IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received 26 September 2025, accepted 22 October 2025, date of publication 28 October 2025, date of current version 6 November 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3625943

RESEARCH ARTICLE

Multi-Factor Replica Placement Strategy for Resource Failure Extenuation in Cloud Replication Environment

FAZLINA MOHD ALI[1], **MARIZUANA MAT DAUD**[2], **HASIMI SALLEHUDIN**[1], **(Member, IEEE), NUR ARZILAWATI MD YUNUS**[3], **(Member, IEEE), AZRAN AHMAD**[4], **AND SURYA SUMARNI HUSSEIN**[5]

[1]Center for Software Technology and Management, Faculty of Information Science and Technology (FTSM), Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor 43600, Malaysia
[2]Institute of Visual Informatics, Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor 43600, Malaysia
[3]Faculty of Computer Science and Information Technology (FSKTM), Universiti Putra Malaysia (UPM), Serdang, Selangor 43400, Malaysia
[4]College of Computing and Informatics (CCI), University Tenaga Nasional (UNITEN), Kajang 43000, Malaysia
[5]Faculty of Computer and Mathematical Sciences, University Teknologi MARA (UiTM), Shah Alam 40450, Malaysia

Corresponding author: Marizuana Mat Daud (marizuana.daud@ukm.edu.my)

This work was supported by the Universiti Kebangsaan Malaysia (UKM) Geran Galakan Penyelidik Muda (GGPM) under Grant GGPM-2023-072.

ABSTRACT Cloud replication strategies have become an evolving technology for managing the growing volume and variety of data in cloud storage. Delay in data access and data loss can affect overall system performance, which is why cloud providers often rely on replication strategies to store multiple copies of data in different locations. This approach ensures the availability of multiple data copies, swift responses, and fault resilience while maintaining costs that are affordable for both users and providers. The main challenges in cloud replication strategies are to ensure data is continuously accessible while reducing network consumption. To prevent such issues, an effective replication strategy needs to be established in a cloud environment to enhance overall performance. This study presents a novel Multi-Factor Replica Placement Strategy (MRPS) that effectively reduces the risk of resource failure in cloud replication systems. The performance of the suggested algorithm was evaluated through a comprehensive experiment using the CloudSim simulator, and its ability to choose and replicate data files for user requests was measured. On average, MRPS achieved 16% more data availability compared to DPRS and 17% compared to RS-DCSM. Meanwhile, there is a 26% improvement in network usage by MRPS, surpassing DPRS, and a 23% improvement, surpassing RS-DCSM. These results provide strong evidence of the success of the proposed MRPS and improved performance. By surmounting the problems associated with data replication in cloud environments, this research seeks to provide valuable contributions to replica placement strategies.

INDEX TERMS Data placement, replication, cloud computing, network usage, data availability.

I. INTRODUCTION

The incredible data expansion of the Internet of Everything (IoE) around the globe has driven widespread and limitless data growth [1]. Interconnected devices are extensively used in industries, and there are rising data security concerns too [2]. Consequently, to address this surge, the emergence of cloud computing has substantially changed secure data storage and management models [3]. Duplication of data forms a key role in this change, which creates and maintains multiple copies of data that are distributed across geographically distributed cloud locations, known as replication strategies [4].

The associate editor coordinating the review of this manuscript and approving it for publication was Nitin Gupta.

This process is essential for ensuring high availability, fault tolerance, and data reliability within cloud environments [5], [6].

Dynamic replication has been identified as relevant to adapting to volatile requirements in cloud terrain. Recent studies have provided insights that comprehensive dynamic replica strategies have been developed to offload storage nodes [7], [8]. Nevertheless, keeping sensitive data in multiple replicas safe and always available remains a challenge due to trade-offs in other performance metrics.

Selecting significant or popular data for the replication process, determining the number of replica copies, and placing replicas are key phases in data replication. Respective phases aim to enhance faster replication and shorter response times, decrease latency, optimize network usage, reduce storage usage, and more. In the first phase of the cloud replication environment, the important data will be selected to preserve high data availability for users. Multi-Objective Particle Swarm Optimization (MO-PSO) was developed to select a replica depending on the most requested frequencies. The proposed algorithm functions as an intelligent agent to select the most requested data by the user as replica copy candidates [9]. Response time and network usage were improved in this study. On the other hand, the replication frequency and response time were extremely high due to process overheads. Other research works were focused on selecting popular data by producing new mathematical formulas to ensure only important and most required data are copied for faster retrieval [10], [11], [12]. Still, every contribution has different goals and drawbacks.

Determining the adequate number of replica copies in cloud storage is the next phase in cloud replication. This ensures that data objects with high popularity have adequate replicas to accommodate anticipated workloads, which is imperative. In this context, [4] proposed the Online Social Network (OSN) data as a distribution strategy for multiple Cloud Service Providers (CSPs) to reduce the cost associated with an excessive number of replications [8]. An algorithm designed to prioritize the most popular data objects based on access and engagement rates. The algorithm determines the optimal number of replica copies required before assigning them to the most appropriate storage class, which is Standard Storage (SS), Infrequent Access Storage (IAS), or Reduced Redundancy Storage (RRS). Based on the access pattern, the algorithm dynamically adjusts the storage class, transitioning data to the Reduced Redundancy Storage (RRS) class when it becomes infrequently accessed over time.

The final phase in the replication process is to ensure that the required data is always available, with faster retrieval time and minimized fault tolerance, known as replica placement strategies. Many dynamic data replication management technique was introduced with the primary objective to mitigate the increased costs incurred from data transfers across data centers. However, the investigation overlooked the potential optimization through storage utilization. Researchers [13] introduced a strategy aimed at identifying optimal data placement within a multi-cloud environment. This study leverages the interval assessment technique. Notably, the study primarily concentrated on mitigating storage and network utilization costs without considering the potential benefits of incorporating storage optimization.

Among the three phases, the data placement strategies are the most prominent and appear to be the most difficult to develop to enhance the cloud replication performance. This is because determining the best storage nodes requires many factors to be considered. Some researchers would rather focus on distributing the load equally by introducing a consistency-based replication strategy [14]. Besides that, a few other studies focused on including a few factors, like analyzed data attributes, bandwidth size, user data access patterns, and storage capacities, to determine the best location for placement [15], [16]. They contributed to addressing resource bottlenecks and reducing latency.

Similarly, this study aims to establish a comprehensive multi-objective replica placement strategy to overcome many issues and challenges in a cloud replication environment. The main contributions of this research work are:

1) Discuss state-of-the-art key factors in selecting the optimal data center for replica placement in a cloud replication environment.
2) Propose a Multi-Factor Replica Placement Strategy (MRPS) to enhance performance in cloud replication, focusing on improving key aspects such as data availability and effective network usage (ENU).

The remainder of this paper is organized as follows: Section II reviews existing literature on replica placement strategies within cloud environments. Section III describes the proposed model in detail, including the methodology, measurement formulas, system architecture, and parameter selection. Section IV presents the experimental results along with a comprehensive discussion of the findings. Finally, Section V concludes the study and outlines potential directions for future research.

II. RELATED WORK

Data placement strategies in a replication environment have been proposed extensively to improve performance in the cloud. Various performance metrics have been established, such as response time, network usage, availability, fault tolerance, energy efficiency, storage optimization, and security. Both classical studies and recent findings are consolidated, breaking down into performance metrics in respective studies in this section.

A. AVAILABILITY AND RELIABILITY ENHANCEMENT

Improving availability and reliability are significant objectives in replica placement strategies. The Fuzzy Self-Defence Algorithm (FSDA) adopts a fuzzy system to evaluate parameters like service time, system availability, load, latency, energy consumption, and data center centrality. To simulate the dynamic selection of data centers, this algorithm uses predator-prey modeling to increase hit ratios and ensure system availability [17]. Another study employed fuzzy rules integrated with heuristic learning to select data centers. The algorithm named HH-ALO-Tabu measures load variance, service time, availability, and provider expenditure prior to placing replica copies [18]. Recent studies reinforce the focus on availability by using machine learning clustering to optimize HDFS replication [19]. The study result showed improved fault resilience and data availability.

B. ENERGY EFFICIENCY AND COST REDUCTION

Energy-aware strategies are critical for sustainable cloud computing. The CF-AHP model [20] optimizes data center selection using mean load variance, service time, latency, access rate, and storage usage, balancing energy use with performance. In a similar vein, [21] proposed a dual approach using AHP and ELECTRE-I cost-effective placements, reducing energy consumption and bandwidth use. Identical key factors are employed to select an appropriate location to store replica copies in this placement strategy. As energy-efficient and fault-tolerant replica management policies are equally crucial for edge-cloud environments, there are also researchers who have proposed a dynamic replica placement algorithm that demonstrated effectiveness in reducing mean job completion time, minimizing network bandwidth usage, and enhancing storage utilization [22].

Another study proposed identifying the most visited regions in space to locate the data copies. Reduced energy consumption obtained in HPSOTS algorithm which integrates Particle Swarm Optimization (PSO) and Tabu Search (TS) used to identify energy-optimal replica nodes [23]. In newer research, researchers emphasized that reducing energy usage without compromising data accessibility is achievable via hybrid optimization strategies known as bio-inspired algorithms [24].

C. RESPONSE TIME OPTIMIZATION

Rapid access to data is another key performance driver in replica placement, ensuring that data is placed in proximity to users for frequently requested files. This placement reduces response latency in distributed environments [25]. In a bio-inspired hybrid approach using MO-PSO and MO-ACO, replica placement was evaluated based on distance, host density, access frequency, output, and storage consumption, resulting in accelerated replication and improved response times [9]. Likewise, a hybrid optimization model combining PSO and Genetic Algorithms was developed by a previous researcher. Their method showed a measurable improvement in file retrieval time across distributed cloud systems [23]. Based on user demand for specific files, the RPBLB strategy replicates frequently accessed data to new storage nodes. The research objectives were met by minimizing response time during file downloads, as data is placed in the nearest available data centers. Thereby, the study successfully reduced latency and improved access efficiency [25].

An optimization-assisted frequent pattern mining for data replication in the Cloud was proposed in another study, named Greywolves Updated Exploration and Exploitation with Sealion Behaviour (GUEES). The data replication strategy aimed at enhancing response time in cloud computing environments [26]. In this study, frequently accessed data patterns were identified and replicated efficiently to improve overall data accessibility and system performance. This hybrid approach facilitates optimal edge selection during frequent pattern mining, ensuring that high-priority data is replicated effectively. The strategy also incorporates dual constraints of prioritization and cost, where data is categorized into high and low-priority queues. Based on that, storage demand evaluations guide replica placement decisions made by the algorithm. The study evaluations demonstrate that the proposed method achieves substantial improvements in response time, outperforming other algorithms. However, fault tolerance mechanisms were not incorporated in this study, which are critical for comprehensive replication management.

D. NETWORK USAGE EFFICIENCY

Network usage or bandwidth consumption is crucial for replication scalability. Hence, reducing data movement during data placement is the aim in a cloud replication environment. DPRS strategy uses file access frequency, storage distance, and storage availability average weightage to optimize network traffic via parallel downloads. This strategy improved overall network usage [27]. The BDS+ algorithm dynamically separates bandwidth allocation between real-time and bulk data transfers [28]. The BDS algorithm ensures efficient network usage by estimating traffic as the main condition prior to allocating data to the best location. Another researcher integrated network traffic estimation in real-time placement decisions, which could significantly reduce unnecessary replication and inter-data center communication [29]. However, the risk of resource failure remains unexplored in their studies.

In cloud data replication, a study presents a significant advancement in optimizing network resource utilization known as the Replication Strategy with Comprehensive Data Center Selection Method (RS-DCSM) [30]. The primary objective of RS-DCSM is to improve network usage and reduce replication frequency, which enhances data placement efficiency by minimizing unnecessary data movement across data centers. This strategy introduces a criteria-based decision-making approach that chooses the best data centers based on three key factors: Popularity, which assesses user access patterns; available storage capacity, and Centrality of each data center. RS-DCSM systematically selects optimal locations for replica placement after calculating the composite merit for each data center. This study used CloudSim as a simulation tool, and the experiment results confirmed that RS-DCSM achieves improvements in replication frequency and in network usage. Despite these improvements, RS-DCSM does not address the risk of failures, which can hinder fault tolerance in cloud replication environments.

E. STORAGE OPTIMIZATION AND SPACE MANAGEMENT

The HRS strategy focused on avoiding redundant replications in a cloud environment through fuzzy logic implementation. This is essential for storage efficiency to prioritize data center centrality and access frequency [31]. If storage is full, it intelligently deletes low-priority replicas using temporal locality and last-access patterns. Employing a similar technique, the DMDR strategy improved storage utilization by measuring centrality metrics and data access history to guide replica selection [32]. In a recent study, one of the researchers highlighted the increasing importance of replacement-aware strategies, which integrate deletion policies directly into replica placement to improve overall system throughput and storage usage [18].

F. DATA SECURITY AND PRIVACY-AWARE PLACEMENT

The LAST-HDFS placement framework allows users to define privacy policies that align with the security and privacy considerations, which are increasingly integrated into replica placement [33]. The placement framework logic groups the sensitive data by region and prevents illegal transfers through socket-level monitoring. Additionally, a study by another researcher introduced secure replication using fuzzy logic and data chunking, which reduces risk even if a server is compromised [11]. The data are chunked into a few data blocks and replicated to a number of storage devices, which prevents data loss risk. Further, a review on cloud replication emphasized that security-aware data replication is essential for regulatory compliance [34]. Therefore, using zone-based placement coupled with trust-level scoring for replica nodes is one of the strategies to comply with the regulation. Other proposed solutions may direct the securing of inter-cloud interactions for data-related use cases [35].

Literature insights show that the trend in the number of factors employed over time is practically inconsistent. It is evident that researchers have adopted these factors selectively, based on the specific objectives and contexts of their studies. Table 1 presents a comparative analysis of data placement algorithms, highlighting the number and types of factors considered in optimizing replica allocation within cloud replication environments. Fig. 1 further illustrates the distribution of these factors over the years, which have been incorporated into data placement formulations to determine optimal replica storage locations.

Collectively, the reviewed literature demonstrates a common intent to enhance key performance metrics, such as response time, availability, and resource efficiency, through strategic data placement. However, several studies have overlooked critical limitations, including issues related to fault tolerance, resource failures, storage constraints, and dynamic workload adaptation. Despite these shortcomings, the existing body of work has contributed a range of innovative approaches to replica placement, offering valuable insights into performance-driven storage allocation.

TABLE 1. Number of factors for existing research.

Algorithm	Author	Number of Placement Factors	Placement Factors
FSDA	[17]	6	Service Time, System Availability, Load, Latency, Energy Consumption, And Data Center Centrality
CF-AHP	[20]	5	Service Time, Access Rate, Latency, Load Variance, And Storage Usage
MO-ACO	[9]	5	Distance, Host Density, Access Frequency, Output, And Storage Consumption
HH-ALO-Tabu	[18]	4	Load Variance, Service Time, Availability, And Provider Expenditure
DPRS	[27]	3	File Access Frequency, Storage Distance, And Storage Availability
RS-DCSM	[30]	3	Data Popularity, Storage Availability, And Centrality
DMDR	[32]	2	Centrality Metrics And Data Access History
HRS	[31]	2	Data Center Centrality And Access Frequency
OSN	[8]	2	Access Rate And Interaction Rates
LAST-HDFS	[36]	1	Privacy Policies
BDS+	[28]	1	Traffic Estimation
HPSOTS	[23]	1	Visited Regions
RPBLB	[25]	1	Data Center Centrality
GUEES	[26]	1	Storage Demand Evaluations

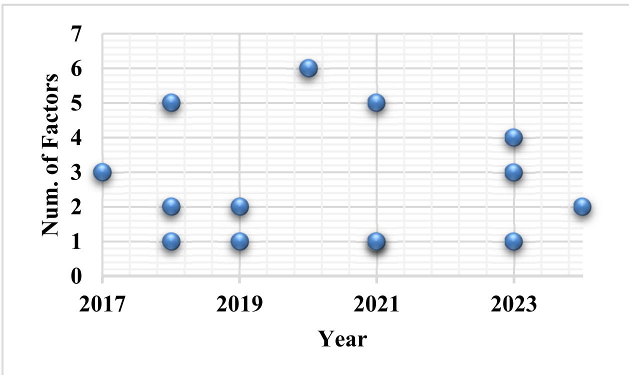

FIGURE 1. Number of factors vs years.

G. RESEARCH GAPS AND MOTIVATION

Designing a comprehensive cloud replication environment remains a challenge, particularly in mitigating the risks of node failures to prevent data loss. While past studies offer diverse strategies addressing performance improvements in replication environments, few explicitly focus on resource failure extenuation. Although computational capabilities have greatly increased, techniques for identifying and forecasting workload failures have not kept pace, indicating a crucial gap as cloud systems grow in size and complexity [37]. Furthermore, by distributing the effects of individual node failures, efficient load balancing can reduce security threats [38]. It is imperative to grasp and analyze failure rates in placement factors because nodes with higher failure rates can result in increased data unavailability and data loss, which compromises reliability. Thus, to improve overall system resilience, considering failure rates in a cloud replication environment allows for optimal replica distribution. Even though availability is frequently considered the main objective in cloud platforms, there are minimal methods to foresee and lessen resource failures. This motivates the development of a multi-factor replica placement strategy that prioritizes fault resilience with equal weight to fundamental performance goals.

III. THE PROPOSED MULTI-FACTOR REPLICA PLACEMENT STRATEGY (MRPS)

Aiming for cost efficiency, minimal storage usage, high availability, and reduced replication time, numerous studies have introduced various techniques to determine what data to replicate, how many copies to generate, and where to place them [24]. While the approach effectively addresses critical performance factors such as efficiency, resource usage, and fault tolerance, further exploration into scalability across diverse network conditions and varying workload intensities could enhance its general applicability [39]. To achieve the goals, researchers need to develop comprehensive factors in data placement strategies with an optimal recovery method grounded on availability metrics to achieve efficient data replication performance.

Acknowledging the persistent gaps, this research proposes a MRPS that integrates the concept of fault tolerance and optimization techniques to improve performance metrics such as data availability and replication efficiency. The strategy seeks to provide a more balanced and comprehensive approach to replica deployment in cloud environments.

A. MRPS PROCESS FLOW

This study proposed an MRPS algorithm that determines suitable replication locations. The MRPS considers three (3) key placement factors: the maximum storage space availability, the failure rate in the data center environment, and the highest degree of connectivity between nodes.

The MRPS outperformed existing strategies, RS-DCSM and DPRS algorithms, in terms of network usage and data availability [27], [30]. This research selected these two strategies for comparison purposes because it has an identical number of factors used to determine replica placement in the cloud, as in Table 1. Thus, the measurement results guaranteed fair and significant contributions to cloud replication performance.

In a typical cloud replication environment, there is a central unit to control and manage the entire replication system. In this MRPS strategies the central unit recognized as Central Manager (CM) and the initial task in replication process is determining the required data or files to be replicated known as *Replication File*, (RF_i) in different clusters, C_n where n is cluster index; $n = \{1, 2, \ldots n\}$. Then, CM is continuing the replication process with placement strategy in the local cluster after approves the RF_i has not existed in the user-local Cluster (C_n). CM triggers the MRPS to run placement process by calculating the *Node Score*, NSC_c^x is node index; $x = \{1, 2, \ldots w\}$ in Cluster (C_n). *Node*(N_x) in this MRPS environment is referring to data centers.

The conditions in MRPS must be obligated to fulfil the node storage selections which are, *Storage Score*(γ), *Failure Score* (γ) and *Connection Score* (₡). The CM responsible for computing respective factor scores in separate functions for every storage node, N_x in the local cluster (C_n) to get final scores. Using this strategy, the most ideal storage nodes (N_x) can be selected and the file or replica RF_x copies can be placed in appropriate locations. The best node is finalized depending on the highest rank of Node Score, NS. The Node Score, NS values are required to be normalized (between 0-1) due to its difference in scale values. The normalization is also a command technical formulation used in every computation, as the scale of computation varies [40]. The normalization method is the best alternative to transform independent values with variation, which is capable of producing standardized values across different factors.

The sub-equations for the MRPS phases and computations are presented in detail as follows.

1) The *Storage Score* (γ), calculation is to access available space in every storage Node, N_x. The minimal usage of particular storage gives a better prospect for the storage to be selected as the best location for replica placement [41]. The equation used to calculate the space merit is as in (1).

$$\gamma = \frac{\sum Av_S}{\sum Storage} in\ Node,\ N_x \quad (1)$$

$\sum$ Av$_S$ denotes the total available space in storage divided by $\sum$ *Storage* refers to total capacity allocated initially in every N_x at cluster(C_n).

2) The *Failure Score*($\mathcal{F}$) computation is required to ensure resource failures are considered before choosing the best location for replica placement. As mentioned in studies by [37] and [42], resource failures has high impact on data center efficiency and a server experiencing frequent failures is more likely to encounter additional failures in the near future.

Therefore, the failure score is calculated, and the lowest number of failure values will have a higher probability of being selected as the best replica placement node. The *Failure Score* ($\mathcal{F}$) is calculated as in (2) and (3) adapted from by [37], [42].

$$Failure\ Score, (\mathcal{F}) = \frac{1}{MTBF} \quad (2)$$

$$MTBF = \frac{(Operation\ Time * Num.of\ \ Servers)}{Num.ofSvr\ Failures} \tag{3}$$

Equation (2) states that MTBF is the abbreviation for Mean Time Before Failure, which refers to the failure rate for the data center environment. Thus, the MTBF is further obtained using (3) where *Operation Time* multiply with *Num.of Servers* in cloud data centers. Further, the calculated value is divided by *Num.of Svr Failures* in particular cloud data center environments.

1) The *Connection Score*, (₵) is computed considering the number of connections to other nodes in the dedicated local cluster, C_n [27]. *Connection Score* (₵) in *Node*, N_x is calculated as (4).

$$Connection\ Score\ (\text{₵})\ N_x = \sum NConnection \tag{4}$$

The degree of connectivity for a data center or storage nodes (N), in (4), directly refers to the connections between nodes, which presenting total number of connections between nodes, $\sum NConnection$. Many researchers advocate placing replica copies at centralized data centers to facilitate rapid data retrieval [17]. Nevertheless, the strategy presented in this study emphasizes positioning replicas at data centers that maintain robust connections to numerous other nodes. This connectivity-centric approach significantly mitigates fault tolerance issues within clusters, as it ensures alternative pathways for replica access or placement, especially during traffic congestion or downtime scenarios affecting individual data centers [43], [44]. Consequently, data centers exhibiting extensive node connectivity possess a distinct advantage in terms of information accessibility and reliability compared to those with fewer connections [45], [46].

As a final step, the MRPS accumulate individual scores, in the principal equation for Node Score *Node Score*, NSC_c^x, as in (5).

$$NodeScore, NSC_c^x = (\gamma) + (\mathcal{F}) + (\text{₵}) \tag{5}$$

The (5), the computation for all scores are completed; *Storage Score*(γ), *Failure Score* ($\mathcal{F}$) , and *Connection Score* (₵). In this study, as suggested in [27], all three factors are assumed to hold equal significance within the real cloud environment due to its high scalability. However, the weighting can be adjusted based on user preferences, allowing users to assign greater importance to specific resources by adding a constant value to any chosen factor prior to computing the final formula (5). As the three (3) score values are different in scale, the value will be normalized to a scale between 0 and 1.

Finally, the replication process will be initiated after MRPS determines the score values as in (5). The *Node Score* is calculated for respective data centers in the local requested cluster, and local manager (LM) will list the scores in descending order, then stored as *Block*1 and return the list to CM. The most fit storage node is indicated to have the top value of *Node Score*, NSC_c^x.

```
Every RF_i in each Cluster, C_n=1,2...n at every Round, n_i =1,2...i
{
CM applies MRPS to replicate RF_i; //CM = central manager
 For All C_n;
  {
   If RF_i ∃ C_n then;
     Return Replication, Rep =0;
     //Abort Placement Process
    Else
      RF_i ∄ C_n;
        Return Replication, Rep =1;
     } //Continue Placement Process
        For all RF_i { //RF= Replication File that initially determined
              For every Node (N_x) in Cluster, C_n Do;} //Initiated in Local Manager=LM
        // Count Score (S), for all Nodes N_x for selected cluster C_n
         f(γ) = ΣAv_S / ΣStorage , in Node, N_x
         f (Ϝ) = 1 / MTBF , in Node, N_x
        // MTBF = (Operation Time∗ Num.of Servers) / Num.of Svr Failures ;
         f(₵) = ΣNConnection, in Node, N_x
//LM compute Node score for every node
        Node Score, NSC_c^x = (γ) + (Ϝ) + (₵)

      Sort NSC_c^x in descending order;
      Store as Block1 ; // Block1 returned to CM
   }
 Select the high score (HS), N from Block1 ;
 Store HSNs as Block2 ; // number of HSN determined by user
 Do Replica Chunking for RF_i; //as N in Block2
//1 RF_i replication process completed
 End;
}
```

FIGURE 2. MRPS algorithm.

Then, according to using preference in their cloud replication environment the CM will select certain number of nodes, N with the highest score, HS values from $Block1_i$ and store it as $Block2_i$. N is referring to the number of nodes that enable concurrent downloads determined by users. This means that CM will start chunking the one RF_i to N number of nodes using fragmentation equation as in [30]. The equation is adopted for file fragmentation to fix file block size chunking before sending it over to N number of nodes. The detailed MRPS algorithm is shown in Fig. 2.

B. METHODOLOGY AND EVALUATION METRICS

This section discusses the evaluation metrics and methodologies used to assess the effectiveness of data placement strategies.

Simulation models are commonly categorized as static or dynamic, deterministic or stochastic, and discrete or continuous [47]. Given the nature of this research, which involves varying parameters throughout experimental runs, a discrete and continuous simulation approach was deemed most suitable for guiding the development of the simulation environment. Accordingly, this study adopts the Discrete Event Simulation (DES) methodology to model the cloud replication environment. The DES framework, adapted from Watkins, encompasses five key phases in its lifecycle: Study Definition, System Analysis, Parameter Estimation, Model Development, Experimentation, Interpretation, and Evaluation. The phases and actions in DES are employed throughout

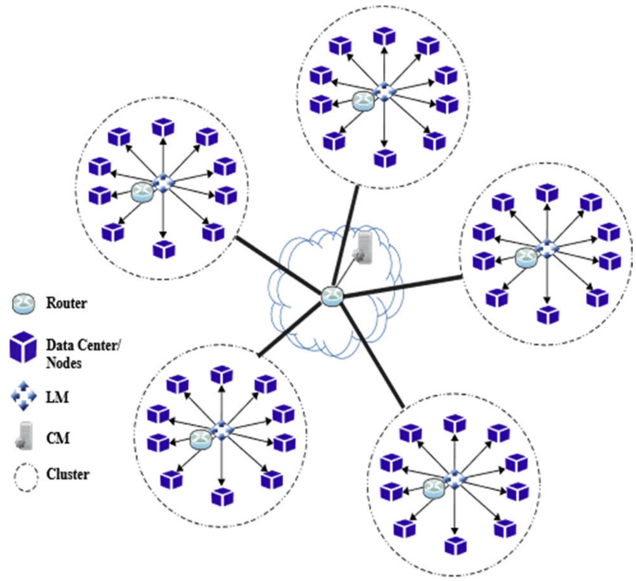

FIGURE 3. **MRPS system architecture.**

TABLE 2. **Simulation parameters.**

Parameters	Values
Total Clusters	10
Total Nodes	100
Total Servers Per Data Center	51
Nodes in clusters	10
Total Files	200
File Size	100 to 10000 (MB)
Storage Size for every Data Center	60 (GB)
Round Length	100
Nodes between 2 nodes	1
Bandwidth Inter-Router	10 (Gbps)
Bandwidth Site-to-Router	2.5 (Gbps)
Bandwidth User-to-Router	100 (Gbps)
Bandwidth CM-to-Router	2.5 (Gbps)
Bandwidth LM-to-Router	1 (Gbps)

this research work to ensure that research outcomes are relevant and significant contributions.

Countless simulation frameworks and benchmark datasets have been utilized to evaluate the performance in existing research for different algorithms. The standard performance metrics for data placement are not limited to data access latency, data availability, replication overhead, and network utilization. In this research, there are 2 main performance metrics enhanced and measured, which are effective network usage (ENU) and data availability. These two metrics were focused on proving the ability of the developed MRPS algorithm in addressing fault tolerance while preserving the data availability and proving that the network is efficiently utilized in a cloud replication environment.

1) DATA AVAILABILITY

A system may operate efficiently while ensuring that data remains accessible to users only when adequate data is available in a particular system [19]. Similarly, to evaluate data availability rate in the proposed MRPS, an experiment was conducted to assess this key performance metric. Quantitative evidence was provided to prove the capability of MRPS to deliver sufficient data accessibility to end users. The measurement of file availability at a storage node, denoted as (PSA_j) follows the formulation adapted from [7], [48] as presented in (6) and (7).

$$PSA_j = b\,(i,j) * (1 - P_j) \tag{6}$$

$$RF = 1 - \prod_{i=1}^{k} (1 - PSA_i) \tag{7}$$

In (6), the variable $b\,(i,j)$ is defined as a binary indicator where $b\,(i,j)\ =1$ if the file (i) is available in storage node (j) and $b\,(i,j)\ = 0$ if it does not exist in storage. The P_j represents the failure probability for servers in cloud data centers. Further, (7) is formulated to compute file redundancy across multiple storage nodes, where the availability of file f_j is determined by its presence across k replica copies in every storage node. The potential unavailability of files is considered as $1 - RF$. The average data availability is calculated, and the individual availability values for each file across all storage nodes are aggregated. Then, those values are normalized by the total number of files at each site. This approach provides a metric value for assessing data availability. In alignment with the methodology proposed by a researcher, who utilized both random and historical datasets to estimate node failure probabilities, this study adopts real-world failure traces [48]. The failure probability of each storage node in this study was derived using empirical data from the Google data center trace log, as reported in another research work [42]. The data refers to historical data on resource failures to mitigate unreliable sites that have been chosen for replica placement. The data are focused on simulating a resource failure in a cloud environment with a relevant number of data center settings, which are not limited to the total number of servers, data center operational period, and many other parameters, as in Table 2.

2) EFFICIENT NETWORK USAGE (ENU)

Effective Network Usage (ENU) is another performance metric used to demonstrate the proposed MRPS's capability in minimizing network resource consumption. The ENU quantifies the efficiency of network utilization during replication and data access processes [30]. Furthermore, ENU values are normalized on a scale from 0 to 1, where lower values indicate more efficient network usage, ensuring consistency and comparability.

C. SYSTEM ARCHITECTURE

The simulation environment for this research experiment was set up using CloudSim. The identical parameters were fixed in the other two (2) research works [27], [30] to ensure a fair comparison is made on performance metrics evaluations with MRPS. In this MRPS strategy, the central unit is recognized

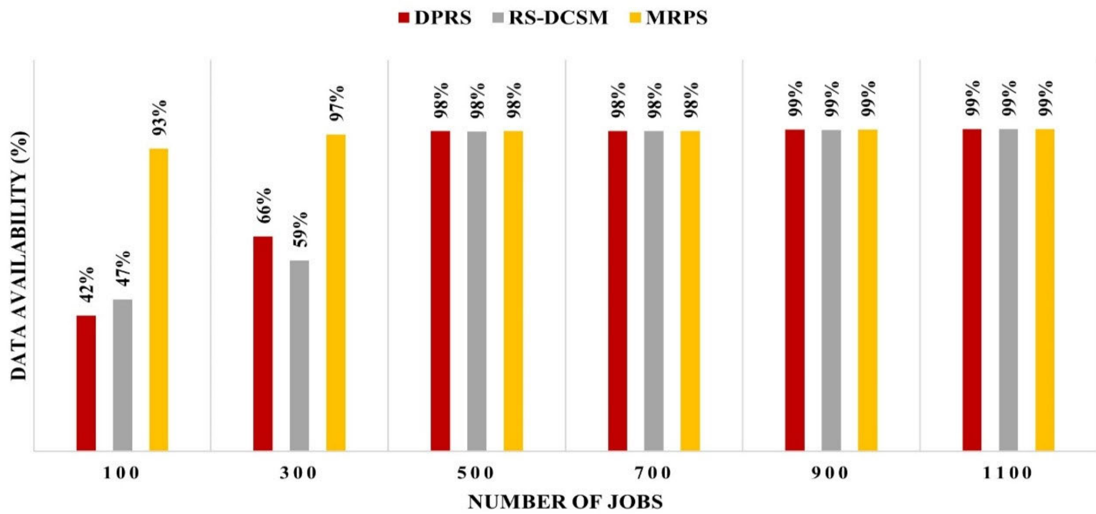

FIGURE 4. Data availability.

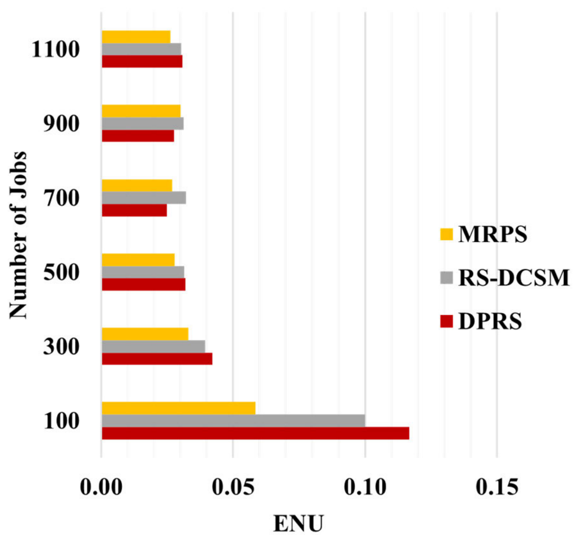

FIGURE 5. Efficient network usage (ENU).

as the Central Manager (CM) that manages the entire replication system in this system architecture. There is a Local Manager (LM) who is responsible for the respective local data clusters. A total of 10 clusters with 10 data centers are developed in the simulation environment. The data centers are also recognized as nodes connected to each other through bandwidth, with 60 Gbps of storage capacity. To ensure experimental validation in scalability and sensitivity analysis, the experiment uses varying file sizes (100–10,000 MB) and different job volumes per round (100–1100 jobs).

Fig. 3 represents the system architecture, and Table 2 is the detailed parameter settings used in the simulation environment.

IV. RESULTS AND DISCUSSION

Data accessibility by users is often associated with source device availability; thus, data availability is measured as a crucial performance metric in data replication environments [49]. A higher percentage of data availability indicates a system has vigorous consistency and its capability to provide reliable, high-performance access to stored data. Hence, data availability serves users with the required data accessible at all times, regardless of their location and time. Thus, optimizing data availability through strategic replica placement is essential for maintaining seamless data access in cloud environments [50]. Fig. 4 demonstrates the average data availability results for MRPS, RS-DCSM, and DPRS strategies.

The average data availability experiment was executed with random range file sizes between 100 and 10,000 MB. The larger scale of file size could not be tested due to the limitation of the storage capacity embedded in the cloud network topology, which only had 60GB of storage configured. Thus, a larger number of file sizes will lead to full storage and failures in the replication process in this particular cloud setting. The graphical results reveal a comparative trend between the job volume per round and the level of data availability. As job frequency increases, the system consistently achieves higher availability rates, approaching 99%, which reflects the effectiveness of the proposed replica placement approach in supporting intensive data access scenarios. This result implies that the MRPS strategies continue to provide a better percentage of average data availability, even from the beginning simulation with 100 jobs per round until 1100 jobs per round, outperforming the other two (2) RS-DCSM and DPRS algorithms. On average, MRPS improved 16% better data availability compared to DPRS and 17% compared to RS-DCSM. This graph evidenced that the MRPS with three (3)

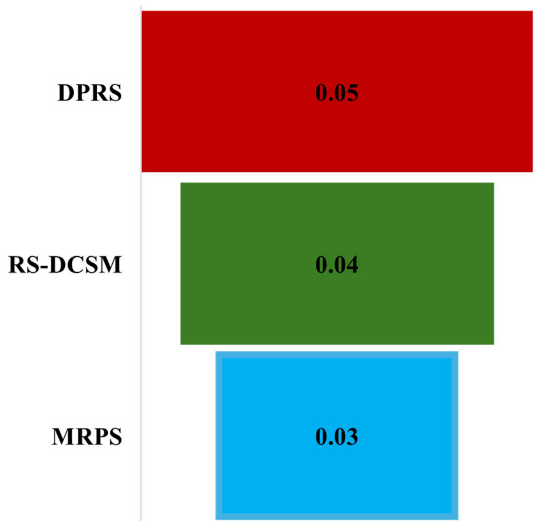

FIGURE 6. Average ENU.

score factors is capable of improving the data availability for user access in a cloud replication environment. As the batch size of jobs increases, all evaluated strategies demonstrate the ability to allocate essential replicas with comparable efficiency, thereby maintaining stable levels of data availability. In this experiment, each algorithm was re-simulated under identical conditions using a standardized failure model. As mentioned in subsection III-A, the failure data was sourced from the Google data center tracelog was simulated into this cloudsim environment, which comprises operational records from 12,532 servers over a 29-day period. During this timeframe, 8,954 server failures were recorded across 5,056 unique servers, with an average failure rate of approximately 308 servers per day. This dataset is widely recognized and deemed appropriate for modeling realistic replication scenarios in cloud environments. Simulation results were generated over 100 rounds, and the average values were computed to evaluate the performance of each strategy in terms of data availability.

The MRPS strategy demonstrated strong efficiency in generating sufficient replica copies and strategically placing them on nodes with the lowest risk of resource failure within the cluster. This targeted placement significantly contributed to improved data availability across the overall experimental evaluation. Therefore, with the resource failure risk mitigation factor ingested in this MRPS algorithm, the data availability is perceived to be more guaranteed compared to the other two algorithms.

Fig. 5 illustrates the ENU performance results obtained from simulations conducted with varying task loads, specifically at 100, 300, 500, 700, 900, and 1100 jobs per round. The experimental setup mirrored that of the two reference models, RS-DCSM and DPRS, to maintain uniformity and ensure an objective performance comparison. The bar chart displays the efficiency of network usage when handling randomly assigned file sizes, ranging between 100 MB and 10,000 MB. In every iteration, the MRPS evidenced a decrease in network usage compared to RS-DCSM and DPRS. These results reflect the system's ability to manage network resources effectively under different workload intensities.

Fig. 6 demonstrates average ENU in a bar chart showing that the MRPS strategy achieved, on average, a 26% improvement in network efficiency compared to DPRS and a 23% improvement over RS-DCSM. The MRPS achieved a value of 0.03 for the Effective Network Usage (ENU), outperforming DPRS and RS-DCSM, which recorded higher ENU values of 0.05 and 0.04, respectively. These findings provide empirical evidence for the efficacy of MRPS in minimizing network overhead by optimally placing replica copies. The greater performance of MRPS is attributed to its ability to retrieve data from storage nodes with minimal failure probability, rather than relying on random or suboptimal node selection that may result in increased network traffic or inaccessibility due to node failures.

Consequently, MRPS demonstrates consistently lower ENU values with a more holistic approach to replica placement. Conversely, DPRS and RS-DCSM were consuming more bandwidth because they neglected resource failures as one of the prominent factors, specifically considering the Mean Time before Failures (MTBF) for the resources in cloud replication environments.

V. CONCLUSION AND FUTURE DIRECTIONS

In conclusion, unlike the state-of-the-art studies, which emphasized performance enhancement while overlooking fault tolerance and resource failures, this research incorporates failure rates into replica distributions, eventually mitigating the data loss and sustaining optimal performance. Thus, the proposed MRPS for data placement strategy using multi-factor strategies in cloud replication environments demonstrates potential and promising results. Although DPRS claims to allocate replicas across all available sites, it fails to account for the inefficiencies and risks associated with resource wastage. This oversight stems from the indiscriminate placement of replicas in data centers with high failure rates, which increases the likelihood of data loss or inaccessibility when files are requested by users. Such limitations highlight the need for more strategic and failure-aware placement approaches, as addressed by the proposed MRPS. On the other hand, even the RS-DCSM considered the fault tolerance using the highest number of paths as one of their factors, yet neglected the cruciality of considering the MTBF in their factors. Therefore, the data center chosen by the DPRS and RS-DCSM method to place replicas may cause low data availability and high network usage to place and retrieve replicas. Literally, MRPS attained the advantage of placing replicas in a safe site that has a low possibility of failures to meet the demand for various file access by users. The MRPS outperformed the RS-DCSM and DPRS in the entire experiment, whereby higher chances of failures of data centers were discarded. Most required files are stored in the best data centers that preserve MRPS to access files either locally or remotely with low fault tolerance.

Despite the achievements in this study, certain limitations and gaps still persist and must be acknowledged. Primarily, the experiments and evaluations conducted to address the data placement issues relied on synthetic data due to the use of simulation tools. Therefore, perhaps inadequate datasets and configurations may not fully capture the complexity and variability of real-world cloud environments. This raises concerns regarding the scalability and applicability of the proposed approaches in practical settings. By tackling these limitations, future studies have the potential to drive meaningful progress in the domain, leading to more resilient and context-aware solutions for optimized data placement in cloud-based systems. Thus, more realistic evaluations can be performed with standardized benchmarks, considerations of data security, and multi-objective optimization in real cloud environment settings.

ACKNOWLEDGMENT

The authors sincerely acknowledge the financial support and research facilities provided, which were instrumental in the successful completion of this study.

REFERENCES

[1] F. M. Ali, N. A. Md Yunus, N. N. Mohamed, M. Mat Daud, and E. A. Sundararajan, "A systematic mapping: Exploring Internet of Everything technologies and innovations," *Symmetry*, vol. 15, no. 11, p. 1964, Oct. 2023, doi: 10.3390/sym15111964.

[2] N. H. Kamarudin, N. H. S. Suhaimi, F. A. Nor Rashid, M. N. A. Khalid, and F. Mohd Ali, "Exploring authentication paradigms in the Internet of Things: A comprehensive scoping review," *Symmetry*, vol. 16, no. 2, pp. 1–32, Feb. 2024, doi: 10.3390/sym16020171.

[3] S. S. Alnuaimi, E. A. Sundararajan, and A. H. A. Rahman, "Data distribution optimization over multi cloud storage," *J. Theor. Appl. Inf. Technol.*, vol. 100, no. 5, pp. 1378–1389, Apr. 2022.

[4] F. M. Ali, M. Mat Daud, N. Bahar, S. M. Salleh, F. A. N. Rashid, and N. A. M. Yunus, "A bibliometric analysis: AI-driven data placement optimizationin cloud replication environments," *J. Inf. Syst. Technol. Manage.*, vol. 9, no. 37, pp. 86–97, Dec. 2024, doi: 10.35631/jistm.937007.

[5] P. Elango, K. Kuppusamy, and N. Prabhu, "Data replication using data mining techniques," *Asian J. Comput. Sci. Technol.*, vol. 8, no. S1, pp. 107–109, 2021, doi: 10.51983/ajcst-2019.8.s1.1939.

[6] I. Kuraj, A. Solar-Lezama, and N. Polikarpova, "Optimizing consistency for partially replicated data stores," in *Proc. 27th ACM SIGPLAN Symp. Princ. Pract. Parallel Program.*, Apr. 2022, pp. 457–458, doi: 10.1145/3503221.3508438.

[7] M. A. Fazlina, R. Latip, M. A. Alrshah, and S. Member, "Vigorous replication strategy with balanced quorum for optimizing the storage and response time in cloud environments," pp. 1–13, 2021.

[8] A. Y. Aldailamy, A. Muhammed, R. Latip, N. A. W. A. Hamid, and W. Ismail, "Online dynamic replication and placement algorithms for cost optimization of online social networks in two-tier multi-cloud," *J. Netw. Comput. Appl.*, vol. 224, Apr. 2024, Art. no. 103827, doi: 10.1016/j.jnca.2024.103827.

[9] A. Awad, R. Salem, H. Abdelkader, and M. A. Salam, "A novel intelligent approach for dynamic data replication in cloud environment," *IEEE Access*, vol. 9, pp. 40240–40254, 2021, doi: 10.1109/ACCESS.2021.3064917.

[10] N. Mansouri and M. M. Javidi, "A review of data replication based on meta-heuristics approach in cloud computing and data grid," *Soft Comput.*, vol. 24, no. 19, pp. 14503–14530, Oct. 2020, doi: 10.1007/s00500-020-04802-1.

[11] N. Mansouri, M. M. Javidi, and B. Mohammad Hasani Zade, "A CSO-based approach for secure data replication in cloud computing environment," *J. Supercomput.*, vol. 77, no. 6, pp. 5882–5933, Jun. 2021, doi: 10.1007/s11227-020-03497-3.

[12] S. Nannai John and T. T. Mirnalinee, "A novel dynamic data replication strategy to improve access efficiency of cloud storage," *Inf. Syst. e-Business Manage.*, vol. 18, no. 3, pp. 405–426, Sep. 2020, doi: 10.1007/s10257-019-00422-x.

[13] P. Wang, Z. Chen, M. Zhou, Z. Zhang, A. Abusorrah, and A. C. Ammari, "Cost-effective and latency-minimized data placement strategy for spatial crowdsourcing in multi-cloud environment," *IEEE Trans. Cloud Comput.*, vol. 11, no. 1, pp. 868–878, Jan. 2023, doi: 10.1109/TCC.2021.3119862.

[14] A. Noraziah, A. A. C. Fauzi, S. H. S. A. Ubaidillah, B. Alkazemi, and J. B. Odili, "BVAGQ-AR for fragmented database replication management," *IEEE Access*, vol. 9, pp. 56168–56177, 2021, doi: 10.1109/ACCESS.2021.3065944.

[15] C. Li, M. Song, M. Zhang, and Y. Luo, "Effective replica management for improving reliability and availability in edge-cloud computing environment," *J. Parallel Distrib. Comput.*, vol. 143, pp. 107–128, Sep. 2020, doi: 10.1016/j.jpdc.2020.04.012.

[16] J. Zhou, Y. Chen, W. Xie, D. Dai, S. He, and W. Wang, "PRS: A pattern-directed replication scheme for heterogeneous object-based storage," *IEEE Trans. Comput.*, vol. 69, no. 4, pp. 591–605, Apr. 2020, doi: 10.1109/TC.2019.2954089.

[17] N. Mansouri, B. M. H. Zade, and M. M. Javidi, "A multi-objective optimized replication using fuzzy based self-defense algorithm for cloud computing," *J. Netw. Comput. Appl.*, vol. 171, Dec. 2020, Art. no. 102811, doi: 10.1016/j.jnca.2020.102811.

[18] B. Mohammad Hasani Zade, N. Mansouri, and M. M. Javidi, "A new hyper-heuristic based on ant lion optimizer and Tabu search algorithm for replica management in cloud environment," *Comput. Intell.*, vol. 56, no. 9, pp. 9837–9947, Sep. 2023, doi: 10.1007/s10462-022-10309-y.

[19] M. A. Ahmed, M. H. Khafagy, M. E. Shaheen, and M. R. Kaseb, "Dynamic replication policy on HDFS based on machine learning clustering," *IEEE Access*, vol. 11, pp. 18551–18559, 2023, doi: 10.1109/ACCESS.2023.3247190.

[20] M. R. Djebbara and H. Belbachir, "Cost function based on analytic hierarchy process for data replication strategy in cloud environment," *J. Theor. Appl. Inf. Technol.*, vol. 99, no. 9, pp. 2638–2648, Jun. 2018.

[21] M. R. Djebbara and H. Belbachir, "Cloud data replication strategy using multiple-criteria decision analysis methods," *Multiagent Grid Syst.*, vol. 14, no. 2, pp. 203–218, Jun. 2018, doi: 10.3233/mgs-180288.

[22] C. Li, Y. Wang, Y. Chen, and Y. Luo, "Energy-efficient fault-tolerant replica management policy with deadline and budget constraints in edge-cloud environment," *J. Netw. Comput. Appl.*, vol. 143, pp. 152–166, Oct. 2019, doi: 10.1016/j.jnca.2019.04.018.

[23] Y. Ebadi and N. J. Navimipour, "An energy-aware method for data replication in the cloud environments using a Tabu search and particle swarm optimization algorithm," *Concurrency Comput., Pract. Exper.*, vol. 31, no. 1, p. 4757, Jan. 2019, doi: 10.1002/cpe.4757.

[24] F. Karamimirazizi, S. M. Jameii, and A. M. Rahmani, "Data replication methods in cloud, fog, and edge computing: A systematic literature review," *Wireless Pers. Commun.*, vol. 135, no. 1, pp. 531–561, Mar. 2024, doi: 10.1007/s11277-024-11082-7.

[25] X. Fu, J. Li, W. Liu, S. Deng, and J. Wang, "Data replica placement policy based on load balance in cloud storage system," in *Proc. IEEE 3rd Inf. Technol. Netw., Electron. Autom. Control Conf. (ITNEC)*, Apr. 2019, pp. 682–685, doi: 10.1109/ITNEC.2019.8728995.

[26] D. Rambabu and A. Govardhan, "Optimization assisted frequent pattern mining for data replication in cloud: Combining sealion and grey wolf algorithm," *Adv. Eng. Softw.*, vol. 176, Feb. 2023, Art. no. 103401, doi: 10.1016/j.advengsoft.2022.103401.

[27] N. Mansouri, M. K. Rafsanjani, and M. M. Javidi, "DPRS: A dynamic popularity aware replication strategy with parallel download scheme in cloud environments," *Simul. Model. Pract. Theory*, vol. 77, pp. 177–196, Sep. 2017, doi: 10.1016/j.simpat.2017.06.001.

[28] Y. Zhang, X. Nie, J. Jiang, W. Wang, K. Xu, Y. Zhao, M. J. Reed, K. Chen, H. Wang, and G. Yao, "BDS+: An inter-datacenter data replication system with dynamic bandwidth separation," *IEEE/ACM Trans. Netw.*, vol. 29, no. 2, pp. 918–934, Apr. 2021, doi: 10.1109/TNET.2021.3054924.

[29] D. Rambabu and A. Govardhan, "Survey on data replication in cloud systems," *Web Intell.*, vol. 22, no. 1, pp. 83–109, Mar. 2024, doi: 10.3233/web-230087.

[30] M. A. Fazlina, R. Latip, H. Ibrahim, and A. Abdullah, ''Replication strategy with comprehensive data center selection method in cloud environments,'' *Comput., Mater. Continua*, vol. 74, no. 1, pp. 415–433, 2023, doi: 10.32604/cmc.2023.020764.
[31] N. Mansouri and M. M. Javidi, ''A hybrid data replication strategy with fuzzy-based deletion for heterogeneous cloud data centers,'' *J. Supercomput.*, vol. 74, no. 10, pp. 5349–5372, Oct. 2018, doi: 10.1007/s11227-018-2427-1.
[32] N. Mansouri, M. M. Javidi, and B. M. H. Zade, ''Using data mining techniques to improve replica management in cloud environment,'' *Soft Comput.*, vol. 24, no. 10, pp. 7335–7360, May 2020, doi: 10.1007/s00500-019-04357-w.
[33] A. Atrey, G. Van Seghbroeck, H. Mora, B. Volckaert, and F. De Turck, ''UnifyDR: A generic framework for unifying data and replica placement,'' *IEEE Access*, vol. 8, pp. 216894–216910, 2020, doi: 10.1109/ACCESS.2020.3041670.
[34] A. Folorunso, O. Babalola, C. Edgar Nwatu, and A. Adedoyin, ''A comprehensive model for ensuring data compliance in cloud computing environment,'' *World J. Adv. Res. Rev.*, vol. 24, no. 2, pp. 1983–1995, Nov. 2024, doi: 10.30574/wjarr.2024.24.2.3514.
[35] S. I. Akhtar, A. Rauf, M. F. Amjad, and I. Batool, ''Inter-cloud data security framework to build trust based on compliance with controls,'' *IET Inf. Secur.*, vol. 2024, no. 1, pp. 1–19, 2024, doi: 10.1049/2024/6565102.
[36] A. Bowers, C. Liao, D. Steiert, D. Lin, A. Squicciarini, and A. Hurson, ''Detecting suspicious file migration or replication in the cloud,'' *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 1, pp. 296–309, Jan. 2021.
[37] J. Li, R. Wang, G. Ali, T. Dang, A. Sill, and Y. Chen, ''Workload failure prediction for data centers,'' in *Proc. IEEE Int. Conf. Cloud Comput.*, May 2023, pp. 479–485, doi: 10.1109/CLOUD60044.2023.00064.
[38] A. Ganesh, A. K. Sekar, R. Ramakrishnan, and J. Logeshwaran, ''A load balancing architecture to improve the security of cloud computing in the disease management centers,'' in *Proc. 2nd Int. Conf. Smart Technol. Smart Nation (SmartTechCon)*, Aug. 2023, pp. 1299–1305, doi: 10.1109/SmartTechCon57526.2023.10391528.
[39] F. M. Ali, M. M. Daud, F. A. N. Rashid, N. H. Kamarudin, S. M. Salleh, and N. A. M. Yunus, ''A taxonomic study: Data placement strategies in cloud replication environments,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 11, pp. 825–840, 2024, doi: 10.14569/ijacsa.2024.0151182.
[40] I. Z. Mukhametzyanov, ''Rank reversal in MCDM models: Contribution of the normalization,'' in *Normalization of Multidimensional Data for Multi-Criteria Decision Making Problems: Inversion, Displacement, Asymmetry*. Cham, Switzerland: Springer, 2023, pp. 111–127, doi: 10.1007/978-3-031-33837-3_6.
[41] N.-E. Bakni and I. Assayad, ''Smart data placement strategy in heterogeneous Hadoop,'' *HighTech Innov. J.*, vol. 6, no. 1, pp. 42–53, Mar. 2025, doi: 10.28991/hij-2025-06-01-03.
[42] X. Li, X. Jiang, P. Garraghan, and Z. Wu, ''Holistic energy and failure aware workload scheduling in cloud datacenters,'' *Future Gener. Comput. Syst.*, vol. 78, pp. 887–900, Jan. 2018, doi: 10.1016/j.future.2017.07.044.
[43] A. Singh, ''Comparative study of concurrency and replica control protocols in distributed environment,'' *IJAER*, vol. 14, no. 2, pp. 329–334, 2019. [Online]. Available: http://www.ripublication.com
[44] N. Mansouri, R. Ghafari, and B. M. H. Zade, ''Cloud computing simulators: A comprehensive review,'' *Simul. Model. Pract. Theory*, vol. 104, Nov. 2020, Art. no. 102144, doi: 10.1016/j.simpat.2020.102144.
[45] S. Slimani, T. Hamrouni, and F. Ben Charrada, ''Service-oriented replication strategies for improving quality-of-service in cloud computing: A survey,'' *Cluster Comput.*, vol. 24, no. 1, pp. 361–392, Mar. 2021, doi: 10.1007/s10586-020-03108-z.
[46] D. Y. Kenett, M. Perc, and S. Boccaletti, ''Networks of networks—An introduction,'' *Chaos, Solitons Fractals*, vol. 80, pp. 1–6, Nov. 2015, doi: 10.1016/j.chaos.2015.03.016.
[47] K. Watkins, *Discrete Event Simulation in C*. New York, NY, USA: McGraw-Hill, 1993.
[48] N. Mansouri and M. M. Javidi, ''A new prefetching-aware data replication to decrease access latency in cloud environment,'' *J. Syst. Softw.*, vol. 144, pp. 197–215, Oct. 2018, doi: 10.1016/j.jss.2018.05.027.
[49] R. W. A. Fazul and P. P. Barcelos, ''The HDFS replica placement policies: A comparative experimental investigation,'' in *Proc. IFIP Int. Conf. Distrib. Appl. Interoperable Syst.*, Lucca, Italy, Jun. 2022, pp. 151–166, doi: 10.1007/978-3-031-16092-9_10.
[50] D. S. Thavamani, ''A study on dynamic replication techniques with AI in cloud computing environments,'' *Int. J. Sci. Res. Eng. Manag.*, vol. 9, no. 1, pp. 1–9, Jan. 2025, doi: 10.55041/ijsrem40982.

FAZLINA MOHD ALI received the bachelor's degree in computer science (Networking) from Universiti Putra Malaysia (UPM), the master's degree in science (information technology) from UiTM, Shah Alam, and the Ph.D. degree in parallel and distributed computing from UPM. She is a Senior Lecturer with the Faculty of Information Science and Technology, affiliated with the Center for Software Technology and Management (SOFTAM), National University of Malaysia (UKM). She brings over 15 years of experience in information systems. She has served as the Senior IT Director with the Malaysian government prior to transitioning into academia. Her research areas of focus encompass parallel and distributed computing, data management, data replication, cloud computing, disaster recovery management, big data, ICT project management, and information system management.

MARIZUANA MAT DAUD received the Ph.D. degree in computer vision with deep learning from the Universiti Kebangsaan Malaysia (UKM). Her Ph.D. dissertation focused on eye disease detection using computer vision. She is a proactive Researcher with nine years of experience in the field of computer vision, imaging, and machine learning. She holds a keen interest in the practical applications of imaging, which has driven her to explore the intricacies of robotic operating systems. Her research aims to expedite advancements in computer vision, machine learning, and artificial intelligence, making significant contributions to various fields. Currently, she is with the Institute Visual of Informatics, UKM, as a Senior Lecturer. She has gained valuable industry experience working with MIMOS, Malaysia's national applied research and development center. During her tenure with MIMOS, she was involved in several projects that leveraged cutting-edge technologies to solve real-world problems, further honing her expertise in her areas of interest.

HASIMI SALLEHUDIN (Member, IEEE) received the bachelor's degree in computer engineering from the Universiti Teknologi Malaysia, the master's degree in computer science from the University of Malaya, and the Ph.D. degree from the Universiti Malaysia Kelantan. He is a Senior Lecturer with the Faculty of Information Science and Technology and the Deputy Director (Strategic and Development) with CRIM, Universiti Kebangsaan Malaysia (UKM). He has over 20 years of experience in academia, public service, and the defense sector. He has led national projects, including MyGDX DRP, MySUKSelCloud, Digital Twins, and Metaverse Banking, with over 70 indexed publications. His research focuses on information systems, cybersecurity, cloud computing, the IoT, green software engineering, and health informatics.

NUR ARZILAWATI MD YUNUS (Member, IEEE) received the Bachelor of Computer Science, M.Sc., and Ph.D. degrees in computer network from the Universiti Putra Malaysia. She is currently a Senior Lecturer with the Universiti Putra Malaysia. Her research interests include network routing, optical switching, interconnection networks, parallel computing, 6G networks, the Internet of Things, cybersecurity, and machine learning. She is a member of the International Association of Computer Science and Information Technology Press (IACSIT).

AZRAN AHMAD received the Diploma degree in information technology from the Universiti Sultan Zainal Abidin (UNISZA), in 2002, the Bachelor of Science degree (Hons.) in computer science from the Universiti Kebangsaan Malaysia (UKM), in 2008, and the Master of Science and Doctor of Philosophy degrees in information technology from the Universiti Teknologi MARA (UiTM), in 2014 and 2021, respectively. He is a Senior Lecturer with the College of Computing and Informatics (CCI), Universiti Tenaga Nasional (UNITEN). His research interests include software engineering, software quality and digital competency, reflecting his commitment to advancing knowledge and practice in these critical areas of computing.

SURYA SUMARNI HUSSEIN received the Ph.D. degree in information systems from the Universiti Teknologi Malaysia. She is a Senior Lecturer with the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM), Malaysia. She has over 20 years of experience in the field of information systems, with a strong focus on enterprise architecture, ICT strategic planning, and digital transformation in the public sector. She has contributed extensively to national ICT initiatives, including the Malaysian Government Enterprise Architecture (MyGovEA) and various digital strategic plans for government agencies. She is a certified professional in TOGAF 9.1, COBIT 5, ITIL, CPRE, and CTFL, and actively engages in research and consultancy projects related to digital governance and enterprise systems.