Performance Analysis of Existing Storage and Processing Systems (Survey Paper)

Sushama Shirke, Anurag Chaudhary, Shubham, Aman Sharma, Puneeth Varma
[1,2,3,4,5]Dept. of computer engineering Army Institute of Technology Pune, India.

Abstract:
Enhancing the effectiveness and scalability of present-day computer systems requires performance study and enhancement of existing storage and processing hardware, soft- ware and their connections between them in order to effectively utilize them. The performance analysis methodologies, strategies, and case studies as they pertain to storage and processing devices are covered in great detail in this survey report. The objective is to comprehend the variables affecting system performance, detect performance bottlenecks, and suggest efficient optimization tech- niques. An overview of the significance of performance analysis in the present-day technological landscape introduces the paper. It draws attention to how quickly data must be processed, how much data must be stored, and how effectively resources must be used. The necessity to overcome these obstacles and realize the full potential of storage and processing devices served as the impetus for this survey. In-depth analysis of the various performance measures for assessing storage and processing systems is provided in the survey report. IOPS (Input/output Operations Per Second) and other important metrics are explained in detail. The importance of workload characterization and comparison as important methods for performance analysis are also covered in the paper. Bench-marking enables systematic comparison and as- sessment of various devices or configurations, whereas workload characterization includes understanding the nature of the jobs and data patterns that the system processes. The performance analysis techniques are carefully analyzed. Simulators are one of these approaches; they offer a controlled and virtual environment for assessing system performance. Researchers may simulate various workloads and evaluate the behavior of storage and processing devices under different situations using simulators like MQ-Sim and Gem5. The significance of precise as well as realistic simulations is emphasized in the paper in order to produce outcomes in performance analysis that can be trusted. The study also includes case studies that highlight how performance analysis methods are applied in real-world situations. These case studies address a variety of topics, including machine learning, scientific computing, data analytics, and database management systems. The issue statement, the approach used, and the out- comes are all thoroughly examined in each case study. For system architects and developers looking to optimize storage and processing devices in their respective sectors, these insights provide invaluable lessons. In conclusion, this survey report serves as a comprehensive guide to the field of performance analysis and improvement for existing storage and processing

devices. By exploring various performance metrics, evaluation methodologies, and real-world case studies, the report provides a deep understanding of the challenges and opportunities in optimizing system performance. The findings presented in this report will be valuable to researchers, practitioners, and industry professionals seeking to enhance the performance of storage and processing devices in diverse application domains.

DOI: 10.24297/j.cims.2023.6.13

1. Introduction

Overview

In order to guarantee effective and optimized operations across a variety of sectors, from data-centers and cloud com- puting to edge computing and AI applications, performance analysis and upgrading of current storage and processing systems are crucial. It is increasingly important to evaluate the needs and difficulties of these systems and come up with efficient techniques for boosting their performance as organizations struggle to manage escalating data volumes and heavy workloads. Understanding of the precise needs and objectives of performance analysis and improvement activities is the purpose of requirement analysis. It involves perform- ing a thorough analysis of the existing systems, workload characteristics, scalability, fault tolerance, energy efficiency, integration, and cost concerns. An extensive study can provide organizations with useful information about the shortcomings and possible areas for improvement in their storage and processing systems. This introduction provides an overview of the requirement analysis process for the performance analysis and improvement of existing storage and processing systems. It highlights the importance of understanding performance metrics, workload characteristics, system architecture, scala- bility, fault tolerance, energy efficiency, integration, and cost factors.

B.Various common ways to evaluate the performance of storage and processing devices are as follows:

- Performance Metrics: The performance metrics that will be applied to evaluate the performance of storage and pro- cessing systems are one of the main areas of attention for requirement analysis. Included in these measurements are things like reaction time, throughput, latency, data trans- fer rate, scalability, and energy efficiency. Organizations can accurately measure and evaluate the effectiveness of various systems and identify areas in need of development by explicitly defining these metrics.

- Workload Analysis: For the purpose of evaluating sys- tem performance, it is essential to fully understand the characteristics of the workload being managed. In order to do this, it is necessary to analyze the different apps and tasks being used, the data access patterns, and the distribution of read-and-write activities. Organizations can customize their storage and processing systems by analyzing the workload to find any unique requirements or restrictions. Dept. of Computer Engineering 2022-23 14 Performance Analysis and Improvement of Existing Storage and Processing Systems.
- System Architecture Analysis: For the purpose of find- ing bottlenecks or other restrictions that could affect performance, it is essential to analyze the architecture of the current system. The hardware analysis procedure takes into account the network infrastructure, memory, processors (CPU, GPU), storage devices (HDD, SSD), and processors (CPU, GPU). To make sure they are optimized for fast access to and processing of data, the software stack and data management strategies should also be evaluated.
- Scalability and Capacity Planning: Analyzing require- ments also includes determining the storage and process- ing systems’ capacity needs and scalability requirements. The analysis of the projected increases in data volume, user base, and workload requirements is all part of this. It is possible to handle increasing demands without compromising performance by being aware of the scalability constraints of the present systems and planning for the needs of the future.
- Integration and Interoperability: Analyzing the inte- gration needs and feasibility of the storage and processing systems with the current infrastructure and applications is a part of requirement analysis. This involves determining how well the APIs, data formats, and software programs function together. Reducing interruptions and enhancing system performance can be accomplished by ensuring interoperability and seamless integration.
- Cost Analysis: Making decisions requires awareness of how performance improvements may affect costs. This involves evaluating the cost-effectiveness of various improvement techniques while taking into account things like infrastructure expenditures, maintenance expenses, and hardware/software updates. Setting aside activities according to their future research. Impact is aided by evaluating the return on investment (ROI) of performance improvement efforts.

Requirement analysis is a crucial step in the performance analysis and improvement of existing storage and processing systems. It involves understanding performance metrics, work- load characteristics, system architecture, and scalability.

The rapid growth of data-intensive applications has led to an increased demand for high-performance storage solutions. Traditional solid-state drives (SSDs) have proven to be effi- cient in terms of random access and latency, but they still face limitations when it comes to processing large volumes of data. To address these challenges, researchers have introduced a new approach called Near-Data Processing (NDP) for example papers [1], [2], etc uses to improve the performance, which leverages the computational capabilities of storage devices to perform data processing tasks. This survey report aims to provide a comprehensive overview of NDP-based SSDs, their benefits, challenges, and future prospects. The idea of decreasing data travel and bringing processing closer to the data is referred to as NDP. Parallel data processing, lower latency, and increased system efficiency are all made possible by NDP-based SSDs since they provide computing power to the storage device. This paradigm change offers a more balanced approach to processing and storage and creates new opportunities for data-centric applications which is mention in [13], [20] and [10] uses SQL database for SaS operations.

2. Literature Survey

A. NDP based SSDS

The number and variety of data produced by various ap- plications, ranging from social media platforms to scientific simulations, have significantly increased in recent years. Due to the data’ s exponential increase, there is now a need for high-performance storage solutions that are more effective. Hard disk drives (HDDs), a type of conventional storage device, have a difficult time keeping up with the demands of contemporary data-intensive applications. Since SSDs offer considerable performance gains over HDDs, this has led to the investigation of alternate storage technologies. SSDs are non-volatile storage devices that store information using flash memory technology. Compared to HDDs, they offer quicker data access, reduced latency, and better I/O throughput. Due to these benefits, SSDs are a common choice for servers, workstations, laptops, and mobile devices, among other com- puter applications. The design and operation of SSDs still have room for development, despite their natural performance advantages. Applying Near-Data Processing (NDP) methods is one possible strategy for enhancing SSD performance. Data processing close to the storage devices rather than solely relying on the host system’ s central processing unit (CPU) is the idea behind NDP. By moving processing operations closer to the data, NDP may significantly reduce data transport and communication overhead, improving system performance as a whole. In-device computations like data filtering, compression, encryption, and indexing are carried out by NDP-based SSDs by utilizing the parallel processing capabilities of current stor- age systems. Throughput is increased and latency is reduced by doing away with the need for data transfers between the storage device and the host system. Additionally, NDP-based SSDs can free up significant processing resources for other crucial operations by offloading computation-intensive tasks from the CPU. When integrating NDP techniques into SSDs, there are a number of factors that must be carefully considered, including the workload description, data management, and system design. The workload description requires research into the characteristics of the tasks and data patterns that the system encounters. In The types of computing tasks that NDP might be useful for must be understood in order to design and op- timize NDP-capable SSDs. a management is crucial in NDP- based SSDs because it necessitates efficient data placement and organization within the storage device. Another aspect of system architecture to consider is the design of hardware and software components that enable the seamless integration of NDP capabilities into SSDs. Techniques like data partitioning, data replication, and caching are used to maximize data access and processing. Another aspect of system architecture is to consider the hardware and software components that enable the seamless integration of NDP capabilities into SSDs. Here, optimized CPUs, memory configurations, and software frameworks are used to facilitate efficient data processing and management inside the storage device. This survey report’ s goal is to give a thorough review of NDP-based SSDs with a special emphasis on performance analysis and improvement. The study examines the body of knowledge, academic publica- tions, and commercial developments in this area, highlighting the major discoveries, approaches, and difficulties. This report aims to provide useful information for researchers, practi- tioners, and industry professionals interested in optimising storage systems for data-intensive applications by analysing the performance metrics, evaluating the experimental results, and discussing the implications of NDP-based SSDs.

B. Near Data Processing based SSDs (Background)

In [1], addition to I/O requests (like read and write), NDP- based SSDs can handle NDP requests like object recognition, data from intelligent and hybrid systems, and accelerated devices. NDP requests and regular I/O requests on NDP- based SSDs will compete with limited processing units such as controllers, NAND flash, onboard processors, and accelerators. Therefore, it is necessary to schedule mixed NDP-based regu- lar I/O requests to take advantage of NDP-based SSDs. How- ever, most previous work has focused on developing NDP sys- tems for specific applications, such as relational data and maps, while neglecting the term mixed NDP-normal I/O requests. In general [1], current NDP-based SSDs only handle I/O requests on a first-come, first-served (FCFS) principle, which can cause serious damage to processing facilities, especially for NDP jobs. RecSSD in [4], uses beacons to ensure that NDP opera- tions always take priority over normal I/O requests, which is an NDP priority strategy. In this scenario, while the average I/O execution time will continue to rise, the overall execution time of NDP requests can be improved. In order to ensure equitable resource distribution and decrease resource hunger, FusionFS in [15], uses fair planning (CFS) to schedule the execution time of the configuration process and the location of DRAM on the SSD. However, it does not take performance as an optimization target and does not take into account the priority of other units, such as accelerators and flash memory chips. SSD timing algorithms like DBRS and FLIN focus on the timing of the flash chip rather than the overall operation, which may be impacted by dependency level. Thus, in order to achieve good performance, it is necessary to take into account all the challenges associated with handling NDP and standard I/O requests. In this essay, we suggest Horae in [1], a hybrid I/O request programming protocol. In order to understand the hybrid NDP-normal I/O request planning problem (HISP) in NDP-based SSDs, we must first understand that each request is a multi-stage task created using the acyclic graph (DAG) and NDP representation. Different types of connection technology are used for SSD-based SSDs. HISP is NP-hard because it is a combination of machine configuration problems and function store configuration problems. On the net, we create a two-stage workflow to optimize the overall processing time of the composite request. In the first stage, Horae identifies the importance of each workplace stage by identifying the main ways to compete in the current business. In the second stage, Horae quickly records the spatiotemporal position of the work according to the importance of the work and the work done. Horae can use the parallel phase better than the FCFS algorithm, NDP-first strategy, DBRS, and FLIN in [15] because Horae always focuses on the main path of the most competitive class. We evaluated the Horae using RecSSD’ s NDP request-written work items, Slacker’ s traditional I/O, and a self-developed NDP system for writing. Experimental results show that Horae can shorten and improve the use of resources and ratios according to different tasks. Horae can reach 23. Compared to the NDP’ s first strategy used by RecSSD in [4], average performance increased by 4%. Compared to FCFS, Horae can achieve a reduction of 11.7% and a maximum reduction of 33% as a combination of operations, while improving utilization efficiency by 6.8% and the average rate by 5%.

The objective is to set up a computing workbench that can handle CPU and GPU-based processing frameworks. The workbench includes different memory capacity devices such as SSD (Solid State Drive) and HDD (Hard Disk Drive). The researchers use various workload files for their analysis. They run two algorithms, namely FCFS (First-come, First- Served) and Horae, on these workload files. The algorithms are executed separately on the CPU and GPU, as well as simultaneously on both SSD and HDD. This experiment aims to evaluate and compare the performance of different combina- tions of processing units (CPU and GPU) and storage devices (SSD and HDD). We are measuring metrics such as execution time, resource utilization, or any other relevant performance indicators to analyze the efficiency and effectiveness of these storage and processing systems.

C. Performance Metrics

Since, availability of data everywhere each electronic de- vices is itself a source enormous data. With the advancement of technologies in storing the over period of time now come the problem of Data processing. over time various techniques have introduced to process the data in memory, processor etc. to improve the overall performance, increase the effectiveness in terms of time and space complexity, decrease the latency input

– output request, increase the throughput and IOPS measure. In these paper we will mainly focuses on latency, throughput and some IOPS etc.

- Latency it is total time taken by computer system to respond the given request for action. Latency can be depends on various factors such as disk access times, processing delays and memory access time etc. It means that if request is processed faster then the user experience is faster.
- Throughput it is defined as rate of total amount workdone by computer system. It generally measure in transfer rate of instructions, data etc, and Number of instructions exe- cuted per second, and other similar metrics. A computer that have better throughput they can manage balance between software and hardware.
- IOPS stand for Inputs/Outputs per second. It is define number of inputs/Outputs that a computer system can perform in one second. IOPS can be used as performance metric such as Storage devices SSDs (Solid State Drives), HDDs (Hard Disk Drives) and SANs (Storage Area Networks).

To implement NDP-based SSD scheduling techniques and analyze the performance and verification. We mostly deal with the memory (HDD, SSD, or both) and processor (CPU, GPU, or other) input and output requests per second (IOPS) flow. Based on this, we will check the performance of the data flow.

D. Latency

[1] proposes a completely new hybrid idea for the NDP based SSDs. It proposes a hybrid Input/Output request scheduling techniques different from traditional SSDs for Near Data Processing (NDP)-based Solid-State Drives (SSDs). The techniques proposed by [1] reduces both latency and total energy consumption. This done by using a smart hybrid scheduling algorithms which optimize the I/O requests be- tween the processor and memory. It is present at the interface between the host and NDP-based SSDs. Horae in [1], takes the advantages of low latency and high bandwidths communica- tion capabilities of I/O requests data flow in NDP architectures to perform the firstly scheduling of Normal I/O or Hybrid I/O requests to process the certain request (which is Data) tasks directly near to memory which have small processors driven by processing steps algorithms to preprocess the data. Insort this makes the utilization of independent data and parallelism etc. However this reduces the amount of data that need to be transmitted to and fro from the memory to processor or vice versa also improve the overall performance of the system.

[1] proposed idea uses only CPUs as for processor (Means no GPU or any other systems) and in terms of memory utilization it uses SSDs not even HDDs. Further more, [1] uses hybrid scheduling process that is combination of both batch and deadline - based scheduling to optimize the processing and reduce latency. [2] don’t directly mention about latency but uses vectorization in PIM (Processing in Memory) which reduces the latency significantly.

In [2] uses PIM in which data is directly processed in memory, reduces the movement of data between processor and memory (or cache memory etc.) which reduces latency and improved performance. Paper [2] proposed uses of dynamic vectorization techniques to improve performance, dynamic vectorization allows to adjust size of vector while executing the instructions on the available of resources, which help in reducing the latency associated with processing the large amounts of data generated from electronic devices or etc. As we moves from low processor to high processor computation power of systems increases TPU or GPU have have better performance than CPU and both used together as vectorization involve parallelism and multi processor system increase speed and performance. In terms of memory SSDs are faster than traditional HDDs, since [2] explicitly didn’t talk about memory but SSDs can perform better for vectorization. [3] presents approach to improving performance in PIM architectures by automatically converting instructions set (Data instruction) into physical circuit (means Hareware) for execution near or inthe memory.

Paper [3] didn’t mention about the types of processor used. However the proposed techniques for convert instruction in to hardware circuits GPU is advantageous than CPU as GPUs can used in parallel processing, integrating processing elements within the memory hierarchy of GPUs. Thus it reduces the data movement between memory and processor hence latency decreases. Both HDDs and SSDs have different access time, transfer rates and other performances characteristics that can induces latency. [3] doesn’t discuss types of memory used. Paper [4] discusses on NDP technique for recommendation inference using SSDs which reduces the amount of data transferred between storage devices and processor thus reduces the latency of overall system. Any processor can be used including the CPUs, GPUs or Others.

[4] primarily focuses on performance improving of using NDP in SSD based recommendation inference systems. [5] proposed cognitive SSD architecture help in improving per- formance and latency compared to traditional SSDs. It also states that deep learning can help in reducing data movement.

[5] is independent of processor, very small effect of types of processor used whether CPU or GPU.proposes architectures for large scale AI application (which require lot of data handle and manages) to have better and fast storage computing power. [6] architectures includes enhancement in storage and computing parts to improve the performance for large AI or Data Driven applications. [6] proposes a new architecture that combines or integrates storage and computing into one single unit system which enables improved performance and reduced energy consumption. [6] mentions SSDs for storage and GPUs for processor computing can accelerates the AI workloads in data processing tasks. However HDD or CPU in permutation and combination can also be used with result produced with reduced performance and Latency. Experimentally proven that [6] proposed archi- tecture efficient in terms of cost, energy and scalable also.

Just like other NDP architectures paper [7] is also designed and build to reduce the latency and to increase the throughput by processing the data near to storage as much as possible. From above discussed paper [8] proposed new idea for design- ing and implementing flash storage systems. The framework deigned in [8] consists of hardware modules, firmware mod- ules, and software modules. Using all these [8] implemented several flash storage systems including NVMe-oF (NVMe over Fabrics) storage system. [8] don’t directly mention about specification of processor types for latency performance. But it discussed use specialized hardware accelerators such as FPGAs and GPUs, to maximize the performance of certain program or tasks such as compression and encryption, in the flash memory. The accelerators can reduce the latency of overall systems instead of realying on CPU for processing.

[8] solely uses SSDs so discussion on other memory such as HDDs.

[9] is simulator that uses SSD as the storage devices and CPU as the processor by using sufficient algorithms can be used to prove effect of latency on overall systems which have virtual hardware and software. [10] uses SQL workloads whereas other uses data or instruction workload to process. [10] don’t compare the performance of SSDs to HDDs or CPUs to GPUs. Primary goals is to optimize the database designing to take advantage low latency. [10] proposes distributed storage engine and memory manage- ment optimizations, to improve performance. Sas paper [10] out performs traditional highlights benchmarks with proper utilizations of SSDs potential for Database workloads and applications. Paper [11] compares the two configurable NDP servers, then it evaluates it behaviour and performance with traditional server without NDP in it. The results comes out in favour of NDP servers, it reduces latency sufficiently in the workloads were high and heavy processing is required such as Machine Learning and graph processing. However the latency paper mostly relies on characteristics of the workload and specificity or specialization of NDP servers architectures. It don’t discussed about memory whereas in terms of processor with mutli-processing is better to handle high and heavy workloads. But it compared to paper to only CPU based processors.

[12] proposes blockNDP (which combines traditional block storage in the memory for the processing of data closer to stor- age location) it uses a distributed key-value store and compares the performance using variety of benchmarks. BlockNDP reduces latency as compare traditional block storage. [13] argues that traditional storage devices such as SDDs, HDDs, may not able to keep the increasing pace for storage and performance demands. It suggest concepts of programmable solid state (PSS), which can be defined as a storage devices that can be programmed to perform custom data processing as data in being stored and retrieved from

memory. PSS is deigned in such way to be flexible and programmable to cope with demand of performance and storage. PSS can improve the performance, lower power consumption and reduced latency. However this architecture is for the cloud datacenters which reduces the latency and faster access times especially for the workloads including the large data. [14] discuss more on applications of NDP, how NDP can help in improving of various applications which includes Database processing, data analytics, machine learning and scientific computations. It also compares various other architectures PIM, SCM (Stor- age Class Memory) etc. By reducing the movement of data movement overheads and latency, it highlights its potentials advantages. [15] overcome the issue faced by NVMe SSDs, where some I/O requests may experience long delays due to interference from concurrent requests. It proposes FLIN a solution to do resolved it. FLIN is based on mechanism that select priorities of a request based on the requirement and dynamically it arranges the number of outstanding requests per queue. It showed that FLIN output can reduce the average tail latency by up to 76% compared to existing solutions. In [17] author suggest that parallelism inside can be exploited at multiple level, including parallelism between multiple chips, or with each individual NAND flash chip. To achieve this it uses multiple techniques including mapping schemes that balance the workload across multiple chips, a parallel garbage collection mechanism that allow multiple NAND flash chips to perform garabage collection in parallel etc. However it claims that it improves performance and latency by upto 40%.

In [17] consists of a platform programmable SSD con- troller and set of processing core that can be used aas for executing the set of programs inside the SSD. It consists of detailed information about the REGISTOR using various different workloads including data compression, encryption, pattern matching etc which effectively proved that it reduces data movement and improves processing performance and latency. In [18] proposes a hardware accelerator for regular string matching is called REACT. The designed presented the accelerator and storage devices are part of each other as one unit, that enables in-storage processing of regular string experssions, which reduces data movement and improves overall performance hence it can also reduce latency.

Paper [19] solve the issue in existing SSD’ s scheduling algorithms which arranges request based on the completion time. As it lead to low efficiency and performance dur to there large variation in processing time of each requests. It proposed algorithm called as Delay-based I/O request scheduling (DIOS) that is combination of number of pending requests and processing time of each request. Then algorithms schedules requests based on their delays values, higher delay time given higher priority. Time taken between memory and processor reduces as results latency also reduces.

The paper [20] proposed a system that takes advantages of capabilities of existing modern storage devices such as SSDs etc, to execute tasks such as Data compression and duplication etc inside the memory. INSIDER involves a mod- ified hardware-software designed approach, a combination of SSD and lightweight processor controller runtime system to enables sufficient and flexible in-storage computation. Thus it reduces data movement and latency significantly. This surveys paper [21] provides depth analysis of various architectures including in-memory computing, processing in-memory, near- memory computing and near-storage computing. As it show NDP reduces the Data movement and reduces the latency using neural networks.

E. Throughput

The paper [1] mostly focuses on to reduce the latency and energy consumption, but it also discuss the potential and importance to improve the throughput of systems in proposed techniques. [1] uses scheduling techniques which schedules to kinds of request first normal request and other is hybrid request which to processed near the memory. [1] proposed idea uses only CPUs as for processor (Means no GPU or any other sys- tems) and in terms of memory utilization it uses SSDs not even HDDs. As results the movement of data while processceing in CPU data movement from processor and memory is reduced which improving the overall performance of the system. By utilizing I/O request for scheduling and taking advantage of uniques features of NDP based SSD architectures it reduces the amount of data needs to be transmited, techniques in [1] can increase the throughput sufficiently to help in performance. Primary focus of [2] was on vectorization of PIM pro- cessing depends on ability to handle multiple thing paral- lel. GPU are designed to do multiprocessing so throughput in processing is advantageous but depend on specific other factors also. In terms storage memory for paper [2] SSDs have higher throughput as compare to HDDs as they have faster Read/Write speeds. This can be restrictions as certain cases requires high throughput. However, the performance of [2] depends on specific architecture and implementation. [3] explicitly never discussed about the memory since SSDs have throughput rate as compare to the HDDs due to their fast access time period etc. aslo not much about GPU or CPU or types of processor used but multi-programming is better than single program in computers. [4] it mainly uses SSDs memory in NDP architecture which can improve throughput and performance of recommendation inference tasks. Results of [4] showed output as 50x higher throughput and 5x lower la- tency than the traditional or existing CPUs based solutions for recommendations inference. As [4] don’t mention that GPU as the processor will outperform the CPU as the processor in recommendations inference. [5] uses Conginitive SSDs and Deep Learning techniques to increases the throughput using SSD whereas it doesn’t tell details about HDD. In order to perform NDP in [5] it uses deep learning techniques to make the decision so that movement between storage devices and processor is reduced. In [5] throughput don’t changes much on types of processor CPU or GPU. [6] don’t clearly discussed specific processor while comparing throughput. Moreover it focuses on design and architectures of storages computing which will benefits NDP to reduce data movement and increase energy efficiency.

[6]proposes SSDs as memory and FPGA as accelerators as processor for storage computing system to perform NDP, which leads to increase energy efficiency, better performance of AI workloads.

[7]performance wise is same as of [6], but [7] didn’t give specific details about the storage or memory devices such as HDDs and SSDs or processor or computing systems such as CPU and GPUs etc mostly because it is a frameworks it is suitable on maximum platforms with minimum requirement to run. [8] don’t rewill about types of processors CPU or GPU comparison for NDP. However, [8] the systems use of multi-core CPUs as the operating unit for SSDs and the CPU is responsible for operating systems function or activity such as managing I/Os request, data placements and garbage collections operations. Further more, [8] can support multi SSD using PCIe switch which will increase overall system throughput. Measurement of throughput by [9] is cost effective only very less systems required and to understand different scenario. [10] have high throughput with highest performance achievement in its fields with proper utlization of distributed storage and memory management for databases workloads and applications.

[11] shows NDP servers improves the throughput by reduc- ing data movement and improving data locality. Improvement in throughput depends on architectures of NDP used. It shows the improvement in throughput of 2.5x to 5.5x for TPC-H (Stands for Transaction Processing Council Ad-hoc/decision support benchmark) using NDP-FPGA, and performance im- provement of 1.5x to 3.0x using NDP-CPU. [13] proposed systems have the capability to provide higher throughput as compare to traditional storage systems, because it can perform data processing directly on storage itself. Which reduces the transfer of data between the storage device and the processor. This output results occurs in higher throughput and faster data access time, more importantly

for workloads have large data. Types or choice of processor may effect PSS’ s performance and throughput, but it mostly depend on character and be- haviour of workloads and their actions to be performed. As CPU is better situated for handling general purpose workload, while GPU can handle mulit or heavy parallel workloads used in Machine Learning and Higher computations. [14] also reduces throughput over system by reducing data movement.

[15] give focus on the importance of fairness issues on modern NVMe SSDs, which also improves the throughput by upto 30%.

[16] proposes varies techniques that increase its endurance and throughput by up to 35%. [17] also showed this techniques can lead to significant performance and energy consumption lead to higher throughput. REGISTER and programmable SSD, set of cores consists it main parts to experiment the results. The REACT accelerator in [18] paper includes its architectures, regular expression syntax, and performance eval- uation on a real storage devices. It performs better than software based string expression handlers, it also can handle large query or expression with results in high throughputs. The DIOS used in [19] is tested on SSD and compared with traditional scheduling algorithms which achieves higher I/O throughput and lower response time even with higher workloads with high variation in processing time of each re- quests. In [20], INSIDER experimental results shows sufficient improves performance and high throughput as compares to traditional systems. [21] discuss various emerging techniques to get high throughput such as RPUs (resistive processing dis- cussessing Units) and spintronic devices which can enhance NDP performance.

3. Summary

Unprocessed data is useless, to get process data in such time where every electronic devices is source of the data. So to process the data in computer architecture or data processing in some applications a special techniques is mentioned. Every paper uses some hardware, software or both to reduce the latency and increase the throughput. So that data movement between memory and the processor is reduced and IOPS between is also optimized.

All the above paper can be categorized in mainly in parts Hybrid request and heterogenous function unit under the three different region which are as follows Userspace, kernel, and NDP-based SSD. Somehow, other paper made changes or enhancements in the architecture of NDP-based SSDs to reduce the data movement. In userspace tells about types of workloads: normal I/O requests or hybrid requests, or databases, etc. kernel mainly includes drivers; in NDP based SSDs, its about embedded processors, NAND flash, and accelerators.

We observe that using a scheduling techniques to select normal I/O request or hybrid I/O request for processing, as it reduces the number of movement of data from memory and processor. A small processor near to memory is setuped to processed data which reduces the overall time, reduces the energy consumption. Several other techniques also used such as using enhanced SSDs, Deep learning techniques, various recommendation etc. Each some how or others reduces the movement data between memory and processor, hence reduces latency and increase throughput.

4. Conclusion

A NDP based SSD architecture exploits the data movement closer to storage devices, similar to architectures such as pro- cessing in-memory, in-storage processing. Which improves the performance by reducing the costly data movement between the processor and memory. It is feasible to analyzing and improving the performance of such processor which reduces latency and produced high throughput. The bottleneck problem between processor and memory on number I/O request per second for processing can also be handled. Using simulator a NDP based SSD can implemented be implemented to show proof our results.

In conclusion, one of the most essential elements of im- proving system efficiency and boosting overall performance is the performance analysis and enhancement of current storage and processing systems utilizing simulators like MQ-Sim and Gem5. We may learn more about how different workloads behave on different processor and storage device combina- tions, such as CPU and GPU, through algorithm analysis, mathematical modeling, and careful design. We can simulate and measure the performance of various workload scenarios using simulators in a managed and realistic setting. This makes it possible for us to identify potential system problems, inefficiencies, and improvement opportunities. We might ex- plore the effects of changing workloads, static and dynamic behaviours, and various hardware configurations on system performance by applying these simulators. With the use of this information, we are better able to plan workloads, allocate resources, and optimise hardware and software. Preparing the workload data and creating measures of performance for analysis are both part of the pre-processing step. To do this, typical workloads must be determined, data traces must be collected, and performance measures like latency, throughput, and energy efficiency must be defined. We create models and algorithms that evaluate and predict system performance throughout the algorithm analysis and mathematical modelling phase. In order to understand system behaviour, this involves taking into account variables including data access patterns, resource use, queuing theory, and statistical analysis. The development of an advanced architecture and a strategy for execution for improving performance are both included in the detailed design phase. It entails taking into account vari- ables such as workload categorization, system setups, caching methods, request scheduling strategies, and data management tactics. The performance analysis and improvement project' s unique needs and objectives will determine the models that are working. Queuing models, statistical models, machine learning algorithms, and optimization methods are a few examples. In general, using simulators like MQ-Sim and Gem5 offers a practical and economical solution to assess various workload situations and enhance current storage and processing sys- tems. We can improve system performance, increase resource utilization, and maximize energy efficiency by knowing how workloads behave on different hardware configurations. Imple- menting data-driven decisions, placing effective algorithms in place, and improving the system for greater performance are all possible with the help of algorithm analysis, mathematical modeling, and careful design. This helps to increase the overall performance of the current storage and processing systems, resulting in better user experiences, lower costs,and higher system effectiveness. In conclusion, performance analysis and improvement on simulators provide an effective method for exploring, analyzing, and optimizing the behavior of diverse workloads on various processor and storage device configurations. We can maximize the performance of current systems and promote performance improvements in the storage and processing domains through complete algorithm analysis, mathematical modeling, and meticulous design.

References

1. Duo liu, Lin Li, Xianzhang, Lei Qiao ETL, "Horae: A Hybrid I/O Request Scheduling Technique for Near-Data processing-Based SSD" , IEEE Transactions on Computer Aided design of Integrated Circuits and Systems, Vol. 41, NO. 11, November 2022
2. On the SPEC-CPU 2017 opportunities for dynamic vectorization possi- bilities on PIM architectures, Rodrigo M. Sokulski, Sairo R. dos Santos, Marco A. Z. Alves, Department of Exact Sciences and Information TechnologyFederal Rural University of the Semi-arid (UFERSA) -—Angicos, Brazil
3. Proposta de conversa˜o automa´tica em hardware de instruc¸o˜es para execuc¸a˜o em memo´ria, Rodrigo Machniewicz Sokulski , Marco Antonio Zanata Alves, Departamento de Informatica – Universidade Federal do Paran´a (UFPR) Caixa Postal 19.011 – 81.531-990 – Curitiba – PR –Brazil

4. Mr. Umakant Dinkar Butkar, Manisha J Waghmare. (2023). An Intelligent System Design for Emotion Recognition and Rectification Using Machine Learning. Computer Integrated Manufacturing Systems, 29(2), 32–42. Retrieved from http://cims-journal.com/index.php/CN/article/view/783 .
5. S. Liang, Y. Wang, Y. Lu, Z. Yang, H. Li, and X. Li, "Cognitive SSD: A deep learning engine for in-storage data retrieval," in Proc. USENIX Annu. Tech. Conf. (ATC), 2019, pp. 395–410
6. J. Y. Do et al., "Cost-effective, energy-efficient, and scalable storage computing for large-scale AI applications," ACM Trans. Storage, vol. 16, no. 4, pp. 1–37, 2020
7. Mr. Umakant Dinkar Butkar, Dr. Pradip Suresh Mane, Dr Kumar P K, Dr. Arun Saxena, Dr. Mangesh Salunke. (2023). Modelling and Simulation of symmetric planar manipulator Using Hybrid Integrated Manufacturing. Computer Integrated Manufacturing Systems, 29(1), 464–476. Retrieved from http://cims-journal.com/index.php/CN/article/view/771
8. J. Kwak, S. Lee, K. Park, J. Jeong, and Y. H. Song, "Cosmos+ OpenSSD: Rapid prototype for flash storage systems," ACM Trans. Storage, vol. 16, no. 3, pp. 1–35, 2020
9. A. Tavakkol, J. Go´mez-Luna, M. Sadrosadati, S. Ghose, and O. Mutlu, "MQSim: A framework for enabling realistic studies of modern mul- tiqueue SSD devices," in Proc. USENIX Conf. File Storage Technol. (FAST), 2018, pp. 49–66
10. H. Park, S. Choi, G. Oh, and S.-W. Lee, "SaS: SSD as SQL database system," Proc. VLDB Endownment, vol. 14, no. 9, pp. 1481–1488, 2021
11. D. R.-J. G.-J. Rydning, "The Digitization of the World From Edge to Core". Framingham, MA, USA: Int. Data Corp., 2018, p. 16
12. A. Barbalace, M. Decky, J. Picorel, and P. Bhatotia, "blockNDP: Block- storage near data processing," in Proc. 21st Int. Middlew. Conf. Ind.
13. Track, 2020, pp. 8–15
14. J. Do, S. Sengupta, and S. Swanson, "Programmable solid-state storage in future cloud datacenters," Commun. ACM, vol. 62, no. 6, pp. 54–62, 2019
15. P. C. Santos et al., "Survey on near-data processing: Applications and architectures," J. Integr. Circuits Syst., vol. 16, no. 2, pp. 1–17, 2021
16. A. Tavakkol et al., "FLIN: Enabling fairness and enhancing performance in modern NVMe solid state drives," in Proc. ACM/IEEE 45th Annu.
17. Int. Symp. Comput. Archit. (ISCA), 2018, pp. 397–410

18. Y. Hu, H. Jiang, D. Feng, L. Tian, H. Luo, and C. Ren, "Exploring and exploiting the multilevel parallelism inside SSDs for improved performance and endurance," IEEE Trans. Comput., vol. 62, no. 6, pp. 1141–1155, Jun. 2013
19. Umakant Dinkar Butkar, Dr. Nisarg Gandhewar. (2022). ALGORITHM DESIGN FOR ACCIDENT DETECTION USING THE INTERNET OF THINGS AND GPS MODULE. Journal of East China University of Science and Technology, 65(3), 821–831. Retrieved from http://hdlgdxxb.info/index.php/JE_CUST/article/view/313
20. W. S. Jeong et al., "REACT: Scalable and high-performance regular expression pattern matching accelerator for in-storage processing," IEEE Trans. Parallel Distrib. Syst., vol. 31, no. 5, pp. 1137–1151, May 2020
21. R. Chen, Q. Guan, C. Ma, and Z. Feng, "Delay-based I/O request scheduling in SSDs," J. Syst. Archit., vol. 98, pp. 434–442, Sep. 2019.Z. Ruan, T. He, and J. Cong, "INSIDER: Designing in-storage com- puting system for emerging high-performance drive," in Proc. USENIX Annu. Tech. Conf. (ATC), 2019, pp. 379–394.
22. Hassanpour, M.; Riera, M.; Gonza´lez, A. A Survey of Near-Data Processing Architectures for Neural Networks. Mach. Learn. Knowl. Extr. 2022, 4, 66–102. https://doi.org/ 10.3390/make4010004.