AI - Based Framework for Private Cloud Computing

DHAYA R (dhayavel2005@gmail.com)
King Khalid University https://orcid.org/0000-0002-3599-7272
Kanthavel R
King Khalid University

Research Article

Keywords: Artificial Intelligence, private Cloud Storage, efficiency , profitability, complexity

Posted Date: September 21st, 2021

DOI: https://doi.org/10.21203/rs.3.rs-913251/v1

Abstract

Artificial Intelligence (AI) systems are computational simulations that are "trained" using information and expert input to duplicate a professional's choice given the same data. Only using one private cloud storage service to store information can cause a variety of issues for the system administrator. Knowledge providers, scalability, efficiency, privacy, and the potential of vendor support are examples of such concerns. Distributing information across several cloud storage services, comparable to how data is dispersed between various physical disk drives to increase error detection and increase productivity, is a possible approach. Moreover, because multiple private cloud providers have varying pricing strategies and service quality, maximizing the efficiency and profitability of many cloud providers at the same time is difficult. Based on access permission behaviors, this study presents a methodology for dynamically modifying network management rules across several cloud providers. The goal of this research is to look into how to reduce both the estimated cost and delay periods for numerous cloud providers. The architecture was put into practice in a cloud storage systems emulator, which simulated the complexity and effectiveness of numerous cloud providers in a real-world context. In particular, the architecture was evaluated in a variety of cloud storage environments. The outcomes of the platform's testing revealed that many cloud methods were successful.

1. Introduction

Artificial intelligence (AI) is currently the hottest trend in the realm of innovation; yet, strategies for harnessing its full potential in commerce and business are still being developed. The next major task for AI is to transform database management across enterprises, either on-premises or in the cloud. Computing as a model for supplying cost-effective computer complexity to consumers over the Internet [1]. The adaptability and mobility of capabilities in reaction to workload spikes and drops are some of the key benefits of internet marketing. It's also simple to use and manage, and customers just pay for what they use. Cloud providers, in general, supply computer complexity to users via a software platform. The three major offerings are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), each of which includes a multitude of choices [2]. This research solely looks at one type of IaaS solution, specifically cloud storage. Cloud storage also enables businesses to better handle their growing processing power requirements in terms of actual memory, which can skyrocket over time. Rather than just focusing on a single storage service, this principle allows data from redundancy arrays of independent disks (RAID) to be distributed over numerous cloud providers[3].

By dividing data across many storage devices, RAID was already utilized in conjunction with significant to prevent issues by using a solitary memory card. The primary priorities of RAID are to enhance the productivity of learning from and composing to storage devices, as well as to include some level of high availability if one of the storage devices refuses [4]. Enhancing the expense and throughput time of various cloud providers is complicated owing to variations in service productivity and selling prices among cloud services. Furthermore, any cloud provider's pricing and functionality are not constant. Depending on how much data is kept or moved over the network, the fee may fluctuate instantly [5]. As a result, instead of only reviewing the overall condition, the improvement must be considered for lengthy price and efficiency. As a result, finding a dynamic solution effective at effectively solving and lag time while also responding to changes in the status of multiple cloud service providers is critical [6] .

2. Literature Review

This section concentrates on the various research papers which have already undergone deep studies by both theoretical and practical approach to in Artificial Intelligence using Cloud storage systems, integration concepts, challenges, and new directions.

The E-Pareto Active Learning algorithm, proposed by Zuluaga et al (2016), optimally surveys the model structure and predicts a collection of non-dominated explanations that span the genuine design procedure with certain resolutions controlled by settings. Papaioannou et al. (2012) developed a cloud broking system that continually changes study areas among different cloud service providers based on document consumption metrics to reduce storage costs, promote better dependability, and prevent built-in security risks. The investigation, meanwhile, somehow doesn't assess the platform's influence on latency time. Xu et al. (2012) proposed URL, an integrated technique for automating the deployment operations of hybrid computers and modules executed within them. The application service providers themselves the use of data center auto-configuration in real-time. It also enabled quality of service certification by adapting the VM source allocation and equipment constraint values to cloud characteristics and increasing workloads. The method, in particular, allows for a suitable trade-off between framework usage targets and electronic SLA optimization goals. Experiments on Xen VMs with diverse workloads showed that the method is efficient.Voas and Jeffrey (2009) expressed their perspectives on cloud technology, allowing the audience to make their own decisions. For analyzing expressive optimizing goals and strategies in dynamical environments, Zhou et al (2015) proposed a new statistical objective function. Deco utilized the capability of GPUs to locate the solutions rapidly and efficiently with several alternative authorization issues. Deco could accomplish better cost-effective functionality improvements than specifying the techniques by integrating them into a prominent process management solution.

The Department of Defense's artificially intelligent (AI) method calls for the development of transcendent and disturbing functionality that will influence the "identity of the potential frontlines and the speed of risks" that US pressures will face. Candidate frames must also highlight potential purpose areas while allowing for collaboration with the business sector, academics, and closest partners. An adaptable, inexpensive, and accessible computer architecture that includes cutting-edge technology and conforms to strong network security regulations is required to tackle these difficulties. J. Robertson et al. (2021) offered a dynamic way to solve system dynamics problems using cloud services. Tanja Hagemann and Katerina Katsarou (2020) defined three major theoretical domains (artificial intelligence, deep learning, and inferential statistics) and summarized how the respective models are utilized for image classification. Additionally, which particular application sectors are commonly handled by outlier detection in virtualization, including which various governmental databases are frequently used for planning, were clarified.

Lin et al. (2019) suggested the Cloud Capability Planning Tool, a structure that consists of a Relocation Type Comprehensive planning using a unique interactive teaching framework that meets "concept drift," and an AI Coordinator that creates plans from the easily searchable sphere and trouble files with unambiguous requirements such as goal jurisdictions and relevant information in customer input formats. On a real-world migratory task, a set of investigations have been carried out. Vengerov (2008) introduced an optimization algorithm (RL) system for continuously modifying file migrating policies to tackle external demand responsiveness. The migration rules optimized by RL were evaluated using a multi-tier storage system simulator, and such strategies were demonstrated to produce a considerably improved concert above the finest handmade strategies. Hado van Hasselt et al. (2007) introduced the continuous agent criticism feedback automaton, a novel category of techniques that can accommodate continuous variables and interactions. The ordinarily applied is simple to put into practice. This technique is designed to be used against other techniques in an experimental environment. Tanimoto et al. (2013) developed a cloud-based organizational information management system. The organizational breakdown structure (WBS) approach was used to organize business information into 28 sections. The partitioned material was then categorized by whether it contained confidential data. Only roughly 18 percent of the data had to be stored in a private cloud, while the rest 82 percent had to be stored in a public cloud. A core application development concept for the cloud-based setup is carried out in compliance with this reinforce positive. For cloud computing, M. R. Uddin et al. (2019) suggested an extremely developed two-step secure communication. AES is responsible for the first stratum's first strategically placed. When it comes to the number of iterations to be completed, the AES application's operation is determined by the computation key size. For textual cryptography and stream cipher decoding, a MATLAB code was created. Investigations were carried out to determine the response time. An ANN is a naturally motivated computational approach. These merely appear to be simultaneous computations conducted by the physiological cognitive system, which are the foundations of cognitive behavior. ANN is used to develop iris and finger recognition in MATLAB.

Khatib et al. (2019) investigated the use of cloud computing and artificial intelligence in the Middle East's leading telecom, MGA-MENA, corporation. The implementation resulted in a boost in services provided, manufacturing efficiency, quality services, and more consumer assistance for the Smart MGA-MENA Company. The conclusion reached was that cloud computing and artificial intelligence are a new and sophisticated economic possibility for large companies listed like MGA-MENA, which have a massive client base and hundreds of transactions each minute. As a result, it is recommended that telecoms be technologically active and maintained. The platform is built on machine learning, a computational intelligence technique for expressing complicated nonlinear equations (Whiteson, 2012). It has a lot of uses in machine learning. A feed-forward network is the most fundamental basis of a neural network. The training algorithm (BP) approach is generally used in neural network implementations to modify link connections using the conjugate gradient approach. Bowers et al., 2009 used the Confirmation of Attribute-based encryption (PORs) method to redistribute files over a variety of data storages to improve data access and proactively monitor data security issues in the cloud using the RAID principles. To minimize vendor lock-in and affect the reliability and reliability of large datasets, Bessani et al. (2011)

proposed DepSky, a device that includes a cryptographic steganographic technique with erased codes. Even though DepSky improved functionality, the cost increased on average as compared to single online storage. Zhou et al., 2015 designed a method dubbed "Deco" to reduce costs while maintaining adequate performance. However, they built their system to distribute processing tasks among several machines (not data) in a cloud network for task scheduling. Furthermore, their system seeks to plan for connectivity and expenses.

From the literature survey, it is seen that the existing papers detection of ejections on the in artificial intelligence with cloud computing techniques, but with a lack of clarity in the data retrieval part for processing and further utilization in terms of accuracy.

3. Ai- Based Framework Architecture For Private Cloud

The scope of this work is to use AI to disseminate and compress records among numerous private cloud data providers. Management of AI-based File Deployment among Various Private Cloud Storage Services is the term of the AI-based structure we recommend [7]. Focusing on AI-based file access characteristics, the AI framework enabled distribution ns among previous cloud storage services from the customer side [8]. The AI-based framework's major goal is to maximize both quality and cost variables. Storage, network bandwidth, and maintenance are all included in the total cost of this approach. The distribution of files between both the private cloud customer and the private cloud provider is efficiency element (latency time).The data quality and reliability of the system will be enhanced by distributing AI data among different private cloud storage services, while vendor support is avoided [9]. Two AI approaches are included in the proposed model, as shown in Fig. 1:

Supervised Learning (SL)

The Mechanism for Predicting Sequence Availability (MPSA) is supervised learning that estimates the structure information for each file. The system uses a regression model to forecast each document's structure information [10].

Reinforcement Learning (RL)

RL chooses how resources are distributed across many private clouds. This selection should be based on the billing practices and efficiency of private cloud storage providers, as well as file and directory behaviors. The Reinforcement Learning (RL) method was taught using an Artificial Neural Network (ANN) to optimize the efficiency and profitability of private cloud storage services over time. This learned method incorporates a novel approach to converting the merit of each condition into a particular effect [11].

The MPSA analyses the file properties and forecasts access permission patterns in addition to the file's long-term viability. The MPSA then sends the accessibility to files Sequence Availability to the RL system, which manages the distribution policies using an ANN based on RAID technology methodology [12]. The receiver divides every file into different sizes across numerous private cloud storage services using reinforcement learning outcomes. The retailer's goal is to place a varied percentage of each file on each private cloud storage service.

Mechanism for Predicting Sequence Availability (MPSA)

The Mechanism for Predicting Sequence Availability describes how a file behaves over the course of its existence. The pattern specifies how many instances the file will be accessed or modified throughout the course of its life [13]. Recognizing how a file behaves is critical for lowering the cost of private resources for cloud storage. For most private cloud computing, the cost of storage solutions is determined by

- The volume of data retained

- Consumed amount of bandwidth utilization

- Number of functions

As a result, lowering the cost of private cloud storage services entails predicting 'quantitative' file application data as well as their longevity. The Mechanism for Predicting Sequence Availability (MPSA) establishes links between a file's properties and its Sequence Availability values. The following are the attributes:

Lifetime

The length of time the file is alive is referred to as its lifetime. Even though the term lifespan or longevity refers to the duration between establishing and discarding a file, only the productivity of each file was considered in this study [14] .

Frequency of reads

This indicates how many instances the file will be accessed throughout the course of its existence.

Frequency of Writes

The number of times the file will be modified during its lifespan is referred to as the frequency of writes.

The MPSA was developed using iterative algorithms for multiple functional platforms at a big retailer. The datasets were created using the filing process specified in the preceding section for holidays, employment, financial analysis, investigation, and marketing strategy. The following structure properties were included in the log file: the file's administrative user, the file customer's status, the file's creation date, the data format, the file type, and the file section.

MPSA Evaluation

There are three useful functions that were used to evaluate the regression effectiveness to assess the effectiveness of the MPSA prediction. These were the processes

The Pearson's r

Root Means Squared Error, denoted by RMSE.

Mean Absolute Error, denoted by MAE.

The Pearson correlation coefficient, abbreviated as r, is a determination of the stiffness of a relationship involving multiple parameters. Both variables should be regularly spread for the Pearson r correlation. Data points can have a big impact on the best fit and the Pearson correlation coefficient, which is why Pearson's correlation coefficient, r, is so reactive to them. This means that incorporating outliers into the investigation can generate erroneous **conclusions.** Every variable should have a continuous value. Pearson's correlation coefficient criteria

-The measuring scale should be either interval or ratio.

-The parameters should be generally regularly dispersed.

-The relationship should be continuous.

Root Mean Square Error (RMSE) is a normal method to calculate the fault of a representation in forecasting quantitative statistics.

$$\mathbf{RMSE} = \sqrt{\sum_{i=1}^{n}}(\boldsymbol{ki} - \boldsymbol{yi})\mathbf{2}/\boldsymbol{q}$$

K1,k2 ......are predicted values

Y1,y2 ... are observed values

q is the quantity of observations

Mean absolute error (MAE) is a regression loss function. The loss is the arithmetic average disparities among real and projected quantities, or, to put it another way, the given equations:

$$L(y, \hat{y}) = \frac{1}{N} \sum_{i=0}^{N} |y - \hat{y}_i|$$

4. Effective Ai Architecture For File Distribution Enhancement Among Private Cloud Storage Services (Eaifde)

In many other RL systems that use an ANN as a dimensionality reduction, the state vectors are fed into the ANN as insight. Every cloud storage service was vulnerable to disruptions at some period in history. As a result, a strategy that was somewhat adaptable to variations in the quantity of cloud storage services was required. Figure 2 illustrates a new strategy for an ANN with reinforcement learning proposed in this paper.

There are three cloud storage facilities, each with their own spatial domain, and the purpose of this new concept is to permit RL to deal with every cloud individually to yield various transfer functions. The MPSA's condition features are fed into the ANN, which offers the production principles. Every one of them correlates to a separate cloud. Each of these characteristics could be used to determine the current situation of the file in the cloud. Following that, each output value is turned into a precise dimension action, taking into account the value of other activities, as shown in Fig. 2.

(ai)t = P (yi)t / ∀p (p)

- Where i indicates each cloud service provider's alphanumeric code, and p is the state value of the data center numeral i.

This model enables the RL program to assign more of each file to the cloud with the highest situation significance. The reinforcement learning server takes a distinct response from each scenario after completing all activities. The numerous failure formulas are computed using these incentives [15]. The procedure for calculating the objective functions was one of the most challenging issues in this work. Algorithm 2 involves the method. Cost and delay are two unlimited parameters that influence the total score. The proportion of the present latency point to the utmost latency period was calculated to maximize the latency time. To lower the delay time, the amount acquired was then multiplied by-1. The approach employed with latency time, on the other hand, will not operate at cost value. Cloud storage costs are progressive, which implies they increase overtime over the pay period. As a result, evaluating the cost of data storage after each activity is difficult [16]. To maximize the cost, the cost was first estimated regardless of the quantity of data consumed, as shown below in the procedure called.

Algorithm

Illustration of the AI distribution mechanism

Necessitate

Cij and Cjk are the components of an artificial neural network when it is first created.

for < event = 1, Q > do

for < t = 1, N > do

pc count ← verify how long private loud storage services are offered

Regulate information available to have the same number of nodes in the output layer to pc count

Obtain file volume and file contact model quality to the ANN

for < i = 1,pc count > do

(yi)t ← production cost of private cloud node i

End for

For < i = 1, pc count > do

Produce event (ai)t = (yi)t/ ∑ ∀p(p)

End for

Perform all events (a)t

Monitor incentives (in)t from all private clouds

Calculate error for all private clouds

Inform system weight

end for

end for

Where Ocost is the entire cost of data storage in the cloud, and Outilized is the entire quantity of information stored in cloud storage. The entire system cost is denoted by the letter S. S utilized is the quantity of information transferred into and out of the storage services of the cloud. AO cost represents the entire value of all procedures in the cloud. i is the maximum numeral determined on the basis of cloud supplier i and n indicates the probability of cloud services accessible. Beyond this calculation, the goal is to calculate the cost increase in proportion to the cost consumed.

$$\textbf{Cost} = \sum_{i=1}^{n}\left(\frac{Ocosti}{Outilized\ i} + \frac{Scost\ i}{S\ utilized\ i} + \frac{AO\ Cost\ i}{AO\ utilized\ i}\right)$$

In most cases, the cost provides a result that is less than 1. In this investigation, calculating the price ratio using the same method as the latency time won't work. As a result, a distinct model was utilized. The following condition was added to promote learning confluence: every number of the above equations that surpassed 1 was utilized to calculate the relationship between the entire different cycles awaiting a novel worth surpassing 1; then the program substitutes this result with the correct value. This method allowed the algorithm to discover how to reduce costs completely and quickly. Eventually, distinct values for cost and latency were introduced to the performance metric to make the architecture more effective at optimizing both latency and cost at the same time. The ratings were calculated using the file's significance; this was generated from the patterns of data files' characteristics [17]. If the file is expected to be particularly dynamic in the near future, the architecture will place less emphasis on cost and instead aim to reduce latency. The same would be true in reverse: the relevance of latency time was lowered for idle files, and the cost was minimized. Algorithm 2 lays out the whole reward function.

Algorithm 2

The reward function is based on the overall cost of delivering each file among several private cloud storage providers and the delay time.

Necessitate

Execute numeral of private cloud storage services accessible nu. Private cloud

Necessitate

Parameters for the most critical file, as well as maximum latency and cost for each private cloud, should be executed. (max.Weight, max. Latency Array [], max.Cost Array [])

Necessitate

Execute three variables (latency Reward [], cost Reward [], cost []) in order to

Calculate the whole reward

Pa– determine the file's importance based on the predicted access pattern (nu.Read + nu.Write + lifeTime).

if pa > max.Weight then

max.Weight ← pa;. locate the most significant folder

end if

pa = pa/max.Weight. Calculate the relation of the present latency time to the

Maximum weight so far

for < i = 1,nu.privateCloud > do

if file Latency Time[i] > max. Latency Array[i] then

max. Latency Array[i] ← f ile Latency Time[i]. locate the slowest file *end if*

Latency Reward[i] ← −1 ∗ (f ile Latency Time[i]/max Latency Array[i]), .

Calculate the ratio of the present latency time to the slowest file latency so far, for each

Private cloud.

Calculate cost[i]

if cost[i] > 1 then

Max.Cost Array[i] ← cost[i]

end if

Cost Reward[i] ← −1 ∗ (cost[i]/max.Cost Array[i]). ratio of the present cost to the utmost charge so far, for each private cloud

Sum Reward[i] ← (1 − w) ∗ cost Reward[i] + w ∗ latency Reward[i] .

Calculate the whole compensation for allocating each file

end for

Revisit total Reward

5. Experimental Analysis

The cloud spanner simulator was written in Java and was used to simulate cloud storage efficiency and expenses. The time taken to submit (write) and receive (read) a file in and out of all cloud-based activities is referred to as a delay. Furthermore, the simulator computed the overall cost of using each cloud service's memory and network bandwidth. It was adaptable and effective at simulating multiple cloud storage services at the same time [18]. The functionality of the suppliers for the research was set up to imitate the presentation advantages of Google Cloud Storage, Amazon Web Services (AWS), Microsoft Azure Storage, and IBM Cloud in Fig. 3.

Table 1
Presentation variety of cloud storage services

Cloud supplier	highest speed MB/s	Slowest speed MB/s	Mean Speed MB/s
Google Cloud Storage	21.09	17.51	18.92
AWS	24.89	7.06	16.02
MS Azure Storage	20.11	15.90	17.96
IBM Cloud	18.95	15.71	17.78

The initial step in executing the architecture in the simulation is to use MPSA to predict the network traffic for each file. The file size is then provided to the RL program, which is taught with an ANN, along with other characteristics. Several variables must be adjusted before the RL algorithm and ANN can be implemented. These characteristics are likely to have an impact on the application's ability to understand. RL Parameter settings are exposed in Table 2.

Table 2
RL Parameter settings

constraints	standards
knowledge speed	$\alpha = 0.001$
sequential reduction feature	$\lambda = 0.6$
subsequent situation decay parameters	$\gamma = 0.65$
numeral of preparation period(events)	150000

The number of production nodes was determined by the availability of private cloud providers at each instance pace, as indicated above. A biased node with a fixed rate of + 1 was present in both the input and hidden layers. The output node employs stochastic filter coefficients, while all concealed utilized connections are a radial basis relocates function. Initially, the neural network conditions were determined to be: = 0.001; = 0.6; = 0.65.

Each cloud provider's latency times and total costs were evaluated and the results were after the test by sending full files to each service. This data serves as a reference. After dispersing the identical data equitably among the cloud service suppliers with the RAID approach with a knowledge framework, the latency and overall cost were tested once more. The framework was constructed utilizing four datasets totaling more than 9254 documents and more than 874 GB of storage. RAID-6 contains three storage locations to disperse files, therefore four options for cloud services were chosen. The latency times were measured, and each cloud provider's total cost was computed separately. This technique was performed for emulations of Google Cloud Storage, Amazon Web Services, Microsoft Azure Storage, and IBM Cloud, with three ways used as a standard to assess the planned methodology.

By using a mirrored strategy, Fig. 4 depicts the variations in the total cost on average and latency times on average among four distinct cloud providers. The major goal of reflecting information throughout all clouds (without even any divide allocation) was to get a better idea of the real rate and delay of instances of particular cloud storage, as well as the differences between other cloud providers.

Following that, the same information was dispersed by the same private cloud storage providers utilizing the EAIFDE architecture. Figures 5 and 6 indicate that the average overall cost was lower than the Security Disk Detection. The suggested model outperforms Security Disk Detection by around 39% and the EAIFDE technique by about 21%, according to these findings. When compared to Security Disk Detection and the EAIFDE, the average latency period for reading and writing was lowered by more than 71 percent and 52 percent, respectively. Using the EAIFDE architecture, the reading delay was 3.12 secs, and the writing delay was 0.82 secs.

Figures 7 and 8 demonstrate the comparability numbers of the evolutionary algorithms, OFDAMCSS [19], and EAIFDE, as well as the mean overall cost. These findings revealed that the suggested model outscored the heuristic method by about 43%, the OFDAMCSS technique by about 37%, and the EAIFDE technique by about 19%. The average latency times for written language were lowered by around 68 percent when compared to the evolutionary algorithm, OFDAMCSS by around 55 percent, and the EAIFDE by about 42 percent. Using the EAIFDE architecture, the delay time for reading was 2.11 seconds, and the delay time for writing was 0.79 minutes.

The mean figures for each cloud storage service follow transferring all four datasets lacking some break distributions are referred to as lag effort and expense. This is distinct from lag time and cost.

The overall cost and median latency time for each cloud service provider employing Security Disk Detection, heuristic techniques, the OFDAMCSS framework, and EAIFDE are shown in Figs. 9, 10, 11, and 12. This equates to-51 percent of Security Disk Detection levels,-49 percent of heuristic technique levels,-47 percent of OFDAMCSS technique attributes, and-41 percent of the EAIFDE approach. The average latency duration decreased by-58 percent as compared to Security Disk Detection, by 52 percent when compared to a heuristic method, by 48 percent when compared to OFDAMCSS, and by 42 percent when compared to EAIFDE. The mean prices for each cloud storage provider following transferring all four datasets without any partition distribution represent 47 percent of the delay cost and time. This is in contrast to the delay times and costs depicted in Figs. 9, 10, 11, and 12, which are for separate cloud storage services with split dispersion. This equates to-54 percent of Security Disk Detection levels,-51 percent of heuristic approach attributes, 48 percent of OFDAMCSS technique ideals, and-45 percent of EAIFDE framework ideals.

The cost and latency time for each segment was modified by how the EAIFDE structure divided each segment. Furthermore, to effectively assess the architecture, the cost of the networking for each file was increased by increasing the number of learning to read and write features to calculate the true cost of each file over its lifespan.

Figures 14 and 15 indicate that the EAIFDE distributions reduced the cost of space on Google Cloud Storage and IBM Cloud for all writing categories. However, the suggested technique raised the prices of MS Azure and AWS. Conversely, the estimated duration of transmitting data to Google Cloud Storage and IBM Cloud decreased, but it surged for AWS and MS Azure. Figures 14 and 15 show that the reading categories had similar outcomes. The above results indicated that the proposed structure was not capable of reducing the cost of each cloud storage service while improving its functionality. The methodology, on the other hand, takes into account the overall cost and means latency time of all cloud storage services.

This experiment has shown that the recommended EAIFDE structure can reduce costs and delay time for many private cloud providers. When compared to data distribution using normal methods, the cost of transferring files across several cloud storage providers was lowered by up to 39% on certain clouds, according to the findings of this study. Conversely, for specific cloud services, the current proposal outscored the heuristic approach by roughly 27%. When data was dispersed over four cloud storage providers, EAIFDE reduced latency by roughly 79 percent in comparison to Security Disk Detection and by about 67 percent as compared to the heuristic technique [20]. Compared to the OFDAMCSS technique, this solution saves roughly 61 percent. The platform's generating abilities were called into question by a variety of cloud providers. The findings revealed that the suggested EAIFDE architecture might reduce costs and delays for a variety of cloud service providers [21].

6. Conclusion

This proposal proposes an Effective AI Architecture for File Distribution Enhancement among Private Cloud Storage Providers, which would minimize protracted costs and delay time by distributing files across different cloud services. The main problems with this task were how to communicate with various locations while performing a non-fixed variety of operations at the same time, as well as how to manage a variety of "non-stationary" incentive messages. As a result, the reinforcement learning algorithm was developed in an innovative way to achieve the study objectives. The findings indicate that the designed system may dramatically reduce both expenses and mean latency time across numerous cloud storage services. The cloud emulator configurations were produced at random between the optimum and least efficiencies in these investigations, and the identified stakeholders were predicated on the top and cheapest costs of Google Cloud Storage, AWS, MS Azure Storage, and IBM Cloud categorizer. The rationale for the random parameters was that there were not many cloud storage providers commercially available. The paradigm improves cost and latency time in multiple clouds, with a nominal of three clouds and most private clouds, according to the generalized testing results. Moreover, measurements were carried out to investigate the impact of variables RL and ANN on the platform's learning motivation. Costs were computed focused on a tiny number of the additional datasets in this research; real-world enterprises have substantially bigger streams of information to store in cloud services. As previously stated, this process involves transferring files among multiple private clouds. Future work should look at the effects of modifying the factors, as well as the interdependence of the variables.

7. Declarations

Acknowledgement: The authors are grateful to the King Khalid University, Abha, Saudi Arabia, Researchers Supporting Project.

Funding Statement: The authors would like to thank King Khalid University, Abha, Saudi Arabia, for funding this work through a Small Research Project under grant number RGP/161/42. We thank the "anonymous" reviewers for their so-called insights.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

8. References

1. Zuluaga, M., Krause, A., & Puschel, M.,” E-pal: An active learning approach to the multi-objective optimization problem. Journal of Machine Learning Research, vol.17, pp. 1–32, 2016.
2. Zhou, A. C., He, B., Cheng, X., & Lau, C. T. ,” A declarative optimization engine for resource provisioning of scientific workflows in iaas clouds”, In Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing, HPDC ’15, New York, NY, USA. ACM, pp. 223–234, 2015.
3. Xu, C.-Z., Rao, J., & Bu, X. ,” A unified reinforcement learning approach for autonomic cloud management”, Journal of Parallel and Distributed Computing, vol. 72,no.2,pp.95–105,2012.
4. Voas, J. & Zhang, J. ,” Cloud computing: New wine or just a new bottle?”, IT Professional, vol.11, pp.15–17,2009.
5. Vengerov, D. ,” A reinforcement learning framework for online data migration in hierarchical storage systems”, The Journal of Supercomputing, vol. 43,no. 1, pp.1–19,2008.
6. M. Khatib, Al-Nakeeb, A. and G.Ahmed, ”Integration of cloud computing with artificial intelligence and its impact on telecom sector—A Case Study”, iBusiness, vol.11, no.1,pp. 1-10,2019.
7. Van Hasselt, H. & Wiering, M. A. ,” Reinforcement learning in continuous action spaces”, In 2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, pp. 272 279,2007.
8. Van Hasselt, H., Guez, A., & Silver, D,” Deep reinforcement learning with double Q-Learning”, In AAAI, pp. 2094–2100,2016.
9. Tanimoto, S., Sakurada, Y., Seki, Y., Iwashita, M., Matsui, S., Sato, H., & Kanai, A.,” A study of data management in hybrid cloud configuration”, In 2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp. 381–386,2013.

10. J. Robertson, J. M. Fossaceca and K. W. Bennett, "A Cloud-Based Computing Framework for Artificial Intelligence Innovation in Support of Multidomain Operations," in IEEE Transactions on Engineering Management, pp. 1-10,2021.
11. Tanja Hagemann and Katerina Katsarou. ,"A Systematic Review on Anomaly Detection for Cloud Computing Environments", In 2020 3rd Artificial Intelligence and Cloud Computing Conference (AICCC 2020), Kyoto, Japan. ACM, New York, NY, USA 14, pp.18–20, 2020.
12. C. Lin, H. Sun, J. Hwang, M. Vukovic and J. Rofrano, "Cloud Readiness Planning Tool (CRPT): An AI-Based Framework to Automate Migration Planning," 2019 IEEE 12th International Conference on Cloud Computing (CLOUD), 2019, pp. 58-62, doi: 10.1109/CLOUD.2019.00021.
13. M. R. Uddin, K. M. Kabir and M. T. Arefin, "Artificial Neural Network Inducement for Enhancement of Cloud Computing Security," 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1-6.
14. Rao and S. Rao, "Application of Artificial Neural Networks in Capacity Planning of Cloud Based IT Infrastructure," 2012 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), 2012, pp. 1-4
15. Papaioannou, T., Bonvin, N., and Aberer, K. ," Scalia:An adaptive scheme for efficient multi-cloud storage", In High Performance Computing, Networkin, Storage and Analysis (SC)" , 2012 International Conference, pages pp.1–10. ACM..2012.
16. Whiteson, S. ," Evolutionary Computation for Reinforcement Learning",Springer, Heidelberg New York Dordrecht London,pp. 325–355, 2012.
17. Bowers, K. D., Juels, A., and Oprea, A. ,"Hail: A high-availability and integrity layer for cloud storage",In Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 187–198,New York, NY, USA. ACM,2009.
18. Bessani, A., Correia, M., Quaresma, B., Andr, F., and Sousa, P. ," Depsky: Dependable and secure storage in a cloud-of-clouds", In Proceedings of the Sixth Conference on Computer Systems, EuroSys '11, New York, NY, USA. ACM, Pp. 31–46, 2011.
19. Algarni A. and Kudenko D.," Distribution Data Across Multiple Cloud Storage using Reinforcement Learning Method", In Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017, pp. 431-438,2017.
20. H. Ghavamipoor, S. A. K. Mousavi, H. R. Faragardi and N. Rasouli, "A Reliability Aware Algorithm for Workflow Scheduling on Cloud Spot Instances Using Artificial Neural Network," 2020 10th International Symposium onTelecommunications (IST), pp. 67-71, 2020.
21. S. Kapoor and S. N. Panda, "Energy-Aware Parallel Computing for Cloud Architecture using ANN," 2019 Global Conference for Advancement in Technology (GCAT), pp. 1-4, 2019.

Figures

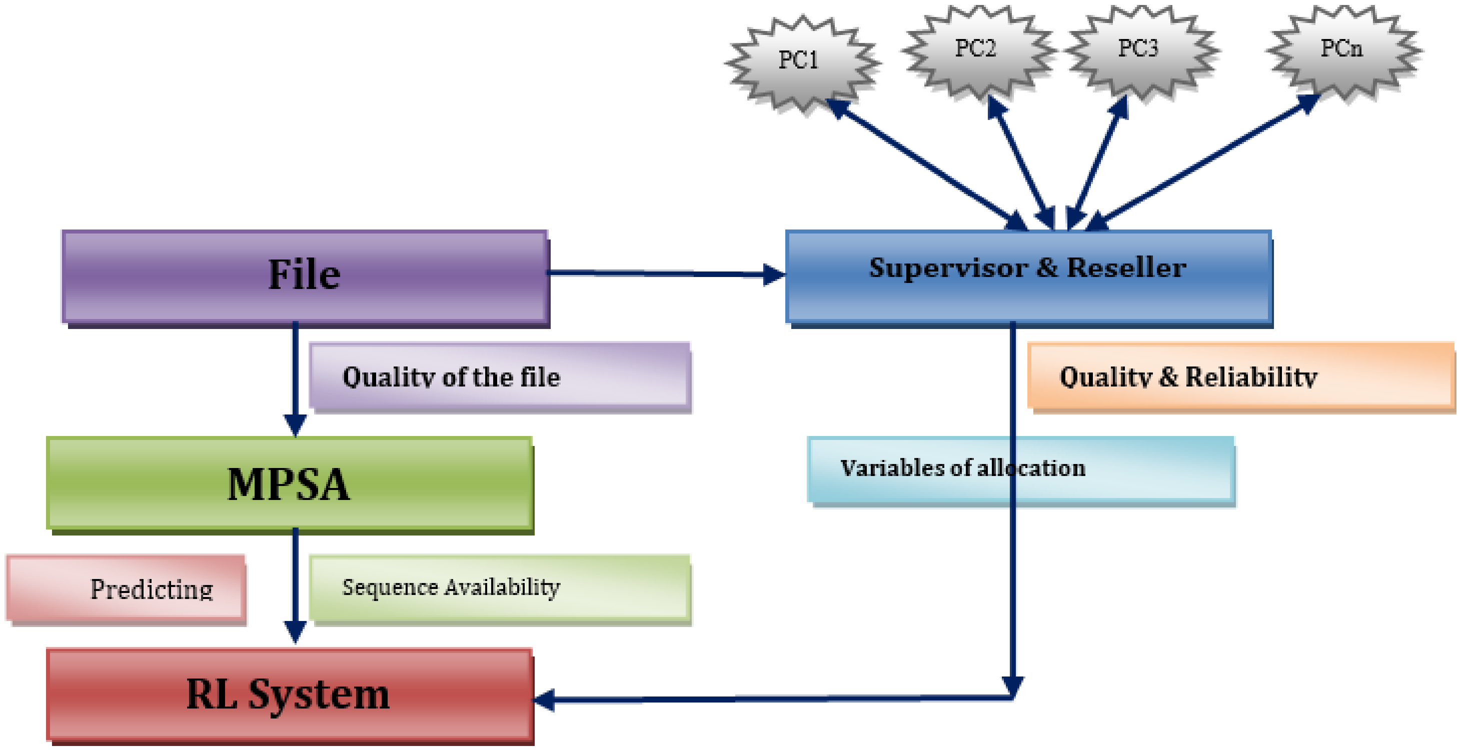

Figure 1

The AI-based Framework Architecture from a moderate perspective

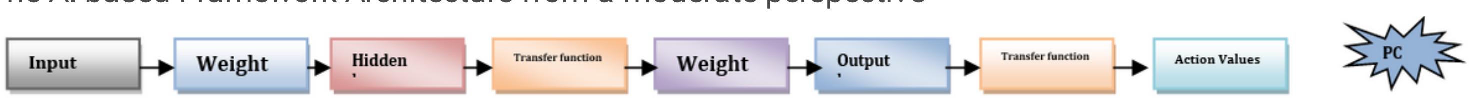

Figure 2

RL of File Distribution Enhancement - Private Cloud Services

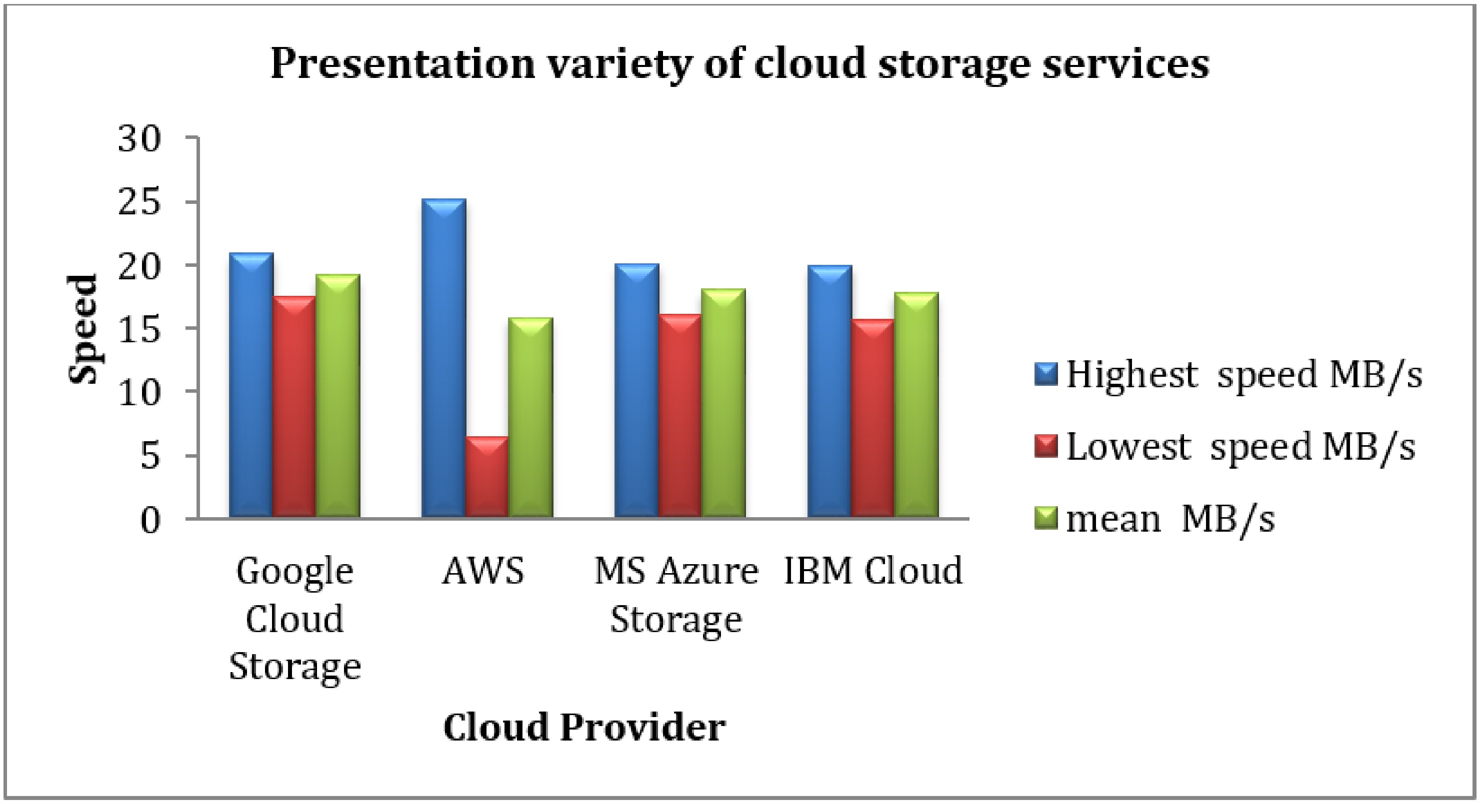

Figure 3

Presentation variety of cloud storage services

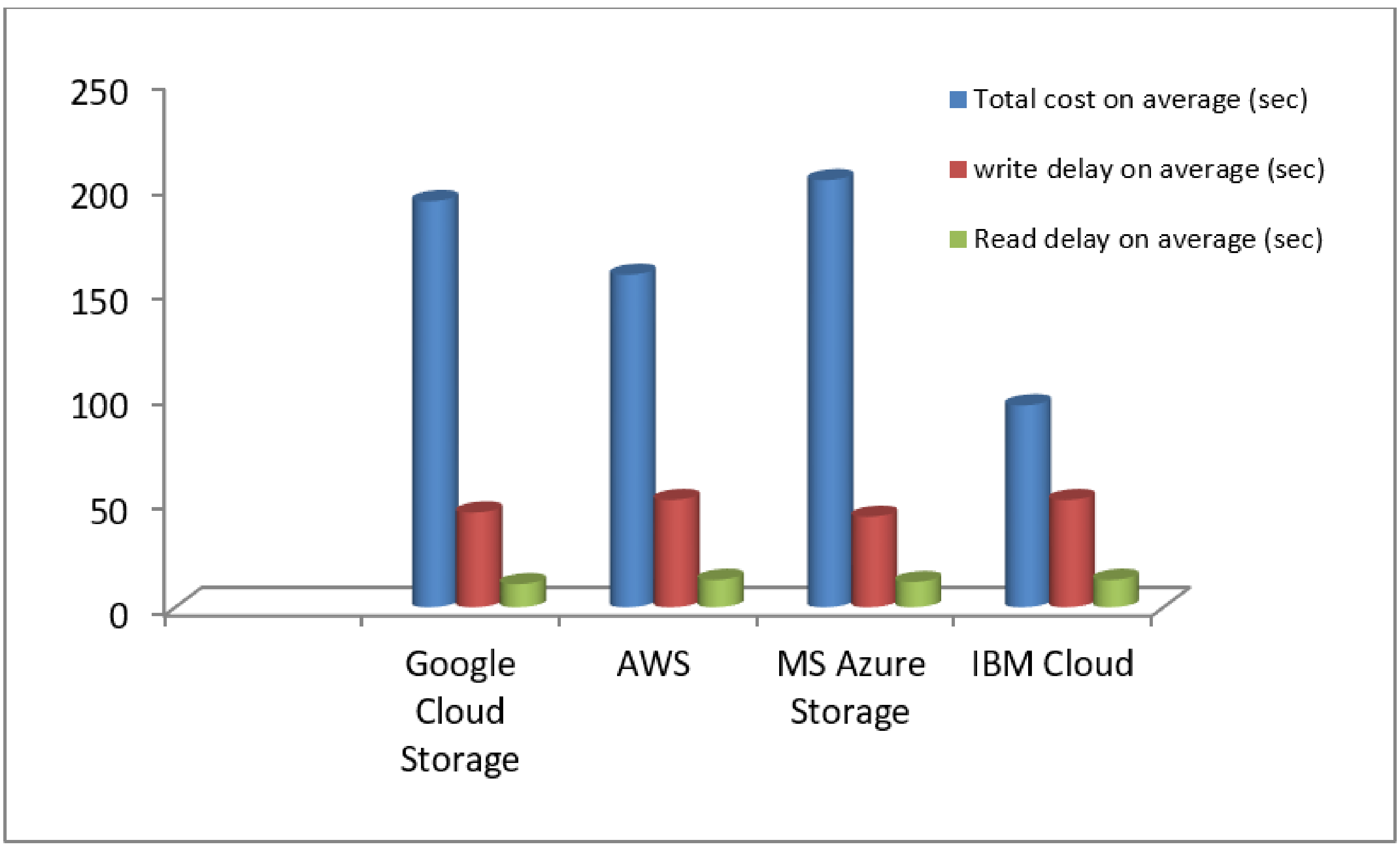

Figure 4

Sending all information from all databases into one cloud platform at average costs and mean latency time

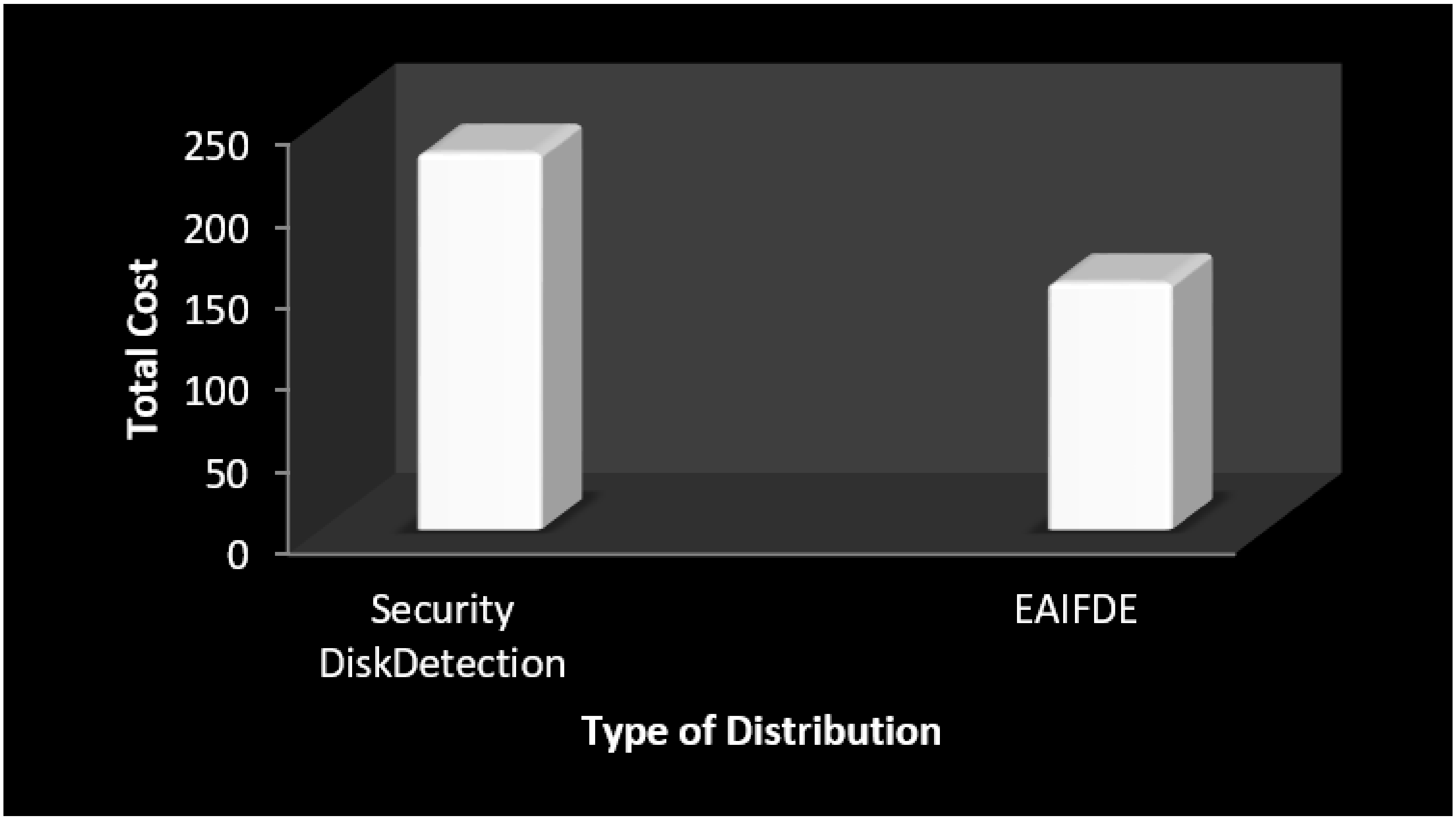

Figure 5

The variation in estimated cost after using Security Disk Detection and EAIFDE to distribute all data.

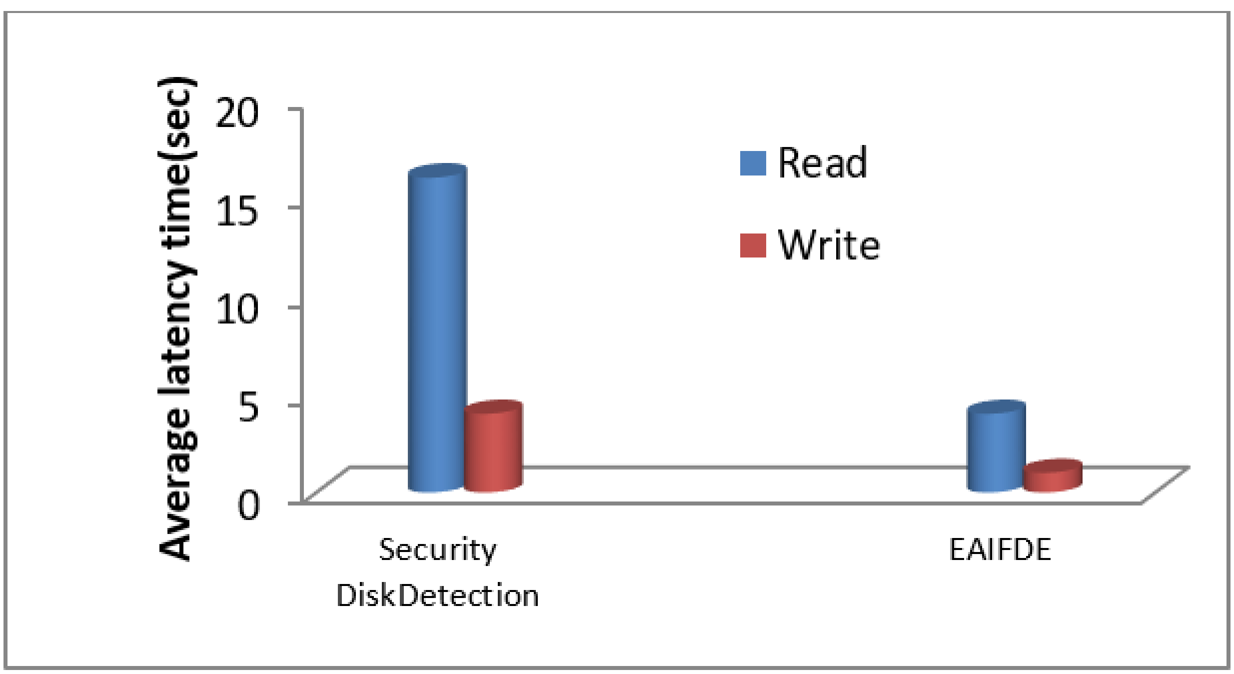

Figure 6

The variation in latency time (read and write) while using Security Disk Detection and EAIFDE to distribute all documents.

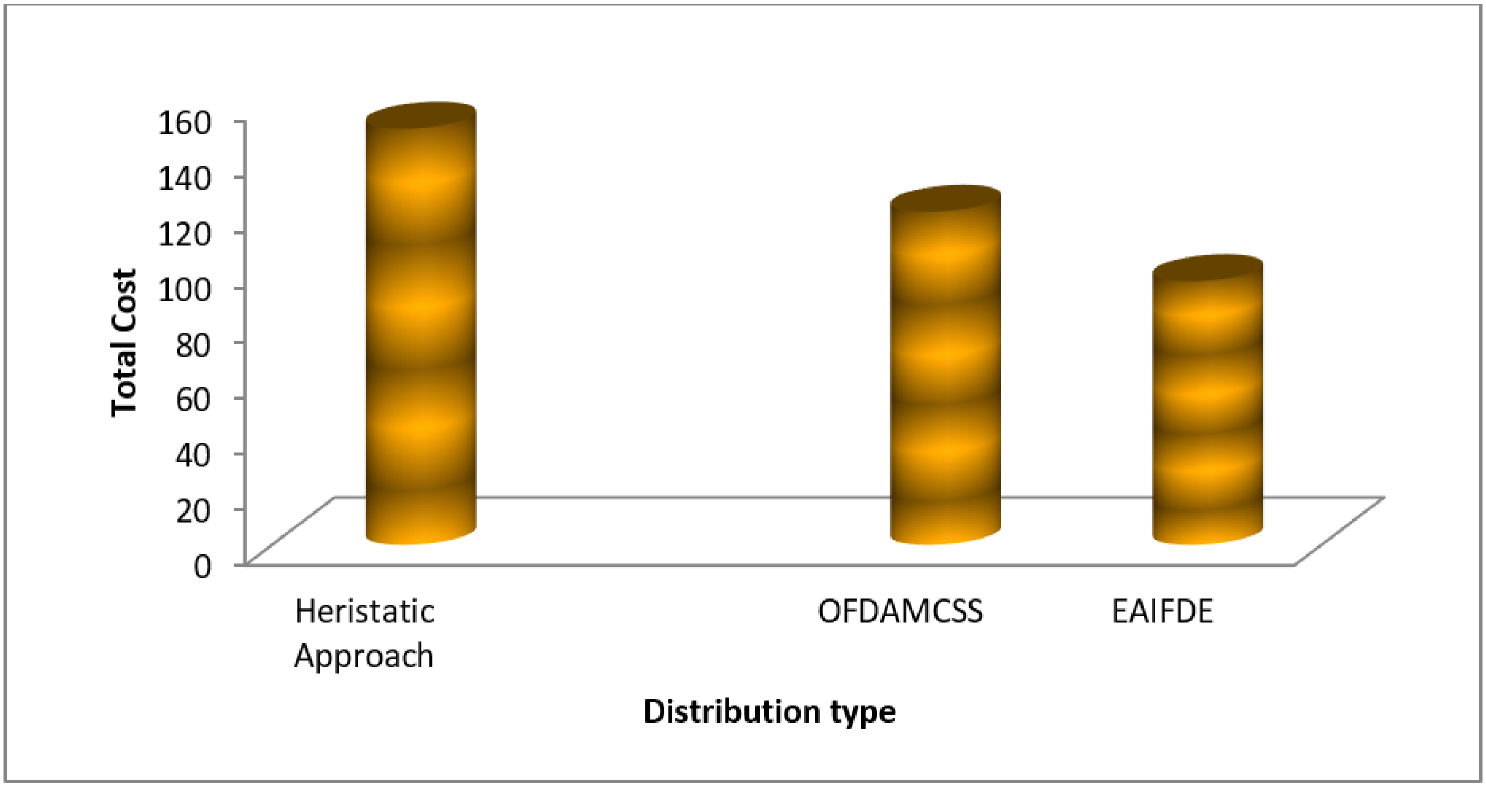

Figure 7

The variation in average cost between OFDAMCSS and EAIFDE after dispersing all files using the heuristic strategy.

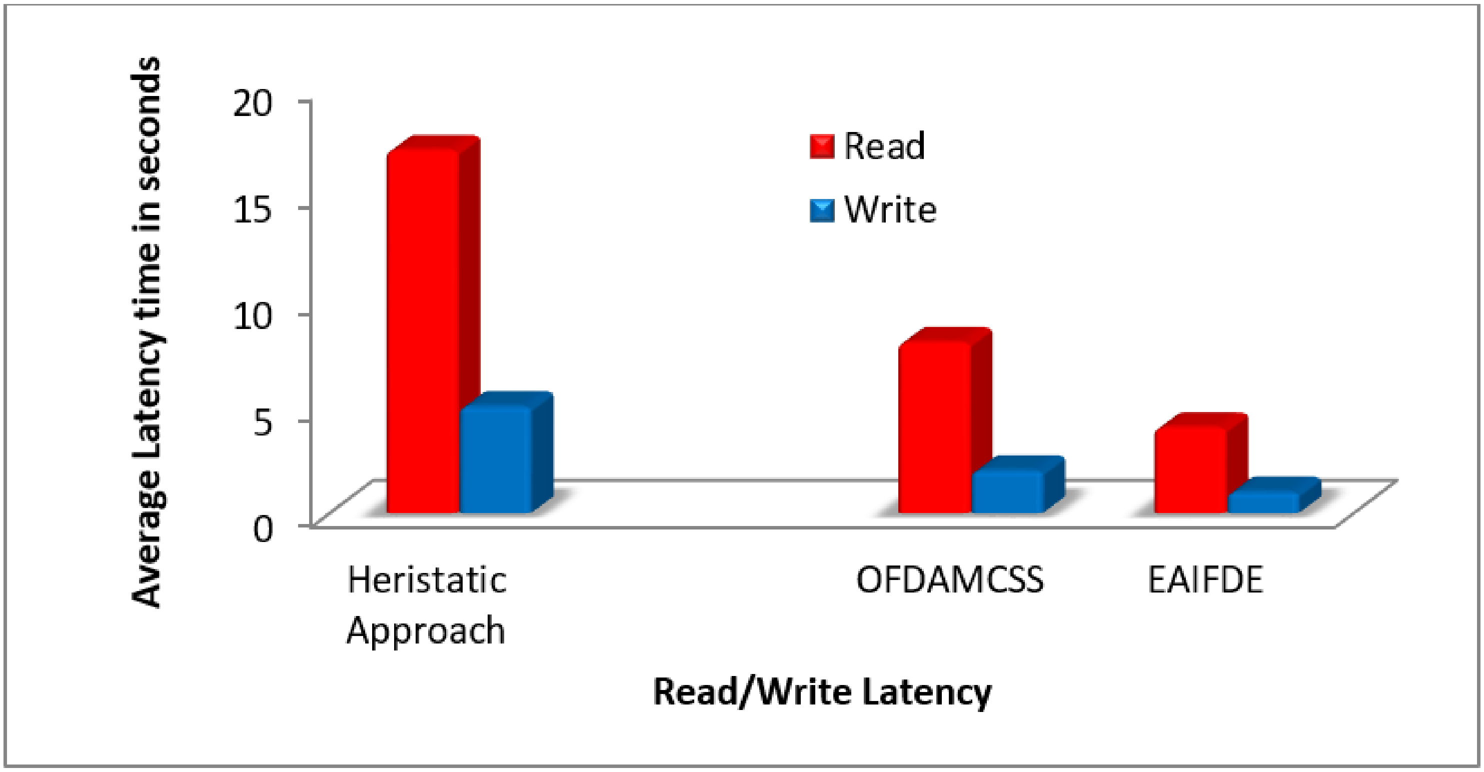

Figure 8

The variation in latency time (read and write) after that using the heuristic approach, OFDAMCSS, and EAIFDE, to distribute all documents.

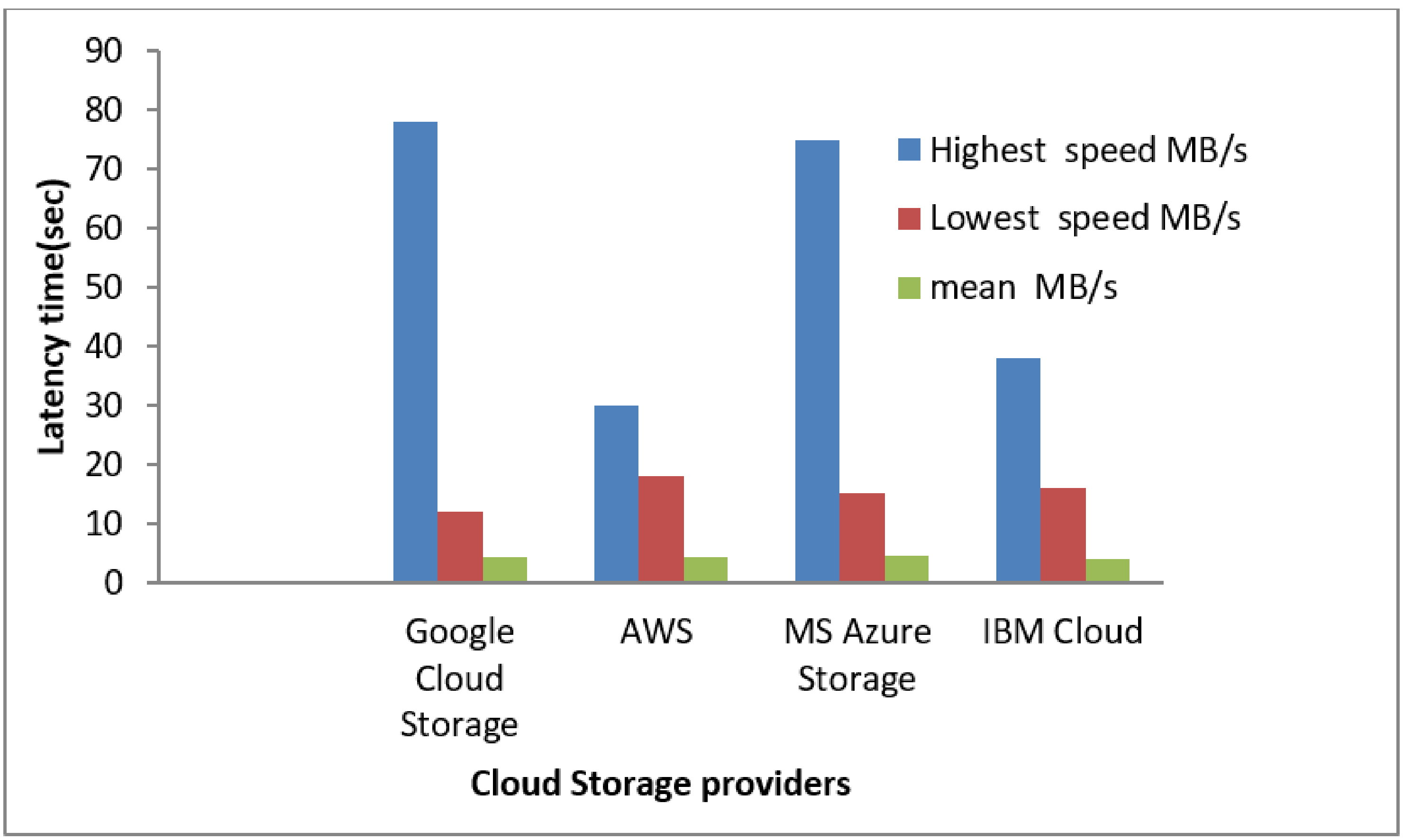

Figure 9

Utilizing Security Disk Detection all cloud providers' total cost and median latency time are evaluated.

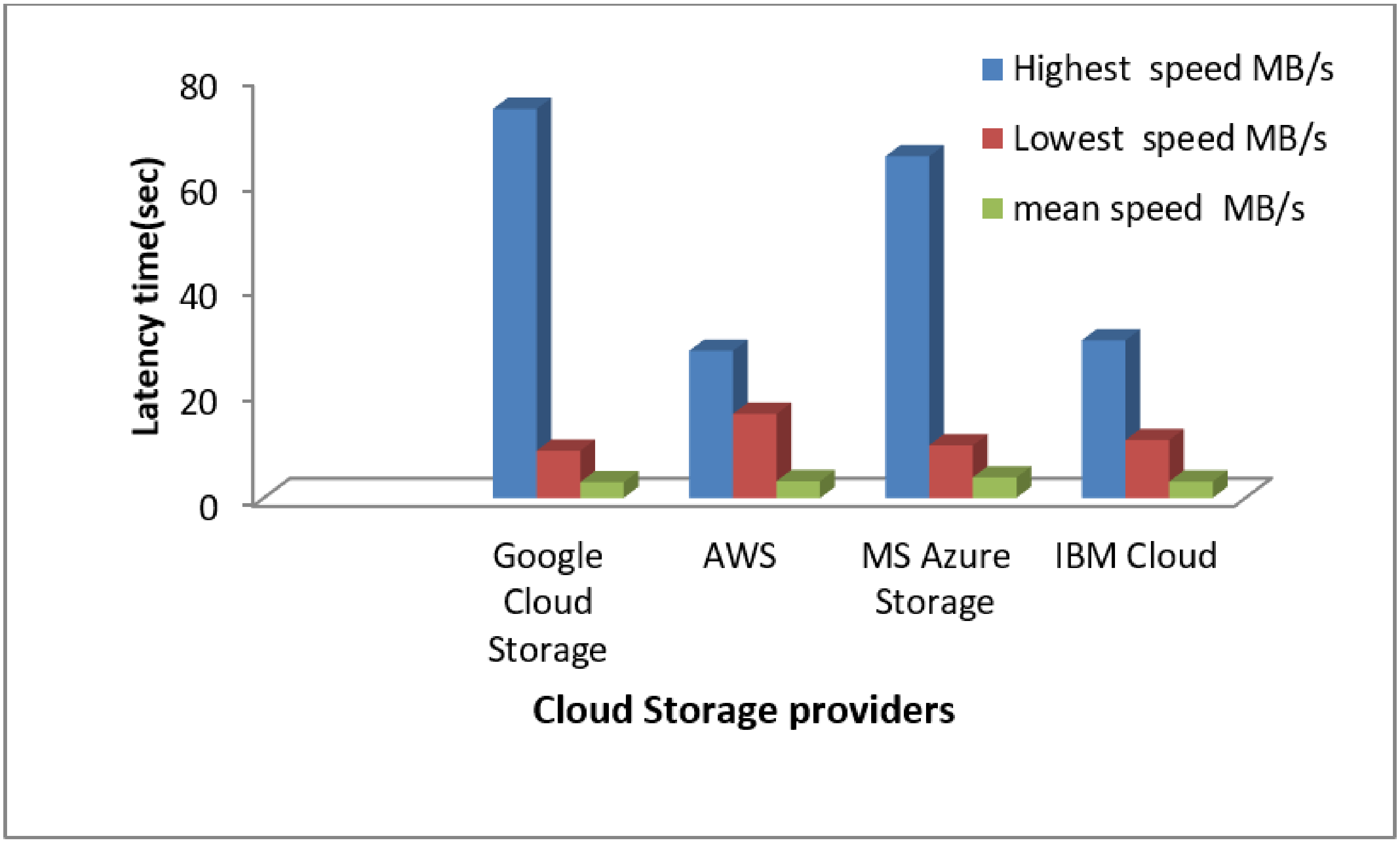

Figure 10

Using a heuristic technique, all cloud providers' total cost and median delay time are evaluated.

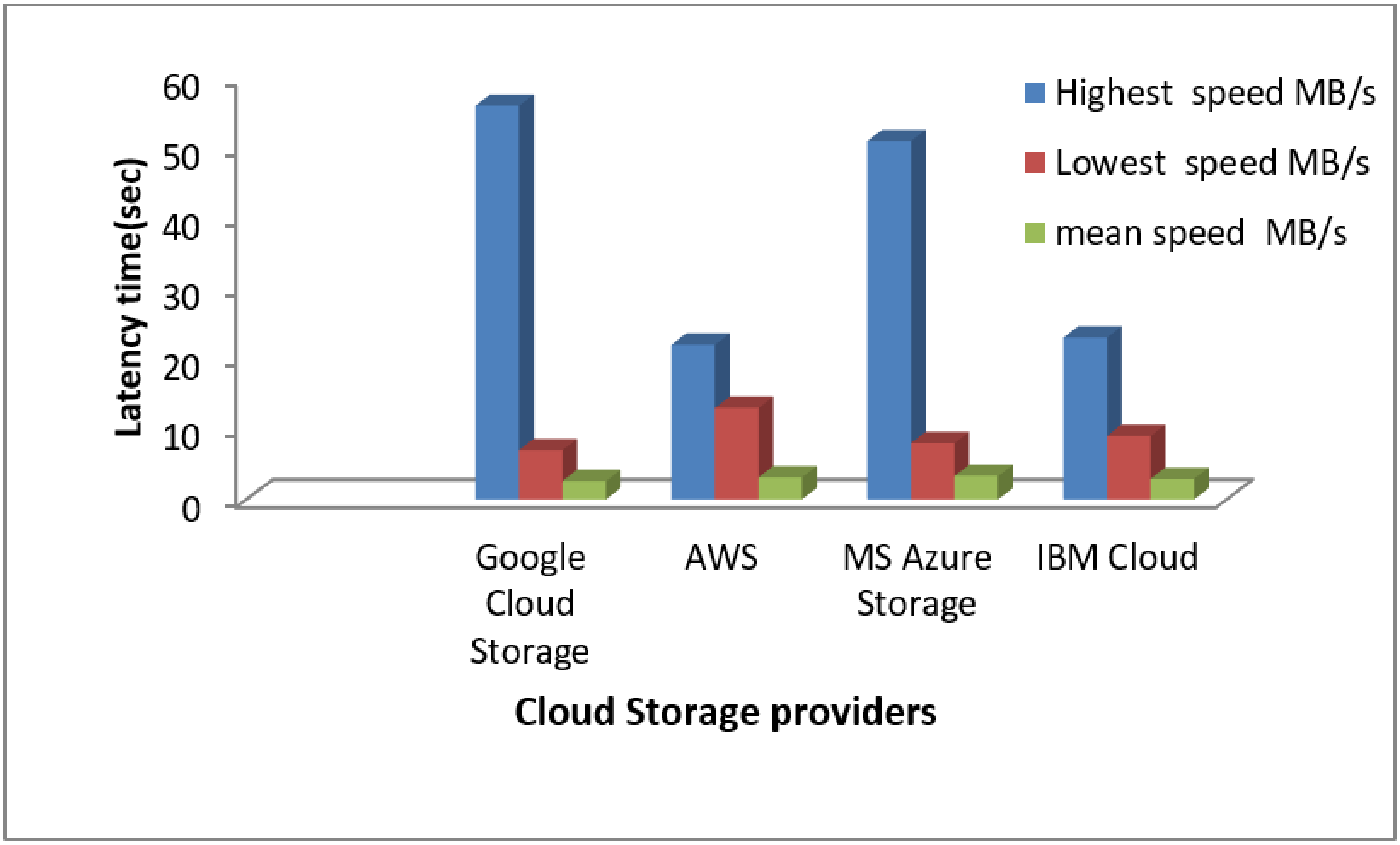

Figure 11

Using the OFDAMCSS methodology, all cloud providers' total cost and median latency time are evaluated.

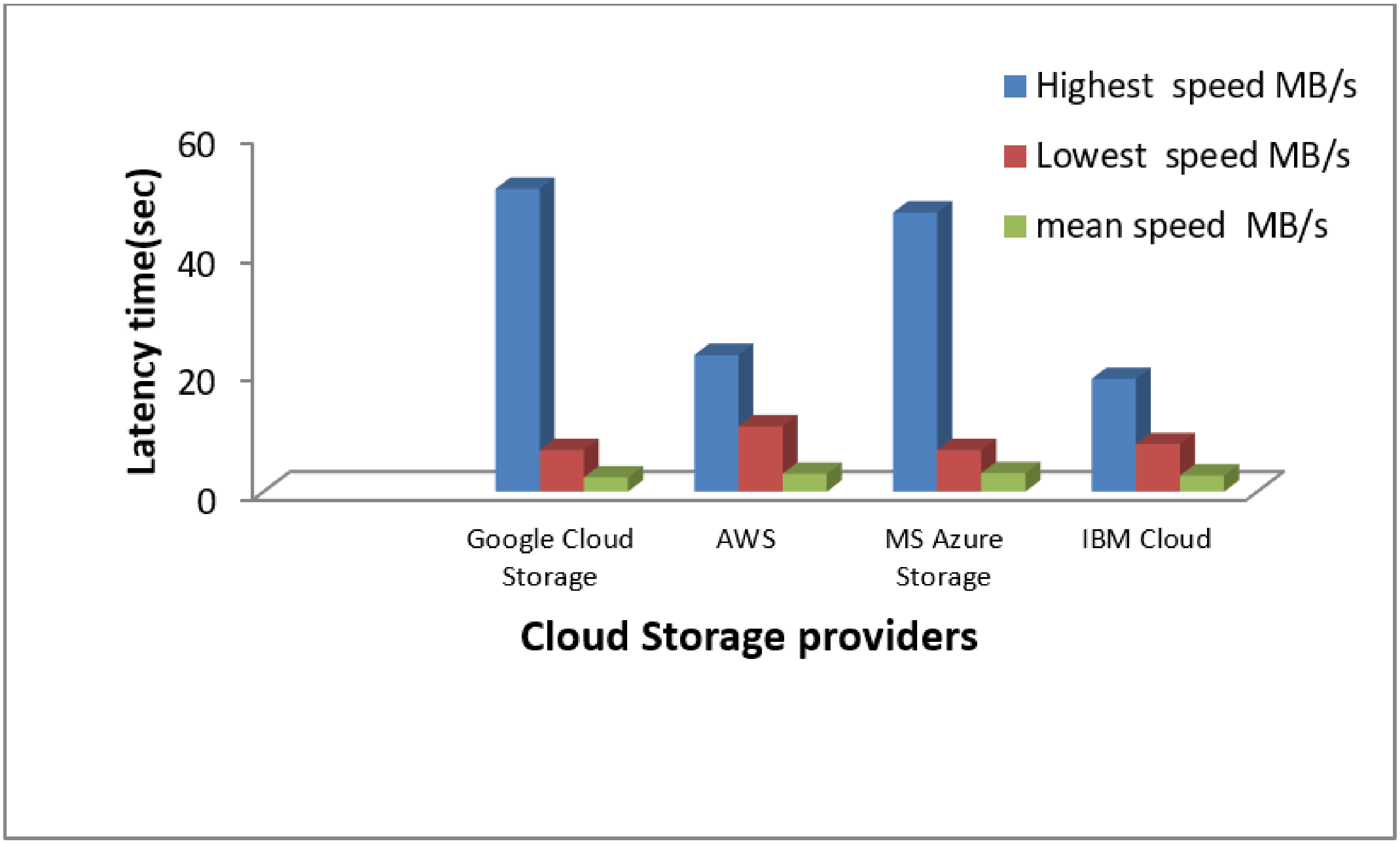

Figure 12

Using the EAIFDE methodology, all cloud providers’ total cost and median latency time are evaluated.

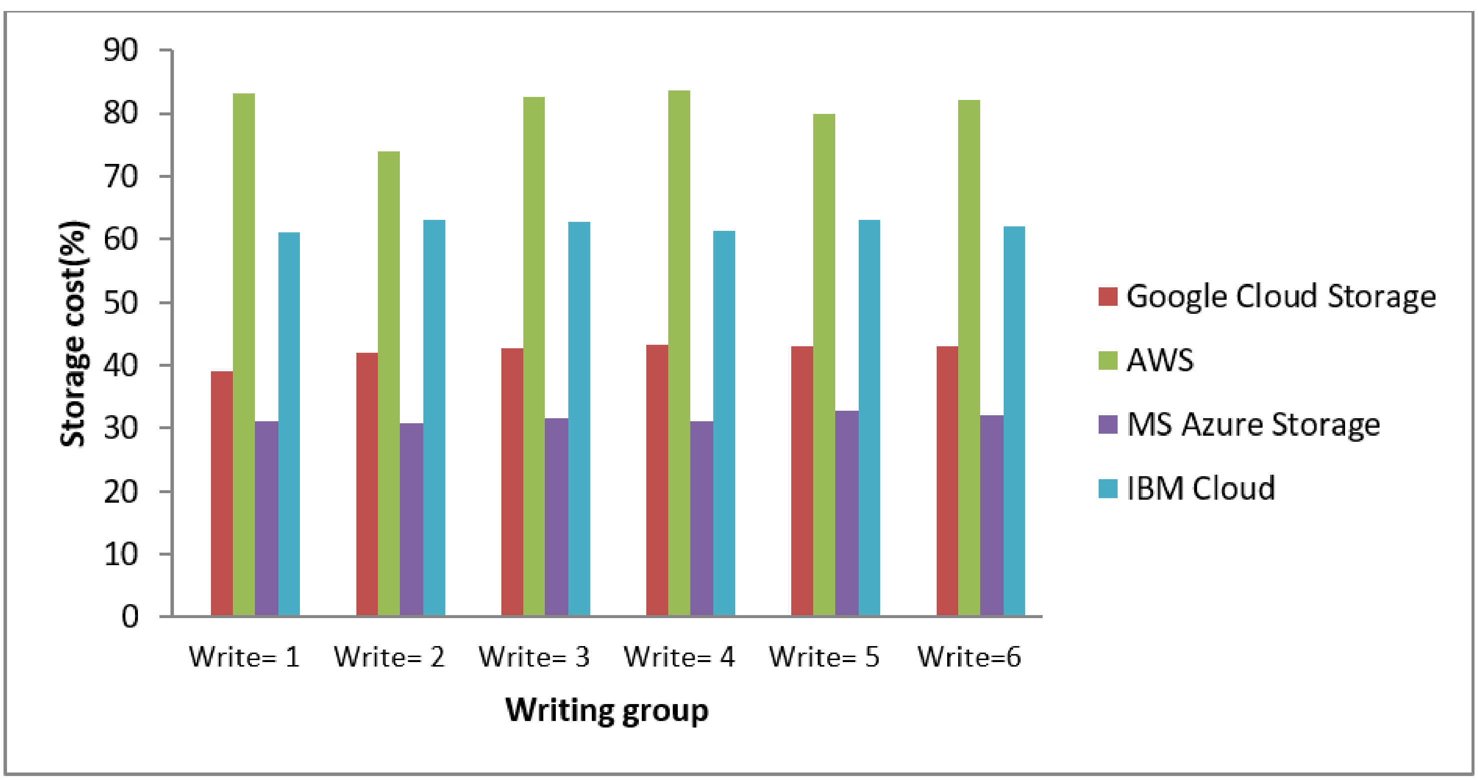

Figure 13

For the writing group, there has been a storage cost change as a percentage of the total

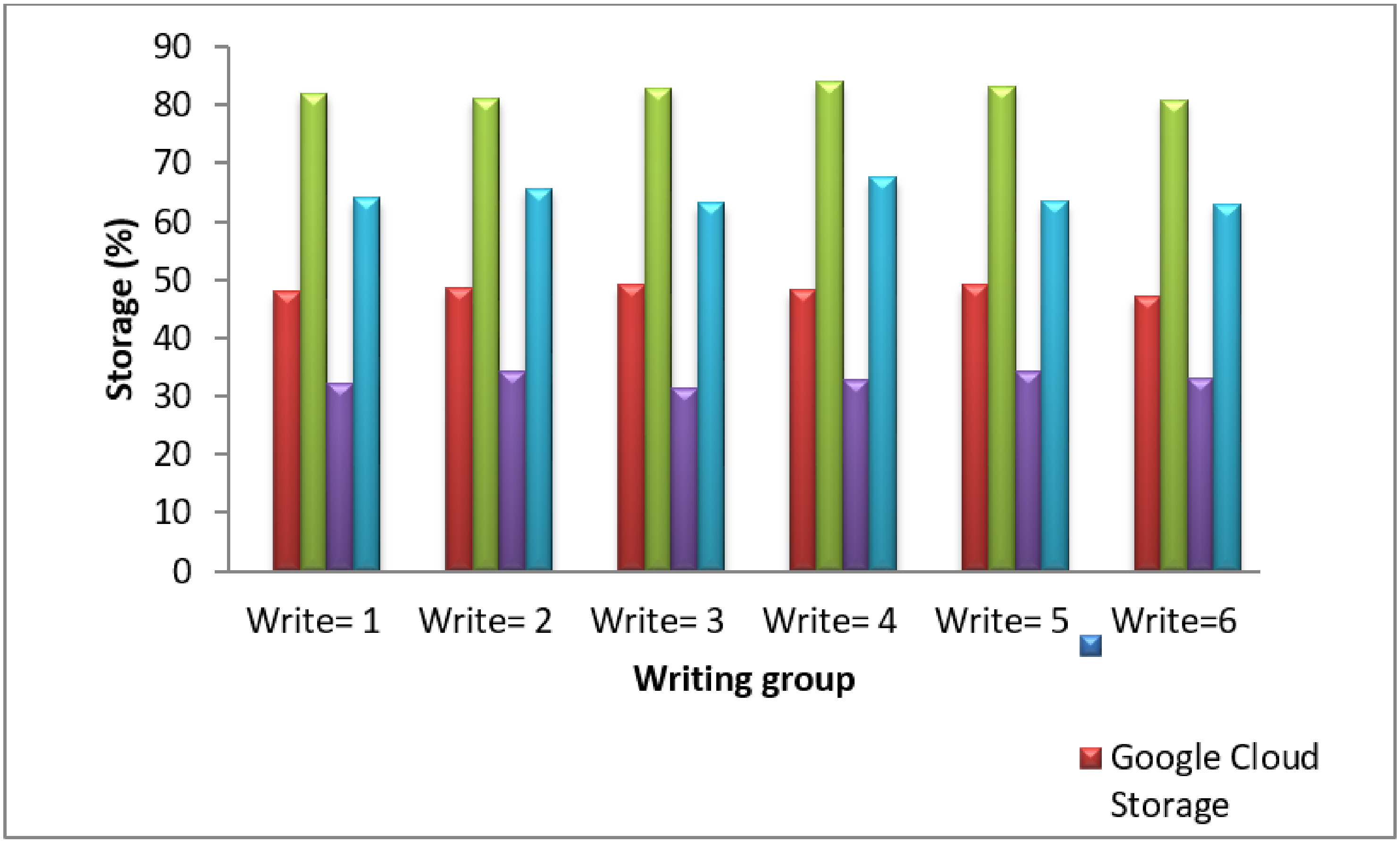

Figure 14

For the writing group, there has been a storage change as a percentage of the total