameworks. These advancements make it a robust solution for diverse applications, including artificial intelligence, big data analytics, and scientific computing. The significant improvements in latency, energy efficiency, and throughput position this architecture as a key enabler for next-generation high-performance computing systems. Future work could further enhance this architecture's adaptability to specific applications and explore its integration into existing computing ecosystems

References

[1] Gathu, S., 2024. High-Performance Computing and Big Data: Emerging Trends in Advanced Computing Systems for Data-Intensive Applications. Journal of Advanced Computing Systems, 4(8), pp.22-35.

[2] Wulf, W.A. and McKee, S.A., 1995. Hitting the memory wall: Implications of the obvious. ACM SIGARCH computer architecture news, 23(1), pp.20-24.

[3] Hur, R.B. and Kvatinsky, S., 2016, July. Memory processing unit for in-memory processing. In 2016 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH) (pp. 171-172). IEEE.

[4] Leiserson, C.E., Thompson, N.C., Emer, J.S., Kuszmaul, B.C., Lampson, B.W., Sanchez, D. and Schardl, T.B., 2020. There's plenty of room at the Top: What will drive computer performance after Moore's law?. Science, 368(6495), p.eaam9744.

[5] Falsafi, B., Stan, M., Skadron, K., Jayasena, N., Chen, Y., Tao, J., Nair, R., Moreno, J., Muralimanohar, N., Sankaralingam, K. and Estan, C., 2016. Near-memory data services. IEEE Micro, 36(1), pp.6-13.

[6] Iskandar, V., Ghany, M.A.A.E. and Goehringer, D., 2022. Near-memory computing on fpgas with 3d-stacked memories: Applications, architectures, and optimizations. ACM Transactions on Reconfigurable Technology and Systems, 16(1), pp.1-32.

[7] Muralidhar, R., Borovica-Gajic, R. and Buyya, R., 2022. Energy efficient computing systems: Architectures, abstractions and modeling to techniques and standards. ACM Computing Surveys (CSUR), 54(11s), pp.1-37.

[8] Murthy, P. and BOBBA, S., 2021. AI-Powered Predictive Scaling in Cloud Computing: Enhancing Efficiency through Real-Time Workload Forecasting. Iconic Research And Engineering Journals, 5(4), pp.143-152.

[9] Zhang, C., Sun, H., Li, S., Wang, Y., Chen, H. and Liu, H., 2023. A survey of memory-centric energy efficient computer architecture. IEEE Transactions on Parallel and Distributed Systems.

[10] Herruzo, J.M., Fernandez, I., González-Navarro, S. and Plata, O., 2021. Enabling fast and energy-efficient FM-index exact matching using processing-near-memory. The Journal of Supercomputing, 77(9), pp.10226-10251.

[11] Beloglazov, A., Abawajy, J. and Buyya, R., 2012. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future generation computer systems, 28(5), pp.755-768.

[12] Ahn, J., Yoo, S., Mutlu, O. and Choi, K., 2015. PIM-enabled instructions: A low-overhead, locality-aware processing-in-memory architecture. ACM SIGARCH Computer Architecture News, 43(3S), pp.336-348.

[13] Ceruzzi, P.E., 2003. A history of modern computing. MIT press.

[14] Fromm, R., 1999. Vector IRAM memory performance for image access patterns (Master's thesis, University of California, Berkeley).

[15] Poremba, M., Akgun, I., Yin, J., Kayiran, O., Xie, Y. and Loh, G.H., 2017. There and back again: Optimizing the interconnect in networks of memory cubes. ACM SIGARCH Computer Architecture News, 45(2), pp.678-690.

[16] Khan, K., Pasricha, S. and Kim, R.G., 2020. A survey of resource management for processing-in-memory and near-memory processing architectures. Journal of Low Power Electronics and Applications, 10(4), p.30.

[17] Jang, J.H., Shin, J., Park, J.T., Hwang, I.S. and Kim, H., 2023. In-depth survey of processing-in-memory architectures for deep neural networks. JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE, 23(5), pp.322-339.

[18] Doumet, M., 2024. High Performance CNN Inference on FPGAs with High-Bandwidth Memory (Master's thesis, University of Toronto (Canada)).

[19] Singh, G., Chelini, L., Corda, S., Awan, A.J., Stuijk, S., Jordans, R., Corporaal, H. and Boonstra, A.J., 2018, August. A review of near-memory computing architectures: Opportunities and challenges. In 2018 21st Euromicro Conference on Digital System Design (DSD) (pp. 608-617). IEEE.

[20] Kaxiras, S. and Martonosi, M., 2008. Computer architecture techniques for power-efficiency. Morgan & Claypool Publishers.

[21] Rai, S., Liu, M., Gebregiorgis, A., Bhattacharjee, D., Chakrabarty, K., Hamdioui, S., Chattopadhyay, A., Trommer, J. and Kumar, A., 2021, February. Perspectives on emerging computation-in-memory paradigms. In 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE) (pp. 1925-1934). IEEE.

[22] Khoram, S., Zha, Y., Zhang, J. and Li, J., 2017, March. Challenges and opportunities: From near-memory computing to in-memory computing. In Proceedings of the 2017 ACM on International Symposium on Physical Design (pp. 43-46).

[23] Yu, S. and Chen, P.Y., 2016. Emerging memory technologies: Recent trends and prospects. IEEE Solid-State Circuits Magazine, 8(2), pp.43-56.

[24] Chen, A., 2016. A review of emerging non-volatile memory (NVM) technologies and applications. Solid-State Electronics, 125, pp.25-38.

[25] Zhou, Z., Li, C., Wei, X. and Sun, G., 2021. GCNear: A hybrid architecture for efficient GCN training with near-memory processing. arXiv preprint arXiv:2111.00680, pp.1-15.

[26] Maity, S., Goel, M. and Ghose, M., 2024. CoaT: Compiler-assisted Two-Stage Offloading Approach for Data-Intensive Applications Under NMP Framework. IEEE Transactions on Emerging Topics in Computing.

[27] Li, Y., Tian, B. and Gao, M., 2024, October. Trimma: Trimming Metadata Storage and Latency for Hybrid Memory Systems. In Proceedings of the 2024 International Conference on Parallel Architectures and Compilation Techniques (pp. 108-120).

[28] Jezghani, A., Young, J., Powell, W., Rahaman, R. and Coulter, J.E., 2023, May. Future Computing with the Rogues Gallery. In 2023 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW) (pp. 262-269). IEEE.