Performance Analysis in HyperFlex and vSAN Hyper Convergence Platforms for Online Course Consideration

CHEN-KUN TSUNG¹, (Member, IEEE), JUNG-CHUN LIU², SHU-HUA LIU², VINOD KUMAR VERMA³, AND CHAO-TUNG YANG²,4, (Senior Member, IEEE)

¹Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411030, Taiwan
²Department of Computer Science, Tunghai University, Taichung 407224, Taiwan
³Department of Computer Science and Engineering, Sant Longowal Institute of Engineering and Technology, Sangrur 148106, India
⁴Research Center for Smart Sustainable Circular Economy, Tunghai University, Taichung 407224, Taiwan

ABSTRACT With the ubiquity of high-speed networks and cloud technologies, hyper-convergence (HC) has become commonplace and satisfies flex allocation requirements. This study compared the VMware virtual storage area network (vSAN), which has the largest market share, with Cisco HyperFlex, released by Cisco for high-level network applications, to assess the performance of various HC technologies in running virtual machines in most virtualized environments. The experiments consider the benchmark (HCIBench) to evaluate the platform’s performance. The benchmark provides objective performance scores. Thus, the performance results derived by HCIBench could be applied to discuss the appropriate scenarios for both VMware vSAN and Cisco HyperFlex. The experiments simulate common application scenarios to discover good HC platforms for real-world requirements. The VMware vSAN demonstrated more robust overall performance (about 38.18% and 22.72% improvement for IOPS and transmission speed, respectively), when a few virtual machines were used, and the write load was heavy. The difference in CPU usage between the two platforms is not too much (about 2.565%). The vSAN is recommended for general purposes, while HyperFlex performs better in areas with high write requirements (such as the data center).

INDEX TERMS Hyper-convergence, HyperFlex, performance analysis, virtual storage area network.

I. INTRODUCTION

With the ubiquity of high-speed networks, virtualization technologies have become widely used in fields such as manufacturing [18], [19], [22], education [20], [21], medicine [23], [28], education [26], [27], and life information [24], [25]. Hyper-convergence (HC) technologies [1], [2], [3] have become the mainstream architecture to integrate virtualization platforms and storage devices effectively. Manufacturers have also released hyper-convergence infrastructures (HCIs) with certain design concepts to provide outstanding performance in various scenarios. Although these particular design concepts can improve user experience and increase the convenience of management, performance analyses for various HC environments, particularly storage I/O, computation throughput, and read/write latency, have yet to be conducted [9], [14].

Hardware virtualization, cloud computing, software-defined storage (SDS), and HCI have been proposed to maximize hardware resources’ utility and facilitate the sharing of server hardware resources. The above technologies are introduced as follows.

- In hardware virtualization, virtual and physical hardware are matched through a hypervisor, which enables an operating system to be run in a virtual hardware environment [4], [5], [6]. Consequently, one server can run multiple virtual machines (VMs), with each device...
executing a different operating system and software. A hypervisor provides servers with physical resource sharing, convenient hardware configurations, and added storage utilization. Although hardware visualization combines multiple services into a single physical server, platform performance is determined by storage deployment strategies. Common storage deployment strategies include direct-attached storage, which is the direct use of the server’s local disks; network-attached storage, built on the File Transmission Protocol; and storage area networks (SANs), which are based on local-area networks.

- Cloud computing enables the integration and application of resources in various fields [7], [8], [9]. The three biggest cloud service providers are Amazon, Google, and Microsoft, which have released cloud computing platforms to provide their customers with storage, computing, database, website, and machine learning services. Cloud computing can be distinguished into three service models: IaaS, PaaS, and SaaS. The commonality among these service models is that they require a large amount of storage space; however, conventional disk arrays and SANs cannot meet this demand. Therefore, the high level of resource integration offered by cloud computing is a solution for heterogeneous environments.
- SDS facilitates hardware expansion by enhancing the horizontal integration of resources. SDS provides the following features in storage management: 1) the use of local disks, 2) the ability to scale out multiple servers, 3) data transmission between servers through high-speed networks, and 4) central management of all disks through a virtual disk pool.
- The purpose of an HCI is to increase the utilization of local resources. Server storage idles after a VM is placed in the SAN storage, and local storage can provide the best data transformation; as such, the performance of a platform can be improved through the integration of local resources. The core concept of HCI is the integration of virtualization platforms and SDS to maximize the utilization of servers’ computation power and storage. HCIs can be distinguished into the following four types:
  1) VMware vSAN implements SDS. Because vSAN is a native application of VMware vSphere, it can be integrated into virtualization platforms to form an HC platform. However, vSAN only supports VMware applications.
  2) Azure Stack HCI is created by Microsoft based on the Storage Spaces Direct, which is similar to SDS. This solution combines Hyper-V and storage spaces to form HC platforms.
  3) Unique feature platforms: In addition to these HC platforms, platforms such as HPE OmniStack (acquired from SimpliVity), Cisco HyperFlex (obtained from Springpath), and NetApp SolidFire Element OS (received from SolidFire) offer performance and management advantages for particular circumstances.

HCI is the state-of-the-art technology for maximizing hardware utilization, and VMware vSAN occupies a high market share while Cisco HyperFlex provides the solution for high-end markets. However, the users need to learn the appropriate platform. Therefore, this work would like to compare the performance between VMware vSAN and Cisco HyperFlex to determine each platform’s advantages and applicable usage scenarios with high user experience.

To capture the system performance of the target HCI platforms, the HCI-Bench, the famous benchmark for evaluating the performance of HCI platforms, is considered in this work. The overall performance of HCI platforms is challenging to be quantized, and the performance required by various scenarios is different. Benchmarks provide objective evaluation results, so discussing the performance gap between target systems is appropriate for benchmarks. To cover common usage, the experiment considers the system throughput, the latency, and the resource utilization to discuss the appropriate scenario of VMware vSAN and Cisco HyperFlex but does not aim at finding the HCI platform with the highest performance.

This study implemented two HCI platforms based on VMware vSAN and Cisco HyperFlex and understood the real-world performance in input/output operations per second, transmission speed, read/write latency, and CPU/RAM usage. Under four simulation scenarios: general use, random read, sequential write, and online transactional processing, VMware vSAN demonstrated more robust overall performance. In contrast, the Cisco HyperFlex provided better performance in reading latency and write latency when few virtual machines were used and the write load was high. The general memory usage of the vSAN was approximately 7.5% lower than that of the Cisco HyperFlex, indicating that the vSAN can run more VMs. Therefore, VMware vSAN is the balanced solution in economy and average efficiency considerations for common usage purposes.

The organization of the remaining paper is listed as follows. Section III describes the information on target HCI platforms, including the hardware deployment, data collection, and benchmark configuration. Section IV shows the experiment results and the performance comparison between target HCI platforms. The conclusion and future works are stated in Section V.

II. RELATED WORKS
The traditional server architecture collects all storage, and the controller virtualizes the repository as a unique service with an access portal. The significant advantage of traditional architecture is easy management. Tsung et al. provide the performance evaluation for the traditional server architecture with vSAN [26]. For the big block transfers, the performance of vSAN is improved by about 69%. Banerjee and Srivastava provide a framework to detect the system performance issue [29]. The traditional architecture provides maintenance convenience, so the system administrators would like
to deploy the system with traditional architecture in heavy traffic [30].

Easy maintenance is the primary advantage of traditional architecture. However, the performance bottleneck takes place in the data transformation. Tsung et al. considered the network with a very wide bandwidth to capture the maximum performance of vSAN [26], but the improvement is limited. To improve the overall system performance, the HCI architecture is proposed [31], [32]. The significant difference between the HCI and traditional architectures is the storage position. The storage is moved from the central storage pool to the servers. However, the storage in various servers is also virtualized as a single access portal, so easy maintenance is not sacrificed in HCI platforms. From the experiment results of Leite et al., the HCI provides 7% additional performance than the system with local storages [33]. Local storage performance is better than centralized repositories, so HCI architectures offer higher performance than traditional architectures.

III. SYSTEM ARCHITECTURE

A. TARGET SYSTEM

The comparison test environments were Dell EMC VxRail E560F with VMware vSAN and Cisco HyperFlex HX220C-M5S with Cisco HyperFlex. The hardware specifications were as follows:

- Dell EMC VxRail E560F involves the central management of the data disk and cache disk of every node in the cluster through a vSAN by using VMFS 6. The disks are managed by a disk group comprising the buffer cache and data storage capacity. Because this study used the all-flash architecture, the disk groups consisted of SSDs. The cache disk is responsible for the read cache and write buffer. Within the all-flash architecture, the entire cache disk is the write buffer; in the hybrid mode (with a SATA or SAS data disk), the cache disk is 70% read cache and 30% write buffer. The recommendation in VMware is to determine the hardware configurations for cache and capacity and the algorithms for the write buffer flush based on terabytes written to optimize cache performance. Multidisk groups improve cluster performance, but this does not mean that the high availability can be increased; instead, a cache failure only affects the performance of the disk group involved. When the capacity disk fails, only the involved disk group must be rebuilt, which shortens the rebuild disk group time. This experiment involved using an HC platform comprising all-flash servers that support VMware platforms [13].

- Cisco HyperFlex HX220C-M5S support common virtualization platforms such as VMware and Hyper-V [10], [11], [12]. This study used all-flash servers. The data disk and cache disk of each node in the cluster is configured and managed by the HyperFlex HX Data Platform using network file system 3. A dedicated virtualization gateway executes virtualization platforms such as VMware vSphere, Microsoft Hyper-V, and Kubernetes.

This study used VMware vSphere to achieve consistency among the test environments.

Table 1 presents the HC platforms used in this study. All storage specifications were the same except for the cache configurations. Because of the requirement for identical capacity, the Dell EMC VxRail used two 800 NVMe SSDs, whereas the Cisco HyperFlex HS used one 1.6TB NVMe SSD. In the experiment, a comparison was made to address this difference.

B. HARDWARE DEPLOYMENT

The Cisco HyperFlex HX220C-M5S also consisted of four servers connected to two Cisco switches (Cisco UCS 6332 IRU Fabric Interconnect), which were related to the Cisco 6807 switch within a 40GbE high-speed network (Fig. 2).

C. CLUSTER STATUS DATA COLLECTION

The HC platforms in this study used VMware, and consequently, their performance was measured using an HCIBench comprising a Controller VM and Guest VMs (see Fig. 3 for the architecture) [15], [16], [17]. With the given test parameters, the Controller VM combined the test and configuration files with the Guest VM template and then created and executed Guest VMs in assigned numbers and specifications through the RVC on the vSphere.

1 https://flings.vmware.com/hcibench
TABLE 1. Hardware specifications of HC platforms.

|              | Dell EMC VxRail E560F           | Cisco HyperFlex HX220C-M5S        |
|--------------|--------------------------------|----------------------------------|
| CPU          | Intel Xeon Gold 6130 * 2        | Intel Xeon Gold 6130 * 2         |
| RAM          | 768GB                          | 768GB                            |
| Storage      | SATA SSD 3.8T * 6 and 800GB NVMe SSD * 2 | SATA SSD 3.8T * 6 and 1.6TB NVMe SSD * 1 |
| NIC          | 40GbE-CPRI/QSFP                 | 40GbE-CPRI/QSFP                  |
| HC Node      | 4                               | 4                                |
| Switch       | Dell EMC Switch S4128F-ON x 2    | Cisco UCS 6332 IRU Fabric Interconnect x 2 |

The HCIBench used telegraf+influxdb+Grafana to gather, store, and display performance data (Fig. 4), while the deployment of HCIBench is illustrated in 5. The HCIBench controller on the VMware vSphere executed the test programs on the Guest VM through SSH; the telegraf proxy program sent the Guest VM performance data from Graphite to influxdb and stored them. Last, Grafana read and visualized the data in influxdb.

D. BENCHMARK CONFIGURATION

Before testing, the read and write caches of both HC platforms were cleared to ensure the accuracy of the data. Because vSAN Debug Mode and EASY RUN were only effective in vSAN, they were turned off. In addition, to reduce the time required to rebuild the Guest VMs, the Reuse VMs If Possible option was activated, and the Clean Up VMs option was deactivated; this reduced the testing time for the various parameters. The size of the VM data disks remained unchanged, and the number of VMs and disks did not increase between tests; the experiments began with the greatest number of VMs and disks, which were then gradually reduced.

Because each RAID class generates different write penalties, errors occur when input/output operations per second (IOPS) are calculated; for example, one write of RAID-1 consists of two writes, so the write penalty is 2. Consequently, the HCIBench initializes disk content to reduce the first-write penalty. If deduplication must be enabled, RANDOM is selected; otherwise, ZERO is set. This study used RANDOM for the HCI to enable deduplication; this was the most time-consuming portion of the test.

Random and Sequential accesses are essential metrics in traditional comparisons, and the metrics provide performance in continuous and discrete use. However, the unit in a cloud environment is VM rather than a file, so the requests come from various users (i.e., VMs). Therefore, the experiments consider random access to simulate real-world behavior.

IV. EXPERIMENT

This section uses the HCIBench to capture the performance results of VMware vSAN and Cisco HyperFlex regarding the system throughput, latency, and resource utilization. Fig. 6 is the experiment flow chart of this work. In the beginning, the HCI hardware, as mentioned in Section III, and the HCIBench are prepared. The configuration of test cases is deployed in the HCIBench. After the HCIBench finishes all test cases, the experiment results are captured and used to analyze and compare the performance of VMware vSAN and Cisco HyperFlex.

HCIBench is the credible benchmark for evaluating the performance of HCI platforms, and some system administrators would like to use the HCIBench to optimize the system...
Three parts of system performance are considered in the experiments.

1) The system throughput: When user applications access the storage, higher system throughput results in less process time, and vice versa. Common system throughput indexes are IOPS and transmission speed.

2) The latency: The user experience comes from the latency. A high latency indicates low response speed, which is the user’s major consideration.

3) The resource utilization: More resource required by VMs indicates that fewer VMs could be deployed on the service side. So, less resource utilization means more VMs can be executed concurrently.

As illustrated in Fig. 5, HCI-Bench provides data collection and visualization, and the performance analyzers could evaluate the system performance in terms of different indexes and scenarios. Therefore, the performance analyzers focus on the system evaluation rather than dealing with manipulating the experiment detail.

### A. TEST SCENARIO

Table 2 presents the experimental scenarios. Four parameters were manually deployed on the HCI-Bench and tested separately on the two HC platforms. The scenario of general usage consists of 70% read and 30% write with 100% random process according to the statistical results of the user behavior. To simulate user behavior in VM environments, the scenario with 100% with 100% random process is considered in scenario 2. Online Transactional Processing (OLTP) is essential in common transactions, so 50% read and 50% with 100% random process provide the OLTP behavior simulation. Most users require random access in the VM environment since the users are working in individual spaces (i.e., VM). However, this study still considers an evaluation of sequential processes in the experiment.

The goal of the experiments was to test the effects of different HCIs (VMware vSAN vs. Cisco HyperFlex) on VM performance in environments with the same hardware specifications and configuration. Therefore, only one of the following variables was altered in each experiment:

- Number of VMs run simultaneously: 4, 8, 12, 16, 20, 24, 28, 32, 36, or 40 (ten sets in total)
- Experimental scenarios: four (Table 2).

The variation in performance of both HC platforms in each scenario was simulated through 40 tests (ten VM configurations multiplied by four scenarios). Each set of parameters was tested for 10 minutes. The time to initialize the disks was 1 to 3 hours, depending on the number of VMs.

### B. EXPERIMENT RESULT

#### 1) IOPS AND TRANSMISSION SPEED

Fig. 7 and Table 3 present the IOPS statistics and averages; the horizontal axis represents the number of VMs, and the vertical axis represents the IOPS. The following is observed:

- **The VMware vSAN outperformed the Cisco HyperFlex in cache policy.** In Scenarios 2 and 4, Dell EMC demonstrated better IOPS for both all-read and all-write situations than Cisco; the IOPS for both platforms was on par with the other two tests. This was posited because the hardware specifications were the same.

- **Experiments showed that the VMware vSAN had better read caching than write caching.** The writing performance of the Dell EMC began to worsen when 24 or more VMs were involved. In the same 100% random read-and-write scenarios, the mean IOPS for 70% reading was 60.76% higher than the IOPS for 50% reading. Thus, the VMware vSAN had better reading caching than writing caching.

- **The mean writing of Cisco HyperFlex was more stable than that of VMware vSAN.** Although the Cisco mean writing IOPS was 28.89% lower than the Dell EMC mean IOPS, its performance was more stable; the Cisco platform outperformed the Dell EMC in Scenario 3.

Fig. 8 and Table 4 present the transmission speed results. The performance of the two HC platforms matched the IOPS results, indicating that the results of this experiment were consistent.
2) READ LATENCY

Fig. 9 and Table 5 present the read latency results.

- The read latency of the VMware vSAN increased with write volume. In Scenario 2, the Dell EMC read latency increased proportionally to the number of VMs but never exceeded 2 ms. The addition of write operations increased the Dell EMC read latency considerably, as did higher write proportions (3 to 7.02 ms). Consequently, the read latency was inferred to be determined by the write volume. This inference is consistent with the observation that the DELL EMC IOPS decreased as the number of test VMs increased in Scenario 4.
- Cisco HyperFlex had stable read latency.

Except for Scenario 2, the Cisco platform demonstrated lower read latencies than the Dell EMC. Fig. 10 and Table 6 present the 95th percentile results, with extreme values excluded. The overall performance of both platforms was similar in Scenarios 1 and 3 and exhibited similar trends to those in Fig. 9 and Table 5. In Scenario 2, the read latency of the Dell EMC platform was 4.26 times lower than that of the Cisco platform, which is consistent with the results of the previous IOPS experiment.

3) WRITE LATENCY

Fig. 11 and Table 7 present the write latency results.

- The write latency increased as the number of VMs increased.
- In all-write scenarios, the Dell EMC platform performed better with fewer VMs, and the Cisco platform performed better with more VMs.
- In scenarios involving both writing and reading, the write latency of the Cisco platform increased as the number of VMs increased. In contrast, the write latency of the Dell EMC platform was consistently within the 1-3 ms range.
Fig. 12 and Table 8 present the 95th percentile write latency results. In Scenario 4, the write latency of the Dell EMC platform was 1.31 times lower than that of the Cisco platform and 3.78 times lower in the data for 28 or fewer VMs. In situations with few VMs, the Dell EMC platform performed exceptionally well.

### TABLE 7. Average write latency in each scenario.

| Scenario | Dell EMC | Cisco | Dell EMC / Cisco |
|----------|----------|-------|------------------|
| Sec.4    | 69.10    | 97.43 | 128.32%          |
| Sec.2    | -        | -     | -                |
| Sec.1    | 1.36     | 3.46  | 253%             |
| Sec.3    | 1.25     | 4.05  | 30.76%           |

### TABLE 8. Average 95th percentile write latency in each scenario.

| Scenario | Dell EMC | Cisco | Dell EMC / Cisco |
|----------|----------|-------|------------------|
| Sec.4    | 146.98   | 192.13| 76.49%           |
| Sec.2    | -        | -     | -                |
| Sec.1    | 1.70     | 10.25 | 16.59%           |
| Sec.3    | 1.10     | 12.05 | 9.13%            |

4) **CPU USAGE**

Fig. 13 and Table 9 present the central processing unit (CPU) usage results. The results indicate little difference in VM CPU usage between the two HCI platforms, with the means falling between 30% and 40%. This shows that the CPU did not affect the I/O performance of the HC platforms.
5) RAM USAGE
Fig. 14 and Table 10 present the simulation results. The results indicate little difference in random-access memory (RAM) consumption between the two HCI platforms, with means of approximately 65%. Because RAM usage was within a reasonable range, it did not affect I/O performance. However, the RAM usage of the DELL EMC platform was approximately 7.5% less than that of the Cisco platform, indicating that the Dell EMC platform performed slightly better in terms of RAM usage.

6) vSAN CPU USAGE
Only the Dell EMC platform used vSAN; therefore, only the performance of Dell EMC in terms of vSAN CPU usage was analyzed (Fig. 15). The average results are listed in Table 11. The results indicate that the vSAN requires more physical CPU computing power when reading a massive amount of data. The CPU usage was 7.64% and 13.97% in Scenarios 4 and 2, respectively; usage was 9.27% and 10.89% in Scenarios 3 and 1, respectively. Consequently, vSAN consumed more CPU power when reading than when writing. This is consistent with the findings that vSAN performed better in read caching than in write caching.

C. DISCUSSION
In the traditional system architectures of server farms, the computation resource is collected as a virtual resource pool, and the storage resource has the same deployment. According to the experiment results proposed by Tsung et al. [26],...
the vSAN architecture drawn in Fig. 16 provides a virtual pool of storage service. Some storage devices (such as HPE MSA 2050 SAN) in the bottom of Fig. 16 provide storage service, and vSAN Network combines all storage devices as a virtual pool for easy access. Therefore, the applications deployed in the VMware application layer could access the vSAN storage via the vSAN Network rather than accessing the storage devices directly. Thus, vSAN provides convenience for accessing and managing repositories.

Convenient access and management are significant advantages of the traditional system architecture, but the performance bottleneck occurs in the vSAN network’s portal. According to the experiment results proposed by Tsung et al. [26], the IOPS result for 100% read 4K data is 43,800 for a 40G network. However, the experiment results show that the HCI platforms of Dell EMC and Cisco provide 131,563 and 81,839, respectively, in the same testing case (Sce. 1) in Table 3. In other words, HCI improves by about 300% and 186% for Dell EMC and Cisco, respectively. The results match the performance bottleneck of vSAN architecture, and the performance improvement is the significant advantage of HCI platforms. Therefore, the HCI platforms provide better system throughput and user experience for high utility usage scenarios.

V. CONCLUSION

This study compared Cisco HyperFlex HX220C-M5S and Dell EMC VxRail E560F, two four-node HC platforms. They had similar hardware specifications and were both VMware virtualization platforms. Consequently, this study can be considered a comparison of Cisco HyperFlex and VMware vSAN HCI. The results indicate that VMware vSAN outperformed Cisco HyperFlex regarding I/O, particularly for reading. This can explain why IDC reported that VMware vSAN dominated 38.7% of the 2020 Q4 HCI market. If few VMs and a large write workload are required, Cisco HyperFlex can provide superior read and write latency.

Other HCI solutions with potential are on the market. No matter which HCI platform is used, HCI will be the mainstream technology for virtualization and storage integration; its low cost, strong performance, and high scalability are considerable advantages in various fields, especially in cloud computing environments that combine reading and writing. Thus, the value and applicability of HCI will continue to increase.

2 https://www.idc.com/getdoc.jsp?containerId=prUS47552021
REFERENCES

[1] C. Melo, J. Dantas, P. Maciel, D. M. Oliveira, J. Araujo, R. Matos, and I. Fê, “Models for hyper-converged cloud computing infrastructures planning,” Int. J. Grid Util. Comput., vol. 11, no. 2, pp. 196–208, 2020.

[2] N. Tabassum, T. Alyas, M. Hamid, M. Saleem, and S. Malik, “Hyper-convergence storage framework for EcoCloud correlates,” Comput. Mater. Continua, vol. 70, no. 1, pp. 1573–1584, 2022.

[3] Y. Zhang, J. Ren, F. Liu, Z. Wang, Y. Song, L. Yin, and Y. Peng, “A novel hyper-converged power for power data centers,” in Proc. IEEE Int. Conf. Power, Intell. Comput. Syst. (ICPCS), Jul. 2021, pp. 391–394.

[4] B. Djordjevic, V. Timcenko, N. Kraljevic, and N. Macek, “File system performance comparison in full hardware virtualization with ESXi, KVM, hyper-V and Xen hypervisors,” Adv. Electr. Comput. Eng., vol. 21, no. 1, pp. 11–20, 2021.

[5] C.-W. Lin, B. Kim, and S. Shiaraishi, “Hardware virtualization and task allocation for plug-and-play automotive systems,” IEEE Design Test, vol. 38, no. 5, pp. 65–73, Oct. 2021.

[6] B. Yi, X. Wang, S. K. Das, K. Li, and M. Huang, “A comprehensive survey of network function virtualization,” Comput. Netw., vol. 133, pp. 212–262, Mar. 2018.

[7] T. Lynn, X. Liang, A. Gourinovitch, J. Morrison, G. Fox, and P. Rosati, “Understanding the determinants of cloud computing adoption for high performance computing,” in Proc. 51st Hawaii Int. Conf. Syst. Sci., 2018, pp. 3894–3903.

[8] L. Wang, Y. Ma, J. Yan, V. Chang, and A. Y. Zomaya, “pipsCloud: High performance cloud computing for remote sensing big data management and processing,” Future Gener. Comput. Syst., vol. 78, no. 1, pp. 353–368, 2018.

[9] N. Zhao, V. Tarasov, H. Albahar, A. Anwar, L. Rupprecht, D. Skourtis, A. K. Paul, K. Chen, and A. R. Butt, “Large-scale analysis of Docker images and performance implications for container storage systems,” IEEE Trans. Parallel Distrib. Syst., vol. 32, no. 4, pp. 918–930, Apr. 2021.

[10] C.-Y. Liu, X. Chen, Z. Li, R. Proietti, and S. J. Ben Yoo, “SL-HyperFlex: A cognitive and flexible-bandwidth optical datacom network by self-supervised learning,” J. Opt. Commun. Netw., vol. 14, no. 2, pp. A113–A121, 2022.

[11] G. Liu, R. Proietti, M. Fariborz, P. Fotouhi, X. Xiao, and S. J. Ben Yoo, “3D-hyperflex-XLION: A flat and reconfigurable hyper-X network for datacenters,” in Proc. OSA Adv. Photon. Congr. (AP) (IPR, NP, NOMA, NV, PVLED, PSP, SPCom, SOF), 2020, Art. no. PwW1F-3.

[12] R. Proietti, Z. Liu, X. Xiao, X. Chen, and S. J. B. Yoo, “Energy-efficient and scalable data centers with flexible bandwidth SiPh all-to-all fabrics,” in Proc. Opt. Fiber Commun. Conf. (OFC), 2021, pp. 1–7.

[13] O. Robik, S. Telenyk, and E. Zhrivnak, “Management of services of a hyperconverged infrastructure using the coordinator,” in Proc. Int. Conf. Comput. Sci., Eng. Educ. Appl. Cham, Switzerland: Springer, Jan. 2018, pp. 456-467.

[14] L. Wan, A. Huebl, J. Gu, F. Poeschel, A. Gainaru, R. Wang, J. Chen, X. Liang, D. Ganushchik, T. Munson, I. Foster, J.-L. Vay, N. Podhorszki, K. Wu, and S. Klasky, “Improving I/O performance for exascale applications through online data layout reorganization,” IEEE Trans. Parallel Distrib. Syst., vol. 33, no. 4, pp. 878–890, Apr. 2022.

[15] A. Banerjee and A. Srivastava, “A cloud performance analytics framework to support online performance diagnosis and monitoring tools,” in Proc. ACM/SPEC Int. Conf. Perform. Engrg., Apr. 2019, pp. 151–158.

[16] Y. Wei, R. Ye, and X. Chen, “Oracle RAC performance analysis on VMware virtual SAN,” in Proc. IEEE/ACIS 18th Int. Conf. Comput. Inf. Sci. (ICIS), Jun. 2019, pp. 562–565.

[17] S. A. Azeeem and S. K. Sharma, “Study of converged infrastructure & hyper converges infrastructure as future of data centre,” Int. J. Adv. Res. Comput. Sci., vol. 8, no. 5, pp. 900–903, 2017.

[18] C. Melo, J. Dantas, A. Oliveira, D. Oliveira, I. Fe, J. Araujo, R. Matos, and P. Maciel, “Availability models for hyper-converged cloud computing infrastructures,” in Proc. Ann. IEEE Int. Conf. Syst. (SyCoS), Apr. 2018, pp. 1–7.

[19] R. Leite, P. Solis, and E. Alchieri, “Performance analysis of a hyperconverged infrastructure using Docker containers and GlusterFS,” in Proc. 9th Int. Conf. Cloud Comput. Services Sci., 2019, pp. 339–346.

CHEN-KUN TSUNG (Member, IEEE) received the B.Sc. and M.Sc. degrees in computer science and information engineering from Dayeh University, Changhua, Taiwan, in 2004 and 2006, respectively, and the Ph.D. degree in computer science and information engineering from the National Chung Cheng University, in July 2014. He is currently an Associate Professor of computer science and information engineering with the National Chien-Yi University of Technology, Taichung, Taiwan. His current research interests include cloud computing, big data, web-based applications, and combinatorial optimization.

JUNG-CHUN LIU received the B.S. degree in electrical engineering from the National Taiwan University, in 1990, and the M.S. and Ph.D. degrees from the Department of Electrical and Computer Engineering, The University of Texas at Austin, in 1996 and 2004, respectively. He is currently an Associate Professor with the Department of Computer Science, Tunghai University, Taiwan. His research interests include cloud computing, embedded systems, wireless networking, artificial intelligence, digital signal processing, and VLSI design.
SHU-HUA LIU received the M.Sc. degree in computer science from Tunghai University, Taichung. Her current research interests include cloud computing applications and high-speed network applications.

VINOD KUMAR VERMA was born in Kalka, Haryana, India. He received the B.Tech. degree in computer engineering from Kurukshetra University, in 2005, the M.S. degree in software systems from the BITS Pilani, Pilani, in 2008, and the Ph.D. degree in computer science and engineering from the Sant Longowal Institute of Engineering and Technology (SLIET), Longowal, India. He is currently an Assistant Professor with the Department of Computer Science and Engineering, SLIET-Deemed University. He has published many research papers in international journals of the IEEE, Springer, Elsevier, ScienceDirect, and Taylor & Francis. His current research interests include wireless sensor networks, the Internet of Things, big data, cloud computing, trust and reputation systems, simulation, distributed computing, cryptography, and software systems. He served as a member for the Organizing Review Committee, NETS 2016, Osaka, Japan; a member for the CITSA 2015 Program Committee, CITSA 2015, Orlando, FL, USA; a Program Committee Member for CITSA 2016, Orlando; an Organizing Committee Member for the World Scientific and Engineering Academy and Society, TELE-INFO, Ischia, Italy, in 2016; and a Program Committee Member for ICIS 2016, Roorkee, India. He received the Session’s Best Paper Award IMETI-CITSA-2014, Orlando; and the Best Paper Award in NCCN-11, Longowal, in 2011. He has attended international conferences at Venice, Italy, Tokyo, Japan, Orlando, USA, and Cambridge, U.K. He served as the Session Chair and an Invited Speaker for ACMOS 2016, Venice, in 2016, and the Session Co-Chair for IMETI-CITSA-2014, Orlando. He is serving as an editorial board member/reviewer for many international journals.

CHAO-TUNG YANG (Senior Member, IEEE) received the B.Sc. degree in computer science from Tunghai University, Taichung, Taiwan, in 1990, and the M.Sc. and Ph.D. degrees in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 1992 and 1996, respectively. He is currently a Lifetime Distinguished Professor of computer science with Tunghai University. He is serving on a number of international journal editorial boards. He won the Outstanding ICT Elite Award, Taiwan, in December 2018. He received the Outstanding Engineering Professor Award from the Chinese Institute of Engineers, in May 2021. He won the K. T. Li Breakthrough Award from the Institute of Information and Computing Machinery, in April 2022. He received the Outstanding Electrical Engineering Professor Award from the Chinese Institute of Electrical Engineers, in December 2022. He obtained the Outstanding Alumni Award from the Department of Computer Science, National Chiao Tung University, in April 2021, and the Outstanding Alumni Award from Tunghai University, in November 2022. His current research interests include cloud computing, big data, the IoT, and edge computing.

* * *