Research Statement

Janki Bhimani
bhimani@ece.neu.edu

We are fast approaching a new era of the *Data Age*. Big data storage management is gaining attention because of its vast storage requirements and exponential growth. International Data Corporation (IDC) forecasts that by 2025 the global datasphere will grow to 163 zettabytes [1]. From autonomous cars to intelligent personal assistants, the world around us is undergoing a fundamental change, transforming the way we live, work, and play. We as consumers will enjoy the benefits of a digital existence, powered by this wealth of data and the insight it provides. However, the current infrastructures of data, compute and storage will not be sufficient to meet the demand posed by these changes. Therefore, these trends present interesting challenges and opportunities towards designing holistic parallel and reliable computing systems for large-scale data-intensive workloads to efficiently store, process and manage many billions of bytes.

To seize these opportunities, in my research I have investigated interactions and implications of executing data-intensive workloads on emerging hardware systems to achieve better performance, endurance and reliability. In particular, I have built analytical models, prediction tools, and devised novel techniques to provide resource management for system computation, communication, memory and storage [2, 3, 4, 5, 6, 7, 8, 9, 10]. My research has proposed new ways to enhance the performance of big-data processing on large-scale enterprise cloud infrastructures [11, 12, 13, 14, 15, 16, 17, 13], and improve efficiency of scientific workloads on massively parallel high performance computing platforms [18, 19, 20]. My work has explored evolving flash technologies by demonstrating deployment to mitigate its proposed challenges and innovating techniques to provision endurance and reliability [21, 22, 23, 24, 25, 26]. My innovations have benefited many large-scale data processing and parallel high performance computing frameworks. My research also has significant impact outside my core technical field by helping to strengthen cyber-physical systems and Internet of Things (IoT). My research articles have currently above 280 citations, with h-index of 9, which shows the significant practical impact of my work. At the same time, my research has opened opportunities for solving myraid un-trodden challenges in evolving technological domains.

In the future, I plan to address full stack acceleration by resolving performance bottlenecks of various parallel and hierarchical system components. I will explore and design novel techniques to mitigate challenges of computing using virtualization frameworks on enterprise cloud, as well as to accelerate parallel processing using high performance platforms and accelerators such as GPUs. I will develop methods to dynamically manage performance and schedule appropriate resources in the systems with heterogeneous components. I will focus on leveraging productivity of data-intensive workloads by efficiently evolving emerging flash based technologies for better memory and storage devices. I will also work towards modeling emerging system technologies to enable to conduct in-depth modular analysis for developing useful insights towards building next generation devices. I plan to continue my research towards reconnoitering endurance and reliability to assess complex computing and storage infrastructures, and develop sustainable solutions. I want to start a new stream of research to securely manage storage capacity transactions using evolving techniques such as blockchain. I will achieve these goals by doing real system deployment and predictive analysis to effectively process, manage and store massive amount of data produced by different infrastructures. I believe that my research efforts have the potential to bring significant positive impacts to our society.

A strong collaboration with Samsung Semiconductors Inc. (Memory Solutions Lab and Memory Platforms Lab) was built through my three summer internships. This collaboration further helped our research group at Northeastern University to receive a $200K research grant from Samsung. Our research group also benefited by receiving extensive cutting-edge storage devices from Samsung Labs. I expect these collaborations to continue because the outcomes of my future research will directly impact systems at these labs. I was fortunate to gain valuable experience of writing successful proposals by helping my advisor to prepare and submit proposals to industries such as Samsung, Mathworks and IBM, and to other national grant agencies such as NSF and Air Force. I plan to take advantage of opportunities towards obtaining funding from national agencies such as NSF, DOE, Air Force, etc. I also believe that my strong collaborative relationship with industrial companies can further help me to obtain research funding.

Research Experience

The research I have done on flash-based storage and high performance computing platforms during my Ph.D study at NEU and internships in Samsung research lab stands in the center of current technology innovation trends. My research has focused on enhancing four important components of computer systems: Computation, Communication, Memory and Storage. My research contribution to date can be classified into the following broad areas:

- **Evolving Emerging Flash Technologies to Provision Endurance and Reliability [21, 23, 25, 15, 24]**

The demand for high speed 'Storage-as-a-Service' (SaaS) is increasing day-by-day. Solid State Drives (SSDs) are commonly used in higher tiers of storage rack in data centers. Also, all flash data centers are evolving to better serve cloud services. Although SSDs guaranty better performance when compared to Hard Disk Drives (HDDs), endurance is still a big concern for SSD devices. Storing data with different lifetime in an SSD can cause high write amplification (i.e. the ratio of amount of information physically written to the storage media to the logical amount intended to be written), and reduce the endurance and performance of SSDs. Recently, *multi-stream* SSDs have been developed to enable data with different lifetime to be stored in different SSD regions, and thus reduce write amplification. To efficiently use this new multi-streaming technology, it is important to choose appropriate workload features to assign the same streamID to data with similar lifetime. However, we found that streamID identification using different features may have varying impacts on the final write amplification of multi-stream SSDs. Therefore, we develop a portable and adoptable framework to study the impacts of different workload features and their combinations on write amplification. We also introduce a new feature, named "coherency", to capture the friendship among write operations with respect to their update time. **We finally propose a feature-based stream identification approach, which co-relates the measurable workload attributes (such as I/O size, I/O rate, etc.) with high level workload features (such as frequency, sequentiality etc.) and determines a good combination of workload features for assigning streamIDs [21, 23, 25].** Our evaluation results show that our proposed approach can always reduce the Write Amplification Factor (WAF) by using appropriate features for stream assignment. **This work was awarded the best paper award in IEEE CLOUD'2018.**

In recent years, more and more datacenters have started to replace traditional Serial Advanced Technology Attachment (SATA) and Serial Attached SCSI (SAS) SSDs with Non-Volatile Memory Express (NVMe) SSDs due to NVMe's outstanding performance. However, for historical reasons, current popular deployments of NVMe in Virtual Machine (VM) hypervisor-based platforms (such as VMware ESXi) have numbers of intermediate queues along the I/O stack. As a result, performance is bottlenecked by synchronization locks in these queues, cross-VM interference induces I/O latency, and most importantly, up-to-64K-queue capability of NVMe SSDs cannot be fully utilized. **We develop a hybrid framework of NVMe-based storage system called H-NVMe [15], which provides two VM I/O stack deployment modes Parallel Queue Mode and Direct Access Mode. The first mode increases parallelism and enables lock-free operations by implementing local lightweight queues in the NVMe driver. The second mode further bypasses the entire I/O stack in the hypervisor layer and allows trusted user applications whose hosting Virtual Machine Disk (VMDK) files are attached with our customized vSphere IOFilters to directly access NVMe SSDs to improve the performance isolation.** This suits premium users who have higher priorities and the permission to attach IOFilter to their VMDKs. H-NVMe is implemented on VMware EXSi 6.0.0, and our evaluation results show that the proposed H-NVMe framework can significant improve throughputs and bandwidths compared to the original inbox NVMe solution. **This work was awarded the best paper award in IEEE IPCCC'2017.**

Recently, the capital expenditure of flash-based SSDs keeps declining and the storage capacity of SSDs keeps increasing. As a result, all-flash storage systems have started to become more economically viable for large shared storage installations in datacenters, where metrics like Total Cost of Ownership (TCO) are of paramount importance. On the other hand, flash devices suffer from write amplification, which, if unaccounted, can substantially increase the TCO of a storage system. **In [24], we first develop a TCO model for datacenter all-flash storage systems, and then plug a write amplification model (WAF) of NVMe SSDs that we build based on empirical data into this TCO model.** Our new WAF model accounts for workload characteristics like write rate and percentage of sequential writes. Furthermore, using both the TCO and WAF models as the optimization criterion, we design new flash resource management schemes named MINTCO, to guide datacenter managers to make workload allocation decisions under the consideration of TCO for SSDs. Based on that, we also develop MINTCO-RAID to support RAID SSDs and MINTCO-OFFLINE to optimize the offline workload-disk deployment problem during the initialization phase. Experimental results show that MINTCO can reduce the TCO and provide relatively high throughput and space utilization of the entire datacenter storage resources.

- **Enhancing the Performance of Big-Data Frameworks by Storage Resource Management [13, 9, 10, 11, 27, 12, 16]**

Hypervisor-based virtualization technology has been successfully used to deploy high-performance and scalable infrastructure for Hadoop, and now Spark applications. Container-based virtualization techniques are becoming an important option, which is increasingly used due to their lightweight operation and better scaling when compared to VM. With containerization techniques such as Docker becoming mature and promising better performance, we can use Docker to speed-up big data applications. However, as applications have different behaviors and resource requirements, before replacing traditional hypervisor-based virtual machines with Docker,

it is important to analyze and compare performance of applications running in the cloud with VMs and Docker containers. VM provides distributed resource management for different virtual machines running with their own allocated resources, while Docker relies on shared pool of resources among all containers. **We investigate the performance of different Apache Spark applications using both Virtual Machines and Docker containers [13].** In addition to makespan and execution time, we also analyze different resource utilizations (CPU, disk, memory, etc.) by Spark applications. Our results show that Spark using Docker can obtain speed-up of over 10 times when compared to using VM. However, we observe that this may not apply to all applications due to different workload patterns and different resource management schemes performed by virtual machines and containers. Our work can guide application developers, system administrators and researchers to better design and deploy big data applications on their platforms to improve the overall performance.

By using fast back-end storage, performance benefits of a lightweight container platform can be leveraged with quick I/O response. Nevertheless, the performance of simultaneously executing multiple instances of same or different applications may vary significantly with the number of containers. The performance may also vary with the nature of applications because different applications can exhibit different nature on SSDs in terms of I/O types (read/write), I/O access pattern (random/sequential), I/O size, etc. Therefore, **we strive to investigate and analyze the performance characterization of both homogeneous and heterogeneous mixtures of I/O intensive containerized applications, operating with high performance NVMe SSDs and derive novel design guidelines for achieving an optimal and fair operation of the both homogeneous and heterogeneous mixtures. [11] We also perform characterization of a Dockerized NoSQL database on an NVMe-over-Fabrics (NVMf) prototype and show that its performance matches closely to that of direct attached storage [12, 16].** We conduct experiments on scaling the performance of NVMf to multiple nodes and present the challenges and projections for future storage system design. **By leveraging these design guidelines, we further develop a new docker controller for scheduling workload containers of different types of applications [9, 10, 27].** Our controller decides the optimal batches of simultaneously operating containers in order to minimize total execution time and maximize resource utilization. Meanwhile, our controller also strives to balance the throughput among all simultaneously running applications. We develop this new docker controller by solving an optimization problem using five different optimization solvers. We conduct our experiments in a platform of multiple docker containers operating on an array of three enterprise NVMe drives. We further evaluate our controller using different applications of diverse I/O behaviors and compare it with simultaneous operation of containers without the controller. Our evaluation results show that our new docker workload controller helps speed-up the overall execution of multiple applications on SSDs.

- **Designing New Techniques for Memory Management on Large-Scale Enterprise Cloud Infrastructures [7, 5, 6, 8]**

  In a shared virtualized storage system that runs VMs with heterogeneous IO demands, it becomes a problem for the hypervisor to cost-effectively partition and allocate memory resources among multiple VMs. There are two straightforward approaches to solving this problem: equally assigning memory to each VM or managing memory resources in a fair competition mode. Unfortunately, neither of these approaches can fully utilize the benefits of memory resources, particularly when the workloads frequently change and bursty IOs occur from time to time. In this research, **we design a Global Resource Management solution - GREM [7], which aims to fully utilize SSD resources as a second level cache under the consideration of performance isolation. In particular, GREM takes dynamic IO demands of all VMs into consideration to split the entire SSD space into a long-term zone and a short-term zone, and cost-effectively updates the content of SSDs in these two zones.** GREM is able to adaptively adjust the reservation for each VM inside the long-term zone based on their IO changes. GREM can further dynamically partition SSDs between the long- and short-term zones during runtime by leveraging the feedbacks from both cache performance and bursty workloads. Experimental results show that GREM can capture the cross-VM IO changes to make correct decisions on resource allocation, and thus obtain high IO hit ratio and low IO management costs, compared with both traditional and state-of the-art caching algorithms.

  While working to increase memory utilization, we notice that fundamental efficiency of many algorithms mainly depends on the data temperature identification. We realize that data temperature identification is an importance issue of many fields like data caching and storage tiering in modern flash-based storage systems. With the technological advancement of memory and storage, data temperature identification is no longer just a classification of hot and cold, but instead becomes a "multi-streaming" data categorization problem to classify data into multiple categories according to their temperature. Therefore, **we propose a novel data temperature identification scheme [5, 6] that adopts bloom filters to efficiently capture both frequency and recency of data blocks and accurately identify the exact data temperature for each data block.** Moreover, in bloom filter data structure we replace the original OR operation with the XOR masking operation such that our scheme can delete or reset bits in bloom filters and thus avoid high false positives due to saturation. We further utilize twin bloom filters to alternatively keep unmasked clean copies of data and thus ensure low

false negative rate. Our extensive evaluation results show that our new scheme can accurately identify the exact data temperature with low false identification rates across different synthetic and real I/O workloads. More importantly, our scheme consumes less memory space compared to other existing data temperature identification schemes.

In recent years, all-flash multi-tier storage systems is widely adopted in the enterprise datacenters. However, existing caching or tiering solutions for SSD-HDD hybrid storage systems are not suitable for all-flash storage systems. This is because that all-flash storage systems do not have a large speed difference (e.g., 10x) among each tier. Instead, different specialties (such as high performance, high capacity, etc.) of each tier should be taken into consideration. Motivated by this, **we develop an automatic data placement manager called AutoTiering [8] to handle VMDK allocation and migration in an all-flash multitier datacenter to best utilize the storage resource, optimize the performance, and reduce the migration overhead.** AutoTiering is based on an optimization framework, whose core technique is to predict VM's performance change on different tiers with different specialties without conducting real migration. As far as we know, AutoTiering is the first optimization solution designed for all-flash multi-tier datacenters. We implement AutoTiering on VMware ESXi, and experimental results show that it can significantly improve the I/O performance compared to existing solutions.

- **Innovating Performance Prediction and Scheduling Techniques for System Computation and Communication [2, 3, 4]**

  How to achieve the best performance with an optimal configuration of parallel resources (e.g., number of processes and number of cores) is a challenging research problem. The prediction of expected performance prior to the porting of an actual implementation on a cloud platform can save time and resources spent in experimentally finding the optimal performance point. **We develop new modeling technique, named Fine grained Model (FiM) [2, 3] which consists of two main components: (1) FiM-Cal, and (2) FiM-Com. The goal of FiM-Cal is to predict the calculation time by using a stochastic Markov model and a machine learning model.** The stochastic markov model is built using the probabilistic technique to estimate the impact of increase in the number of parallel processes. We first develop the base case of the parallel paradigm and then derive a generic model that is applicable to any number of parallel processes as well as any number of dependent stages (e.g., iterations) of an application. The base case of the Markov model is calibrated using the minimum number of system parameters. The machine learning model is then designed to extrapolate the calibrated parameters for the Stochastic Markov model in order to adapt to changes in application parameters such as datasets. **The goal of FiM-Com is to predict the communication time using a set of simulation queuing models.** Here our motive is to get a quick estimate using a simplified prediction model. Such an estimate of communication time along with calculation time can provide instant insight to users. Thus, our FiM approach can use the minimum possible calibration parameters to quickly predict the expected computation and communication time as well as the optimal number of processes for compute platform configuration.

  On the other hand, large scale data analysis is of great importance in a variety of research and industrial areas during the age of data explosion and cloud computing. Hadoop MapReduce ecosystem is evolving into its next generation, called Hadoop YARN (Yet Another Resource Negotiator). YARN uses the same scheduling mechanisms as traditional Hadoop and supports the existing scheduling policies (such as First In First Out (FIFO), Fair and Capacity) as the default schedulers. However, we found that a resource (or "container") starvation problem exists in the present YARN scheduling under Fair and Capacity, when the resource requisition of applications is beyond the amount that the cluster can provide. In such a case, the YARN system will be halted if all resources are occupied by ApplicationMasters, a special task of each job that negotiates resources for processing tasks and coordinates job execution. To solve this problem, **we propose an automatic and dynamic admission control mechanism named, AutoAdmin [4] to prevent the ceasing situation happened when the requested amount of resources exceeds the cluster's resource capacity, and dynamically reserve resources for processing tasks in order to obtain good performance, e.g., reducing makespans of MapReduce jobs.** After collecting resource usage information of each work node, our mechanism dynamically predicts the amount of reserved resources for processing tasks and automatically controls running jobs based on the prediction. We implement this new mechanism in Hadoop YARN and evaluate it with representative MapReduce benchmarks.

Future Research Directions

- **Leveraging Productivity of Data-Intensive Workloads by Emerging Flash-Based Technology and Modeling Memory and Storage Interactions:**

  From my previous research on emerging storage technologies, I found that architecting flash based storage

devices that leverage productivity of data intensive workloads is challenging. Multiple aspects such as performance, capacity and reliability play important roles to process and store big data. I plan to design and implement storage techniques that accommodate data management requirements of modern Machine Learning (ML) and Artificial Intelligent (AI) requirements. In particular, my primary plan is to first evolve flash storage that directly supports key-value. Traditional SSDs use a block interface. Therefore, data must be converted to blocks to allow applications to talk to any SSD. Unfortunately, effective data conversion is costly from an operational standpoint and can often become a performance bottleneck in scale-out and scale-up infrastructures. One unique solution to address this costly issue is to combine the conventional SSD technology and the conversion layer into a single SSD. This would not only simplify the conversion process, but also significantly extend the drive's capabilities. I plan to accelerate time consuming unsupervised and reinforcement learning techniques, by using and evolving fast storage methods.

Second, flash technology is constantly improving itself with rapid developments such as in-flash computing devices like Field Programmable Gate Array (FPGA), Key-Value SSDs and Multi-stream SSDs. In such a versatile environment, both developers and users of flash technology require thorough benchmarking to evaluate performance impacts. However, various real applications need to be installed and configured, large databases need to be generated, and Input/Output (I/O) stack of operating system need to be modified to be compatible with new storage technologies. Executing widely-used real applications is important to study impacts of different hardware and firmware evolution. Moreover, benchmarking to optimize the performance for big data workloads is a critical process. Thus, I plan to leverage evolving flash technologies by development of peripheral infrastructures such as drivers, libraries and OS kernels to study their performance impacts. I also plan to integrate my understanding of flash-based storage operation into the new modeling techniques for simulating these emerging trends. This will allow us to conduct in-depth modular analysis and develop useful insights towards development of future storage devices.

- **Data Security of Large-scale Computing Systems:**

  Despite the essentially "limitless" storage capacity available in a public cloud, the cloud can't entirely solve the enterprise capacity challenge. For data that must be immediately available, the inherent lag in connecting to a public cloud may remove cloud storage as an option. Additionally, cloud storage is expensive, especially if we were dealing with massive amounts of data that must always be available. I see that blockchain technology can play a central role in solving this storage-capacity problem. My vision is to imagine a world that offers a local storage marketplace as a community grid composed of storage providers and consumers. If anyone needs additional capacity, Information Technology (IT) would be able to securely purchase storage from nearby organizations that have more than what they need. Conversely, organizations with excessive capacity could monetize it by selling storage to their neighbors.

  In this world of distributed storage, instead of creating a ledger of cryptocurrency transactions, I want to use blockchain to record data movement and ownership. Thus, there would be no need for a third party to secure trust. The blockchain will secure trust all by itself. Though buying and selling storage with other nearby businesses would reduce latency, the amount of cryptography required to power the marketplace, both in encryption for security and blockchain, would require additional raw power from computing resources. Given these technical barriers, but many economic and operational advantages, this will be interesting research direction to explore.

- **Full Stack Acceleration by Resolving Performance Bottlenecks Intermediate Layers:**

  My research has demonstrated that the massive performance benefits can be achieved when all the layers of stack such as computing, communication, memory and storage are studied together. I recognized that evolving each of these components individually is the primary step. Then, for further performance benefits, it is important to optimize interactions among these components to convalesce performance of the whole united system. I notice that efficient management of data-centric systems can not be accomplished without a proper understanding of full system stacks. Motivated by this, I envision to accelerate large-scale systems (e.g., cloud infrastructures, data centers, High Performance Computing (HPC) systems). My future research will attempt to understand interplay among different system layers by consolidating applications, optimizing parallelism of virtualized frameworks and high performance computing infrastructures, and improving data management for memory and persistent storage devices.

Summary

To conduct productive research, I am looking forward to learn much more than I know now. Interacting with peers, students, and exploring more about other research areas will contribute to many new ideas and exciting research. Computer systems is an area full of exciting but challenging problems. I look forward to putting my best efforts on

solving these problems. I am enthusiastic to further enhance my research, and make significant contributions that leave positive impact on society. Further information about my current research and my publications are available at http://www.http://nucsrl.coe.neu.edu/?q=janki

References

[1] D. Reinsel, J. Gantz, and J. Rydning, "Data Age 2025:The Evolution of Data to Life-Critical, Don't Focus on Big Data; Focus on the Data That's Big," 2018.

[2] J. Bhimani, N. Mi, M. Leeser, and Z. Yang, "FiM: Performance Prediction Model for Parallel Computation in Iterative Data Processing Applications," in *International Conference on Cloud Computing (CLOUD)*, IEEE, 2017.

[3] J. Bhimani, N. Mi, and M. Leeser, "Performance prediction techniques for scalable large data processing in distributed MPI systems," in *Performance Computing and Communications Conference (IPCCC)*, IEEE, 2016.

[4] Z. Yang, J. Bhimani, Y. Yao, C.-H. Lin, J. Wang, N. Mi, and B. Sheng, "AutoAdmin: Admission Control in YARN Clusters Based on Dynamic Resource Reservation," in *Scalable Computing: Practice and Experience, Special Issue on Advances in Emerging Wireless Communications and Networking*, vol. 19, 2018.

[5] J. Bhimani, N. Mi, and B. Sheng, "BloomStream: Data Temperature Identification for Flash Based Memory Storage Using Bloom Filters," in *International Conference on Cloud Computing (CLOUD)*, IEEE, 2018.

[6] J. S. Bhimani, R. Pandurangan, C. Choi, and V. Balakrishnan, "System and method for identifying hot data and stream in a solid-state drive," 2018. US Patent App. 15/895,797.

[7] Z. Yang, J. Tai, J. Bhimani, J. Wang, N. Mi, and B. Sheng, "GREM: Dynamic SSD Resource Allocation In Virtualized Storage Systems With Heterogeneous IO Workloads," in *Performance Computing and Communications Conference (IPCCC)*, IEEE, 2016.

[8] Z. Yang, M. Hoseinzadeh, A. Andrews, C. Mayers, D. T. Evans, R. T. Bolt, J. Bhimani, N. Mi, and S. Swanson, "AutoTiering: Automatic Data Placement Manager in Multi-Tier All-Flash Datacenter," in *Performance Computing and Communications Conference (IPCCC)*, IEEE, 2017.

[9] J. Bhimani, Z. Yang, N. Mi, J. Yang, Q. Xu, M. Awasthi, R. Pandurangan, and V. Balakrishnan, "Docker container scheduler for I/O intensive applications running on NVMe SSDs," *Transactions on Multi-Scale Computing Systems*, 2018.

[10] J. S. Bhimani, A. Subramanian, V. Balakrishnan, and J. Yang, "Container workload scheduler and methods of scheduling container workloads," 2018. US Patent App. 15/820,856.

[11] J. Bhimani, J. Yang, Z. Yang, N. Mi, Q. Xu, M. Awasthi, R. Pandurangan, and V. Balakrishnan, "Understanding performance of I/O intensive containerized applications for NVMe SSDs," in *Performance Computing and Communications Conference (IPCCC)*, IEEE, 2016.

[12] Q. Xu, M. Awasthi, K. Malladi, J. Bhimani, J. Yang, and M. Annavaram, "Performance Analysis of Containerized Applications on Local and Remote Storage," in *International Conference on Massive Storage Systems and Technology (MSST)*, 2017.

[13] J. Bhimani, Z. Yang, M. Leeser, and N. Mi, "Accelerating Big Data Applications Using Lightweight Virtualization Framework on Enterprise Cloud," in *High Performance Extreme Computing Conference (HPEC)*, IEEE, 2017.

[14] H. Gao, Z. Yang, J. Bhimani, T. Wang, J. Wang, B. Sheng, and N. Mi, "AutoPath: Harnessing Parallel Execution Paths for Efficient Resource Allocation in Multi-Stage Big Data Frameworks," in *International Conference on Computer Communications and Networks (ICCCN)*, IEEE, 2017.

[15] Z. Yang, M. Hoseinzadeh, P. Wong, J. Artoux, C. Mayers, D. T. Evans, R. T. Bolt, J. Bhimani, N. Mi, and S. Swanson, "H-NVMe: A Hybrid Framework of NVMe-based Storage System in Cloud Computing Environment," in *International Performance Computing and Communications Conference (IPCCC)*, IEEE, 2017. ***Best Paper Award***.

[16] Q. Xu, M. Awasthi, K. Malladi, J. Bhimani, J. Yang, and M. Annavaram, "Docker Characterization on High Performance SSDs," in *International Symposium on Performance Analysis of Systems and Software (ISPASS)*, IEEE, 2017.

[17] J. S. Bhimani, R. Pandurangan, V. Balakrishnan, and C. Choi, "Representative I/O generator," 2018. US Patent App. 15/853,419.

[18] J. Bhimani, M. Leeser, and N. Mi, "Design space exploration of GPU Accelerated cluster systems for optimal data transfer using PCIe bus," in *High Performance Extreme Computing Conference (HPEC)*, IEEE, 2016.

[19] J. Bhimani, M. Leeser, and N. Mi, "Accelerating K-Means clustering with parallel implementations and GPU computing," in *High Performance Extreme Computing Conference (HPEC)*, IEEE, 2015.

[20] C. Liu, J. Bhimani, and M. Leeser, "Using High Level GPU Tasks to Explore Memory and Communications Options on Heterogeneous Platforms," in *Workshop on Software Engineering Methods for Parallel and High Performance Applications*, ACM, 2017.

[21] J. Bhimani, N. Mi, Z. Yang, J. Yang, R. Pandurangan, C. Choi, and V. Balakrishnan, "FIOS: Feature Based I/O Stream Identification for Improving Endurance of Multi-Stream SSDs," in *International Conference on Cloud Computing (CLOUD)*, IEEE, 2018. ***Best Paper Award***.

[22] Z. Yang, Y. Wang, J. Bhamini, C. C. Tan, and N. Mi, "EAD: elasticity aware deduplication manager for datacenters with multi-tier storage systems," *Cluster Computing*, 2018.

[23] J. Bhimani, J. Yang, Z. Yang, N. Mi, N. K. Giri, R. Pandurangan, C. Choi, and V. Balakrishnan, "Enhancing SSDs with Multi-Stream: What? Why? How?," in *International Performance Computing and Communications Conference (IPCCC), Poster Paper*, IEEE, 2017.

[24] Z. Yang, M. Awasthi, M. Ghosh, J. Bhimani, and N. Mi, "I/O Workload Management for All-Flash Datacenter Storage Systems Based on Total Cost of Ownership," *Transactions on Big Data*, 2018.

[25] J. S. Bhimani, J. Yang, C. Choi, and J. Huo, "Smart I/O stream detection based on multiple attributes," 2017. US Patent App. 15/344,422.

[26] Z. Yang, J. Bhimani, J. Wang, D. Evans, and N. Mi, "Automatic and Scalable Data Replication Manager in Distributed Computation and Storage Infrastructure of Cyber-Physical Systems," in *Scalable Computing: Practice and Experience, Special Issue on Communication, Computing, and Networking in Cyber-Physical Systems*, 2017.

[27] J. Bhimani, H. Huen, J. Yang, M. Awasthi, V. Balakrishnan, and J. Martineau, "Intelligent Controller for Containerized Applications," 2017. US Patent App. 15/379327.