SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V10I5P116

Volume 10 Issue 5, 174-184, May 2023

Original Article

An Effective Approach for Improving Data Access Time using Intelligent Node Selection Model (INSM) in Cloud Computing Environment

K. Rajalakshmi[1], M. Sambath[2], Linda Joseph[3], K. Ramesh[4], R. Surendiran[5]

[1, 2, 3]*Department of Computer Science Engineering, Hindustan Institute of Technology and Science, Chennai, India.*
[4]*Department of Computer Science Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India.*
[5]*School of Information Science, Annai College of Arts and Science, Kumbakonam, India.*

[1]*Corresponding Author : rajee.be89@gmail.com*

Received: 14 March 2023 Revised: 23 April 2023 Accepted: 17 May 2023 Published: 31 May 2023

Abstract - *This Cloud environment offers users friendly services that assist them in achieving their professional and personal objectives. Any personal computer or other device with a broadband connection can access the data. The data can be an image file or a document etc. To protect such data from illegal access, proper security measures must be adopted so that the data resides safe and secure in a third party premise. The centremost node always has the lesser possibility of being hacked, leaving the data to be secure aside. Also, the central nodes have improved retrieval time. This study aims to propose an Intelligent Node Selection Model (INSM) comprising centrality measure and choosing the centre most stable nodes with less energy and security cost in the network for placing the data fragments so that the access time gets improved and node failure is significantly reduced. Improving the client success ratio gives the customer a sense of satisfaction. The cost estimation is high as it includes more parameters than the existing one. The different cost estimation at various stages of the Intelligent Node Selection Model (INSM) is compared with the existing node selection mechanism, the Optimal Selection Model. The average cost of node selection is approximately 1.1% higher than the Optimal Selection model as the cost of nodes in the INSM model is incurred for calculating the degree of centrality and node ranking to determine the success probability, security cost and energy cost. Thus, the node access time is better, and the failure of nodes also gets reduced so that the user will have safe access to data. Calculating the degree centrality of all the nodes in a network takes* $\Theta(V^2)$ *time, and for edges,* $\Theta(E)$ *where V is the vertices and E are the edges.*

Keywords - *Centrality measure, Cloud service provider, Degree centrality, Node access time, Node rank, Stable nodes.*

1. Introduction

There are various areas in which cloud technologies are explored to a greater extent. Resource sharing, Scalability, Flexibility, Self-identifying resources, pay as your use are some of the fantastic features of the cloud that draws the attention of a multitude. Amazon, Google, and Microsoft are well-known reputed vendors of cloud services. One of the essential uses of cloud technology is storage space, which facilitates information suppliers to move data from their constrained computer systems to the Cloud. By providing a similarly low-cost, flexible, category-independent platform for customers' data, cloud storage applications have today become a faster earning development point. While cloud computing is created depending on open architectures and interfaces, it can mutually integrate several interior and exterior cloud applications to supply high interoperability.

Numerous developments started the period of Cloud Computing, which is an Internet-based growth and employ of computer technology. Cloud technology has several uses, including providing free access to costly technologies and lowering the cost of setting up and maintaining machines and applications because no infrastructure is required. Data may be placed anywhere by users. Every user must sign up for a system, like the Internet. Although cloud computing first served as a platform for social networking, it is now commonly used to securely connect applications and storage without concern for maintenance costs and processing capacity. Organizations can acquire access to the cloud and unload their information technology (IT) infrastructure there. Government agencies are also migrating portions of their IT infrastructure to the cloud, in addition to private companies.

Big data comprises the digital information gathered from various digital sources, including the Web, e-mails, media platforms, portable devices, printers, numerical models, etc. Cloud storage enables users to move their enormous volume of data to data centres where massive and parallel information processing could happen, which consumes more power and space. The centermost nodes will always be less likely to be hacked, leaving the data secure. Also, the central nodes will constantly have improved retrieval time. Thus, nodes are selected, and data files are stored. Irrespective of the central nodes selected using the degree centrality measure in this study, selecting stable nodes for the network is another difficulty. The node's success probability is also considered a prime factor for being the critical node for selection.

The centermost node always has the lesser possibility of being hacked, leaving the data to be secure aside. Also, the central nodes have improved retrieval time. With privacy-preserving mechanisms and replication procedures, the issues of data dependability, authenticity, accessibility, and processing speed are addressed in more extensive networks. Additional off-site storage and data users of cloud-based utilities must transport data in a standardized, distributed system, increasing the system's vulnerabilities for data kept in distant locations.

The entire cloud ecosystem is vulnerable to any weak entity. The protection technique must substantially enhance the intruders' attempt to recover a fair quantity of information after successful penetration, perhaps minimising data leakage, to triumph in such a situation. The detailed system architecture described above illustrates the various entities and their roles in safeguarding the data in the cloud premise. In Section 1, a brief introduction about the topic is given, and in Section 2, works related to the proposed algorithm are examined section 3 explains the proposed methodology, section 4 provides results and discussion, and finally explains the conclusion and future work in Section 5.

2. Related Work

The ideas and conceptual structure related to the scope of the study are narrated with the proper explanation that aids in understanding the actual scenario of the work and makes clear the relative importance and interrelations of the proposed work.

2.1. Distributed Node Selection

As part of their research, the authors [Abdel Raouf et al. (2018)] suggest an architecture that may be used in a cloud setting as part of their study. At the beginning of the design process, while the measurement is being done, a formally specified choice for horizontal and vertical segmentation, distribution, and duplication of components might also be established. In this paper, [Bhardwaj, K *et al.* (2019)] The Edge Exchange, a directory service for a multistakeholder edge, is discussed in this article as a potential solution to allow programs to be distributed throughout the finest edge resources even while giving each participant authority over own resources use as well as distribution rules. As a result, the proposed model focuses on data security in hybrid cloud environments, as data communication is the most important function in a network environment.

Cloud infrastructure and its nodes' contribution are essential in a hybrid cloud since the communication system requires a cluster center, the author [Dhinakaran, K et al. (2016)] identified nodes based on an optimization technique. In this study, [Singh, A.P. et al. (2021)] has offered a thorough analysis of cloud data storage services and related security risks, evaluation of the SEDuLOUS technique, and procedure to enhance the SEDuLOUS by defining the lowest fragments to confirm fragmentation of all file types and hash functions of each large piece to distinguish disrupted memory nodes.

2.2. Scalable and Cost-Efficient Algorithms

In this research, the author [Eisa, I et al. (2017)] suggests a clustered file structure for cloud DBMS that decreases file transfer to maintain network load across multiple servers while enhancing flexibility by introducing more storage systems or infrastructure. The author [Rajalakshmi, K et al. (2023)] describes The Intelligent Data Fragmentation Model (IDFM) as operating in two stages, with phase I focusing on breaking the data file into smaller chunks based on selected random parameters. Phase II focuses on restoring the initial file from the evidence file in the event of a disaster. According to the author [Gopinath S et al. (2018)], data production and distribution are done dynamically in response to altering conditions and access permissions behaviours under a dynamic replication approach.

In this research, the authors [Ibrahim, N.M. et al. (2017)] offer a new feature extraction approach that uses Ant Colony Optimization and Decision Tree to improve the cloud IDS recognition rate. The suggested subset of features approach improves conventional optimization strategies for Cloud IDS, such as Genetic Algorithm and Rough Set, according to research observations employing data.The author presents a detailed comparison examination of cryptography protection methods [Kaaniche, N et al. (2017)]. First, [Manogaran G. et al. (2016)] investigate the challenges and potential solutions for protecting large amounts of data in the cloud in this investigation. The company has introduced a new architecture for storing big data in cloud data centres, providing security for Big Data in these facilities. An effective way of analysing vast amounts of data in a cloud-based computing environment is achieved through the use of this architecture, and this further helps the company generate different business ideas.

2.3. Mechanism Design-Based Node Selection Algorithm

Based on the genetic algorithm, a new approach to improving performance, the HNSA, has been proposed by [Kanwal et al. (2021)]. The suggested strategy is derived from a random generation method that combines the actions of several hosts' mean occupations. Choose the head node and candidate node using the HNSA, depending on the available resources. The document was then secured in the suggested approach, according to the author [Manek, M. et al. (2018)] Cloud Oriented Dispersed and Encrypted File Storage (CODE-FS), after which it is separated into fragments and scattered over several cloud nodes. The fact that each node only has a single copy of any file submitted by a user in whatever format assures that critical information will not be compromised in an attack.

This study concentrates on information segmentation strategies and illustrates how they might affect a cloud service's quality. According to the author [Mansouri, N et al. (2021)], various segmentation strategies are developed and evaluated for various data classifications depending on Amazon AWS. An innovative, lightweight cryptographic technique has been proposed in this research by [Thabit F et al. (2021)]. Depending on key encryption, it uses encryption. In comparison to the cryptography algorithms that are often used in cloud computing, the experimental outcomes of the suggested method demonstrated a high degree of protection as well as a noticeable improvement in measurements of encryption runtime and security services. The suggested system [Prasuna, T. et al. (2018)] provides good replicas, solves the information localization challenge with an enhanced data replication layout, and assigns adequate personnel to finish the Map Reduce task to produce more accurate replicas.

2.4. Reduction of False Positives using Optimal Node Selection

Here, [Miloudi, i.e. et al. (2020)] suggests a new approach to dynamic replication based on a categorization model of data that would enable user activity in connection with the data to modify the process of regeneration as a result. In this, [Moral, W.D *et al*. (2016)] the main points of attention are the information available, duplication, and privacy. The recommended approach uses the fragmentation and single replication (FASR) concept in the cloud to boost security and speed up data retrieval instead of using traditional cryptographic techniques, which can provide privacy.

It splits the file into fragments, places each fragment on a separate node, and executes unique replication, meaning each fragment is copied once. According to this approach, [Rajalakshmi, K et al. (2022)] The proposed Controlled Replication Model reduces replicating expenses and utilization of resources. Lack of service and reflection cyberattacks are minimized, and system performance such as security, accessibility, load distribution, and tolerance for failures are improved. The researcher [Sugumar, R et al. (2019)] uses the centrality measure as a location for the distribution and recovery of the information and the encoding method for fragmenting the data. Regardless of the scenario of a successful attempt, nobody is given any data.

2.5. DROPS Methodology of Node Selection using T-Colouring

In this paper, [Polu, S.K. et al. (2018)] present a graph-based method for determining the ranges using the T-Coloring methodology to predict the information nodes for placing distributed data. To ensure the greatest layer of assurance, the author [Kale, R.V. et al. (2017)] assessed the effectiveness of traditional tactics like Random and T-coloring and revised various techniques. Additionally, they refer to our concept, which has been put through extensive quality assessment tests to determine which approach would best meet the user's needs on three cutting-edge structured cloud applications. This study [K. Rajalakshmi et al. (2022)] examines and contrast several cryptographic techniques with attention to three primary methods to protect information stored in the cloud. Modelling software was developed to do the analysis and comparison, and the findings indicate that AES is the best approach regarding calculation time, storage capacity, and degree of security. The suggested stochastic diffusion search (SDS) algorithm by the author [Ramanan, M et al. (2019)] would reduce the cost of data replication.

The outcomes of these tests have demonstrated that the system will be capable of showing the usefulness of the suggested algorithm in both the replication and recovery of data. To prevent an intruder from guessing the positions of the pieces, the nodes that store the fragments are partitioned to a certain extent using graph T-coloring. The cloud-based data that has been exported must be protected, and they must be integrated with CaRP. To further prevent an intruder from speculating about the positions of the pieces, [Sivasankari, M.A. et al. (2016)] use graph T-coloring to differentiate the nodes containing the pieces by a specific range. According to the author, the file is divided into smaller pieces, maintained on a cloud server, and transmitted to clients using the protocol as needed [Divya, S.V. et al. (2018)]. If a problem happens, the fragments can pick an alternate route. The methodology delivers excellent data security using T-coloring methods.

3. System Model

This study aims to propose an Intelligent Node Selection Model (INSM) comprising centrality measures and choosing the centermost stable nodes with less energy and security cost in the network for placing the data fragments so that the access time gets improved and node failure is significantly reduced.

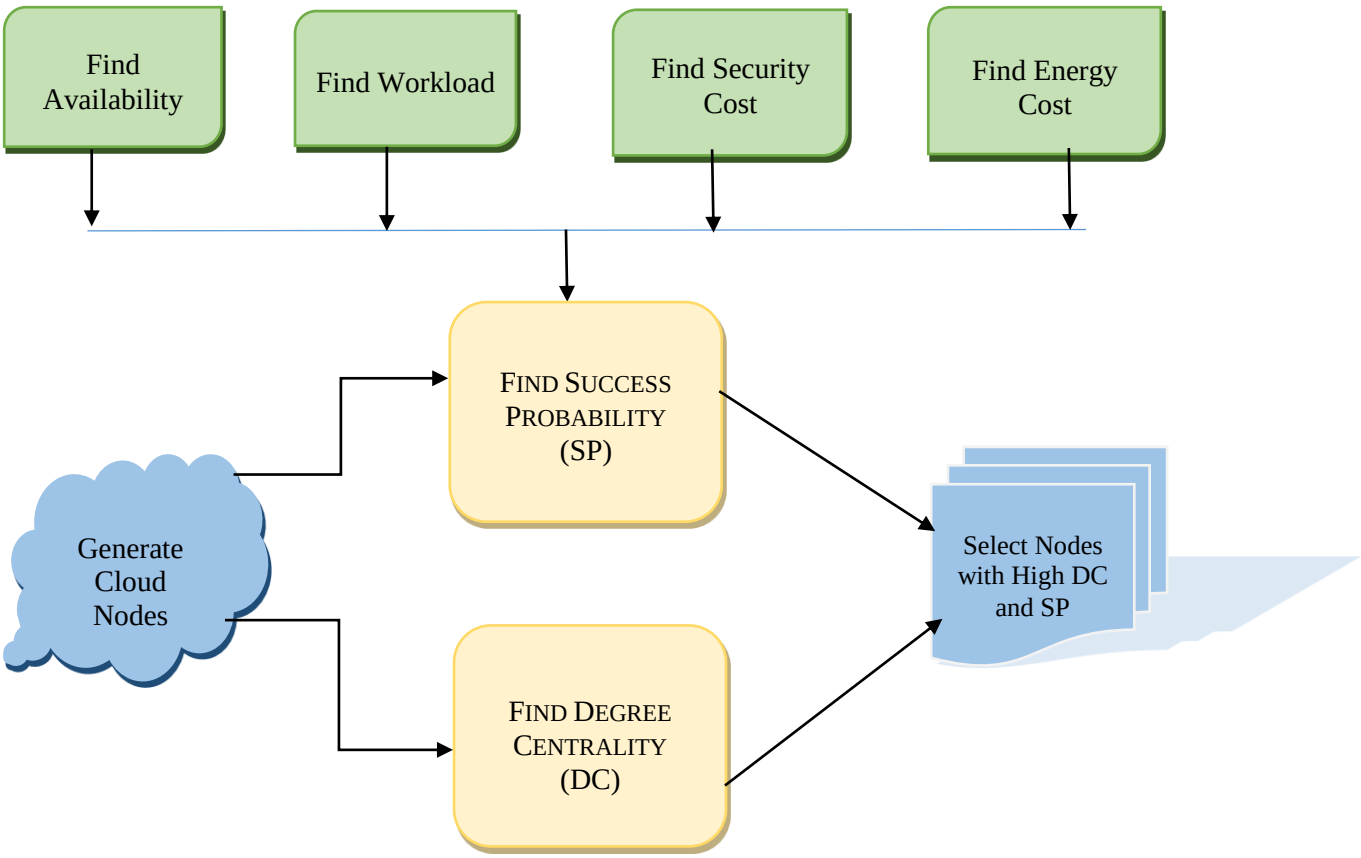


Fig. 1 Proposed intelligent node selection model (INSM)

To store the fragmented data files in the nodes in the cloud network, a node selection mechanism has to be employed where the proposed INSM model Figure 1 accepts the centrality measure and success probability of nodes as input. The centermost node always has the lesser possibility of being hacked, leaving the data to be secure aside. Also, the central nodes have improved retrieval time. Thus, nodes are selected, and data files are stored. The degree of centrality is obtained by considering every node's incoming and outgoing edges.

The ranking model ranks a node as stable based on the parameters like energy cost, security cost, workload and availability factor. Central nodes also help improve data access time and reduce the frequent infection risk. Once nodes are selected, the data fragments are placed and monitored for data popularity that needs replicated data always to be available. The centrality measure is the degree centrality of the node where the edges incident on the node is counted to calculate the centrality value. The success probability of the node takes the availability factor, Workload, security cost and energy cost of each node. Based on such inputs, the nodes with high success probability and degree centrality are measured to find the feasible nodes for selection.

3.1. Node Selection Mechanism

The centermost nodes will always be less likely to be hacked, leaving the data secure. Also, the central nodes will constantly have improved retrieval time. Thus, nodes are selected, and data files are stored. Irrespective of the central nodes selected using the degree centrality measure in this study, another challenge lies in choosing the stable nodes in the network. The node's success probability is also considered a prime factor for being the critical node for selection. The Degree Centrality of a given data node x is given.

$$DC(x) = \deg(I_d + O_d) \quad (1)$$

Where, I_d = Indegree and O_d = Outdegree

The success probability of node selection for node x is given by

$$P^t_{Success} = \sum_t Avail_x + Wl_x + \mu_s(x) + \mu_e(x),$$

$$\text{Where } t_i < t_{i+n} \quad (2)$$

Where, $Avail_x$ the amount of time the node persists without failure is, Wl_x denotes the Workload of node x,

$\mu_s(x)$ specifies the security cost and $\mu_e(x)$is the energy cost of node x. The Workload of node x and its availability is given by:

$$Wl_x^t = \sum_{t_i}^{t_{i+n}} \sum_{i=1}^{m} \alpha_i f_i \quad (3)$$

Where α_i is the size of files stored in the node x, and f_i is the access frequency of node x

$$Avail_x^t = \sum_{t_i}^{t_{i+n}} = \frac{Stab_x}{FD_x} \quad (4)$$

Where, FD_x = Failure Degree of Node x. The stability of node x is given by $0 \leq Stab_x \leq 1$. A node must have good stability and a low failure degree to ensure a higher probability of availability. The fast working of the Intelligent Node Selection Model algorithm to choose the centermost stable nodes is explained in brief.

3.2. Assumptions

Consider a CSP 'C' with 'N' data centres { DC_1, DC_2,……, DC_N }. Each Data Center DC_K Consists of 'l' data nodes, {$dcdn_1$, $dcdn_2$, ……., $dcdn_i$ }.

Each data center node is characterized by five tuples given by,

$$dcdn_i = \{ dn_{id_i}, d_{n_{rqar_i}}, dn_{ast_i}, dn_{fp_i}, dn_{bw_i} \} \quad (5)$$

Where,

dn_{id_i} is the node id,

$dn_{rqar_i}$is the request arrival rate of the data node,

dn_{ast_i} is the average service time,

dn_{fp_i} is the failure probability of the node,

dn_{bw_i} is the network bandwidth of the node, and all computations are done at time 't'.

3.3. Pseudo Code for Intelligent Node Selection Model Algorithm

Input:Degree Centrality and Highest Rank Node

```
Degree Centrality ( )
  {
  for the data node dcdn_i,
  DC(x)=deg(I_d+ O_d )
  Return DC(x)=High Value or low value;
  Arrange the degree in descending order using desc(D_c
  (dcdn_i ));
  }
Highest Rank Node ( )
  {
  at time t ,Calculate the following for each node:
  Compute Workload for each node ,Wl_x^t using
  Calculate the availability of node,Avail_x^t using
  Find out the security cost of node x given by,
  μ_s (x)^t by considering the following security states:
  Safe (S),Vulnerable (v) and Compromised (c)
  Find out the energy cost of node x given by,
  μ_e (x)^t by considering the following energy states:
  high (h_1 ),middle (h_2),low (h_3)
  Compute success probability of the node x,p^t_successx)^t using
  Rank the node x as high or low
  return rank_high or rank_low ;
  }
Node Selection ( Degree Centrality,Hightest Rank Node )
  {
  ode Selection ( Degree Centrality,Hightest Rank Node )
    {
    If (D_c (dcdn_i) ) = =HIgh && Node Rank= = Rank_HIgh
    SELECT the Nodes;
    Else
    REJECT the Nodes;
    End if
    }
  }
```

Output: Central Node, Less Access Time, Better Fault Tolerance

The Intelligent Node Selection Model (INSM) algorithm selects the best nodes for resource allocation in cloud computing. The algorithm takes input from two functions: Degree Centrality and Highest Rank Node. The Degree Centrality function calculates the importance of each node based on the number of connections it has in the network. It assigns a high or low value to each node based on its degree centrality and arranges the nodes in descending order of their degree centrality.

The Highest Rank Node function calculates the suitability of each node for resource allocation based on several factors such as Workload, availability, security cost, energy cost, and success probability. It ranks each node as high or low based on its suitability. The Node Selection function uses the Degree Centrality output and Highest Rank Node functions to select the best nodes for resource allocation. It selects the nodes with a high degree of centrality value and a high rank based on suitability. In simple terms, the algorithm selects nodes with many connections in the network that are suitable for resource allocation based on their Workload, availability, security, energy, and success probability. By selecting the best nodes, the algorithm helps to optimize resource utilization and improve the performance of the cloud computing network.

4. Results and Discussions

Node selection models can help optimize cloud costs by considering pricing models, resource utilization, and workload patterns.

Table 1. INSM node selection based on D_C and node rank

Node ID	Degree Centrality, D_c	Node Stability	Node Rank	Node Selection
5	5	4	Low	No
7	6	6	High	Yes
9	7	8	High	Yes
10	8	2	Low	No
11	9	3	High	Yes
14	14	7	High	Yes
17	10	4	High	Yes

Table 2. Descending order of node rank

Node ID	Degree Centrality , D_c	Node Stability	Node Ranking Level	Node Selection
14	14	7	High	Yes
9	7	8	High	Yes
17	10	4	High	Yes
7	6	6	High	Yes
12	9	3	High	Yes
10	8	2	Low	No
5	5	4	Low	No

By selecting cost-effective nodes based on pricing tiers or spot instances, these models enable users to use cloud resources efficiently while minimizing expenses. Node selection models in the cloud can be customized to meet specific requirements and policies. Administrators can define selection criteria based on security, compliance, geographical location, or specific application requirements. This flexibility allows for tailored resource allocation strategies and aligns with the unique needs of different cloud deployments. By leveraging node selection models, cloud providers and users can optimize resource utilization, improve performance, enhance fault tolerance, and achieve cost-effective operations in their cloud environments. Some node selection models can dynamically adapt their node selection strategies based on changing conditions or evolving system states.

This flexibility allows for real-time optimization and responsiveness to fluctuating demands or varying resource availability. It is important to note that optimality depends on the specific problem domain, objectives, and constraints. The design and implementation of an optimal selection model will heavily depend on the problem being addressed.

The fault tolerance capabilities of the model were also evident, as it incorporated redundancy and backup strategies in node selection. In the event of node failures or disruptions, the model seamlessly redistributes workloads to healthy nodes, ensuring continuous service availability and minimal impact on performance. Experiments were conducted to identify the highly ranked nodes concerning centrality and node stability. The simulation that runs for 500 seconds comprises a Cloud Service Provider with 40 data centre nodes with a bandwidth of 2 Mbps. The results obtained are listed below:

4.1. Node Selection Related to Degree Centrality and Node Ranking

Node selection in the INSM model occurs by finding the centermost nodes in terms of their degree of centrality that considers the number of edges that come towards the node and that emerges. Also, the node stability is considered to be an exact parameter for the ranking of nodes, and by taking into account both degree centrality and node stability, a threshold is fixed for the ranking of nodes as 11. If any node falls between $11 \geq N_i \geq n_i$, where n is any number above 11 and is the node, then the node is ranked High else, it is ranked Low. The nodes with High rank are selected for placing of data files. The values obtained are displayed in Table 1, and the descending order of node rank is listed in Table 2.

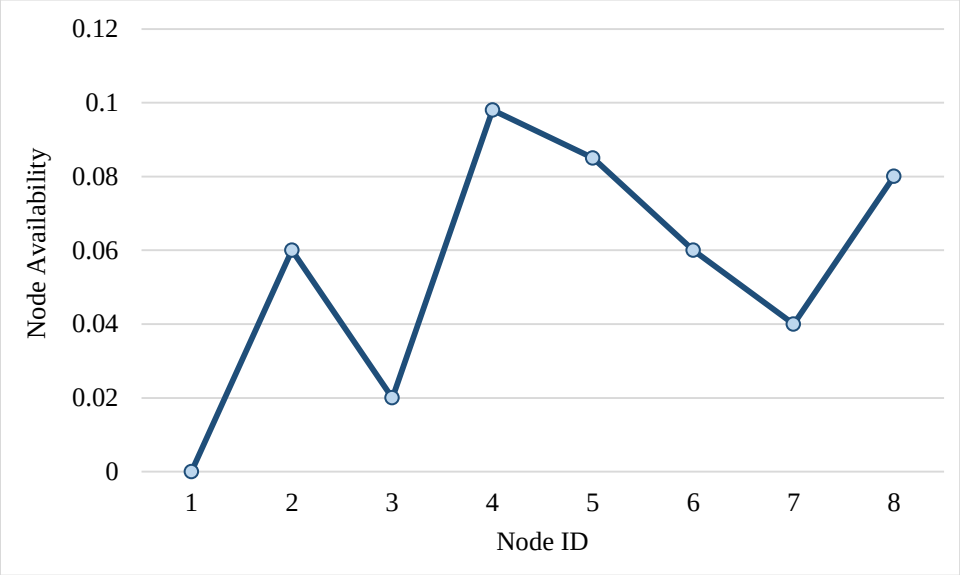


Fig. 2 Node availability with respect to failure degree and stability

4.2. Availability of Stable Nodes

In order to find a node's availability in the cloud network, the stability of the node is chosen between 0 and 1, and the failure degree, which is the duration for which the node is active, is determined. The failure degree is assumed to be 10 seconds, and as the stability factor increases, the availability of nodes also increases. As suggested by Equation 4, the availability factor of the node is calculated. The values arrived are plotted in Figure 2.

4.3. Comparative Analysis

The proposed model is compared with existing models to improve the novel method's results. Table 3 compares the existing Optimal Selection Model (OSM) with the proposed Intelligent Node Selection Model (INSM). The OSM is the most appropriate virtual machine or physical server to run a particular workload or application.

This selection process is critical for ensuring the application performs efficiently and cost-effectively. There are several factors to consider when selecting an optimal node for a workload, including Resource requirements, Availability, Network Connectivity and Cost.

Some different techniques and algorithms can be used to select an optimal node for a workload, such as rule-based methods, machine learning algorithms and optimization models. Some commonly used models for optimal selection in cloud computing include Load balancing, Decision tree-based models, Game theory-based models and Neural network-based models. While the optimal selection model provides numerous benefits in cloud computing, some disadvantages include Increased complexity, Resource wastage, Performance issues, and Cost and Security risks.

Overall, the optimal selection model can be an effective tool for managing cloud resources and optimizing performance. However, it is important to consider the potential disadvantages and carefully weigh the costs and benefits before implementation. The cost incurred for finding the degree of centrality, availability, Workload, energy cost and security cost is calculated, and cost estimation to find the better node is done. The following graphs show the various performance measures like cost estimation and success ratio.

The algorithm selects the best nodes for resource allocation based on their degree of centrality and suitability, which helps to optimize resource utilization and improve the performance of the cloud computing network. By selecting nodes based on proximity, availability, and performance characteristics, node selection models can improve the performance of cloud applications and services. For example, they can dynamically choose nodes with lower latency or higher bandwidth for data-intensive tasks, reducing response times and improving user experience.

Node selection models can consider fault tolerance strategies to enhance the reliability and availability of cloud services. By selecting redundant or backup nodes, these models can mitigate the impact of node failures. They can also dynamically redistribute workloads to healthy nodes, ensuring continuous service availability and minimizing disruptions. In cloud computing environments, various node selection models determine which nodes should be selected for specific tasks or computations. These models aim to optimize resource allocation, maximize efficiency, and ensure high performance in the cloud network. Here are some of the commonly used node selection models.

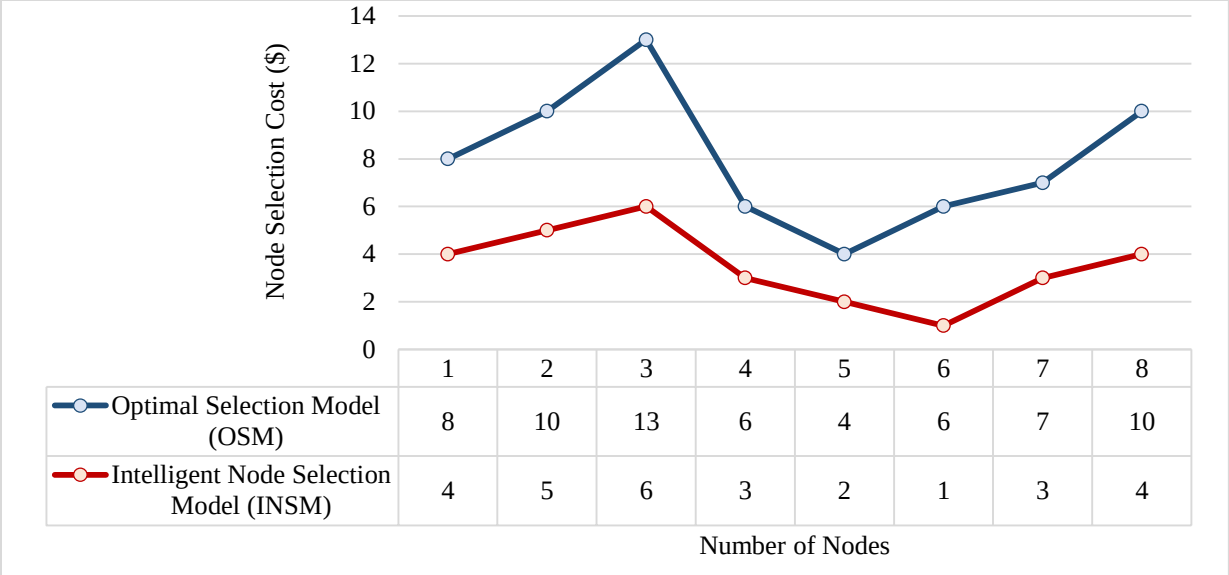
Table 3. Comparison between the optimal selection model and intelligent node selection model

Sl. No.	Existing Optimal Selection Model (OSM)	Proposed Intelligent Node Selection Model (INSM)
1	This algorithm involves a complex algorithm that requires the computation of several metrics, such as degree centrality, availability, security cost, energy cost, and success probability. This complexity can increase processing time, making it difficult to scale the model for large-scale cloud environments.	The algorithm selects the best nodes for resource allocation based on their degree of centrality and suitability, which helps to optimize resource utilization and improve the performance of the cloud computing network.
2	This model relies on accurate data inputs, such as Workload, availability, security state, energy state, and success probability. Any errors or inaccuracies in these inputs can lead to incorrect node selections, resulting in degraded system performance or security vulnerabilities.	The algorithm considers the security cost of each node and ranks them based on their security state. This helps to ensure that the nodes selected for resource allocation are secure and less vulnerable to cyber-attacks.
3	The optimal selection model is designed based on a specific set of metrics and assumptions, and it may not be adaptable to changes in the cloud environment or Workload. For example, changes in the network topology, resource availability, or security threats may require updates to the model, which can be difficult and time-consuming.	The algorithm considers the energy cost of each node and ranks them based on their energy state. This helps to ensure that the nodes selected for resource allocation are energy-efficient and consume less power, which can reduce the overall energy consumption of the cloud computing network.
4	It focuses on a limited set of metrics, such as degree centrality and availability, and it may not account for other factors that may impact node selection, such as cost, performance, or compliance requirements. This limited scope can result in suboptimal node selections and reduced system efficiency.	The algorithm considers the availability of each node and ranks them based on their availability state. This helps to ensure that the nodes selected for resource allocation are highly available and can provide reliable services to users.
5	This model may introduce bias in resource allocation, favouring specific nodes or regions over others. This bias can result in resource underutilization or overutilization, leading to increased costs or decreased performance.	The algorithm can select nodes in both small and large-scale cloud computing networks, making it highly scalable.

The Round Robin model follows a simple and cyclic approach, where tasks are sequentially assigned to nodes. Each task is allocated to the next available node in a circular order. This model ensures fairness by distributing the Workload evenly across the available nodes. The Least Loaded model selects the node with the least current Workload or utilization. It considers CPU utilization, memory usage, or network bandwidth factors to determine the node's load. By selecting the least loaded node, this model aims to achieve load balancing and avoid overburdening any single node.

The Random Selection model randomly chooses a node from the available options for task assignment. This model does not consider specific metrics or load information but relies on chance. While simple to implement, this approach may not ensure optimal resource allocation or load balancing. Genetic algorithms are used to optimize node selection based on a set of predefined criteria. These models involve creating a population of potential node selections and using genetic operators such as crossover and mutation to evolve and refine the selection process. Fitness functions evaluate and compare different node selections based on desired objectives, such as minimizing response time or maximizing resource utilization.

Cost-optimization models focus on minimizing the financial expenses associated with executing tasks in the cloud environment. These models consider node pricing, resource utilization, and service-level agreements. By balancing the cost and performance trade-offs, the model selects nodes that offer the best value for money. Energy-efficient node selection models aim to minimize the power consumption of the cloud infrastructure. They consider the energy efficiency characteristics of different nodes, such as CPU power usage, memory utilization, and network traffic.



	1	2	3	4	5	6	7	8
Optimal Selection Model (OSM)	8	10	13	6	4	6	7	10
Intelligent Node Selection Model (INSM)	4	5	6	3	2	1	3	4

Fig. 3 Node selection cost with different stages

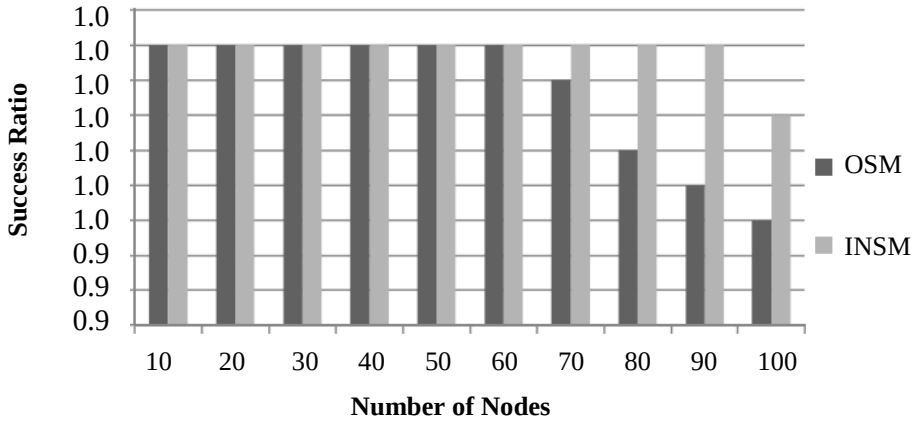


Fig. 4 Success ratio vs no.of nodes

By selecting energy-efficient nodes, these models contribute to reducing the cloud system's overall energy consumption and environmental impact. It is important to note that the suitability of a node selection model depends on the specific requirements and constraints of the cloud computing environment. Different models may be more appropriate for specific scenarios, workload types, or performance objectives. Hence, it is crucial to evaluate and choose the node selection model that aligns with the specific needs of the cloud deployment.

4.3.1. Cost Estimation of Nodes

The different cost estimations at various stages of Intelligent Node Selection (INSM) Model is compared with the existing node selection mechanism, the Optimal Selection Model (OSM).

INSM model outperforms the existing OSM model and the average cost of node selection is approximately 1.1% higher than the Optimal Selection model as the cost of nodes in the INSM model is incurred for calculating the degree centrality, node ranking to find out the success probability, security cost and energy cost. The various stages and the average cost of INSM and OSM are depicted in Figure 3, which is given by:

$$\text{Node}_{\text{Cost}} = \sum((C_1 * \text{DC}(x)), (C_2 * \text{Avail}(x)), (C_3 * \text{Wl}(x)), (C_4 * \mu_e(x), (C_5 * \mu_s(x))) \quad (6)$$

4.3.2. Success Ratio

The success ratio of every node in the network is related to how far the node is safe and finishes the assigned task. This success probability is always equal to 1. If the node is vulnerable or compromised, the success probability is related to $1/\sqrt{2}$ or $1/\sqrt{3}$. An environment with Seven safe nodes and two vulnerable and compromised nodes are generated to find the success ratio. Figure 4 depicts that the INSM model outperforms the existing optimal selection model and other

existing models with a success probability of 1 in most cases, even after an increase in the number of nodes at various stages in the cloud network.

5. Conclusion and Future Work

The node selection process in the cloud has become an essential factor as not all nodes in the network can be believed and chosen for any valuable purpose like computation and storage. The detailed system architecture described above illustrates the various entities and their roles in safeguarding the data in the cloud premise. The degree of centrality is obtained by considering every node's incoming and outgoing edges. The ranking model ranks a node as stable based on the parameters like energy cost, security cost, workload and availability factor. Central nodes also help improve data access time and reduce the frequent infection risk. Once nodes are selected, the data fragments are placed and monitored for data popularity that needs replicated data always to be available. The INSM model is used to find the stable centre of most nodes in the network so that the data access time is reduced and fetching data from those centres of most nodes has become more accessible. The INSM model mainly focuses on the degree of centrality and node ranking mechanism for the selection of nodes in order to place the primary copy of the data. The proposed model has improved the client success ratio, giving the customer a sense of satisfaction. The cost estimation is high as it includes more parameters than the existing one. Thus the node access time is better, and the failure of nodes also gets reduced so that the user will have continued access to data safely. Calculating the degree centrality of all the nodes in a network takes $\Theta(V^2)$ time and for edges $\theta(E)$ where are the vertices and E are the edges. In Future, the Intelligent Node Selection model can be extended by incorporating more centrality measures like eccentricity, betweenness, etc., so that more specific central nodes could be obtained. The whole system could be implemented in different architectures like fat-tree, D-cell.

References

[1] Ahmed E. Abdel Raouf, Nagwa L. Badr, and Mohamed F. Tolba, "Dynamic Data Reallocation and Replication over a Cloud Environment," *Concurrency and Computation: Practice and Experience*, vol. 30, no. 13, 2018. [CrossRef][Google Scholar][Publisher Link]

[2] Ketan Bhardwaj et al., "Addressing the Fragmentation Problem in Distributed and Decentralized Edge Computing: A Vision," *In IEEE International Conference on Cloud Engineering*, pp. 156-167, 2019. [CrossRef][Google Scholar][Publisher Link]

[3] R. Dhaya, R. Kanthavel, and Kanagaraj Venusamy, "Cloud Computing Security Protocol Analysis with Parity-Based Distributed File System," *Annals of Operations Research*, pp.1-20, 2021. [CrossRef][Google Scholar][Publisher Link]

[4] K. Dhinakaran et al., "Enhance Hybrid Cloud Security using Vulnerability Management," *In International Conference on Soft Computing and Pattern Recognition*, vol. 614, pp. 480-489, 2016. [CrossRef][Google Scholar][Publisher Link]

[5] S. V. Divya, R. S. Shaji, and P. Venkadesh, "An Efficient Data Storage and Forwarding Mechanism using Fragmentation-Replication and DADR Protocol for Enhancing the Security in Cloud," *Journal of Computational and Theoretical Nanoscience*, vol. 15, no. 1, pp. 111-120, 2018. [CrossRef][Google Scholar][Publisher Link]

[6] K. Rajalakshmi, M. Sambath, and L. Joseph, "Improving File Storage Mechanism using Intelligent Data Fragmentation Model (IDFM) Algorithm and Providing Confidentiality of Data in Cloud Computing Environment," *International Journal of Engineering Trends and Technology*, vol. 71, no. 3, pp. 130-142, 2023. [CrossRef][Publisher Link]

[7] Suji Gopinath, and Elizabeth Sherly, "A Comprehensive Survey on Data Replication Techniques in Cloud Storage Systems," *International Journal of Applied Engineering Research*, vol. 13, no. 22, pp. 15926-15932, 2018. [Google Scholar][Publisher Link]

[8] Nurudeen Mahmud Ibrahim, and Anazida Zainal, "A Feature Selection Technique for CLOUD IDS using Ant Colony Optimization and Decision Tree," *Advanced Science Letters*, vol. 23, no. 9, pp. 9163-9169, 2017. [CrossRef][Google Scholar][Publisher Link]

[9] Nesrine Kaaniche, and Maryline Laurent, "Data Security and Privacy Preservation in Cloud Storage Environments Based on Cryptographic Mechanisms," *Computer Communications*, vol. 111, pp. 120-141, 2017. [CrossRef][Google Scholar][Publisher Link]

[10] Mohammad Javad Abbasi, and Mehrdad Mohri, "Scheduling Tasks in the Cloud Computing Environment with the Effect of Cuckoo Optimization Algorithm," *SSRG International Journal of Computer Science and Engineering*, vol. 3, no. 8, pp. 1-9, 2016. [CrossRef][Google Scholar][Publisher Link]

[11] Hakim Jebari, Siham Rekiek, and Kamal Reklaoui, "Improvement of Nature-Based Optimization Methods for Solving Job shop Scheduling Problems," *International Journal of Engineering Trends and Technology*, vol. 71, no. 3, pp. 312-324, 2023. [CrossRef][Google Scholar][Publisher Link]

[12] Rahul Vishwanath Kale, Bharadwaj Veeravalli, and Xiaoli Wang, "Design and Performance Characterization of Practically Realizable Graph-Based Security Aware Algorithms for Hierarchical and Non-Hierarchical Cloud Architectures," *In International Conference on Frontier Computing*, Singapore, vol. 464, pp. 392-402, 2017. [CrossRef][Google Scholar][Publisher Link]

[13] Samira Kanwal et al., "Head Node Selection Algorithm in Cloud Computing Data Center," *Mathematical Problems in Engineering*, vol. 2021, 2021. [CrossRef][Google Scholar][Publisher Link]

[14] Mihir Manek et al., "Cloud Oriented Distributed and Encrypted File Storage (CODE-FS)," *In Fourth International Conference on Computing Communication Control and Automation*, pp. 1-5, 2018. [CrossRef][Google Scholar][Publisher Link]

[15] Gunasekaran Manogaran, Chandu Thota, and M. Vijay Kumar, "Meta Cloud Data Storage Architecture for Big Data Security in Cloud Computing," *Procedia Computer Science*, vol. 87, pp.128-133, 2016. [CrossRef][Google Scholar][Publisher Link]

[16] N. Mansouri, M. M. Javidi, and B. Mohammad Hasani Zade, "A CSO-Based Approach for Secure Data Replication in Cloud Computing Environment," *The Journal of Supercomputing*, vol. 77, no. 6, pp. 5882-5933, 2021. [CrossRef][Google Scholar][Publisher Link]

[17] R. Navinkumar, and R. Ramamoorthy, "Improve Resource Allocation for Cloud Computing Environment," *International Journal of P2P Network Trends and Technology*, vol. 6, no. 2, pp. 12-19, 2016. [CrossRef][Publisher Link]

[18] Muzammil H. Mohammed, and Faiz Baothman, "A Study on Methods to Improve Efficiency of Cloud Computing," *International Journal of Computer and Organization Trends*, vol. 10, no. 2, pp. 13-17, 2020. [CrossRef] [Publisher Link]

[19] Imad Eddine Miloudi, Belabbas Yagoubi, and Fatima Zohra Bellounar, "Dynamic Replication Based on a Data Classification Model in Cloud Computing," *In International Symposium on Modelling and Implementation of Complex Systems*, vol. 156, pp. 3-17, 2020. [CrossRef][Google Scholar][Publisher Link]

[20] W. Delishiya Moral, and B. Muthu Kumar, "Improve the Data Retrieval Time and Security through Fragmentation and Replication in the Cloud," *In International Conference on Advanced Communication Control and Computing Technologies*, pp. 539-545, 2016. [CrossRef][Google Scholar][Publisher Link]

[21] Andrea Li, "Privacy, Security and Trust Issues in Cloud Computing," *SSRG International Journal of Computer Science and Engineering*, vol. 6, no. 10, pp. 29-32, 2019. [CrossRef][Google Scholar][Publisher Link]

[22] K. Rajalakshmi, M. Sambath, and Linda Joseph, "Towards Improving Cloud Security and Performance by using Proposed Controlled Replication Model," *In International Conference on Computer, Power and Communications*, pp. 12-16, 2022. [CrossRef][Google Scholar][Publisher Link]

[23] S. Ravichandran, R. Benjohnson, and K. Ramanathan, "Design and Development of Honey Bee Behavior Excited Modern Quality Constructed Entry Controller using Bell-Lapadula Paradigm in Cloud Computing Methodology," *International Journal of Recent Engineering Science*, vol. 7, no. 2, pp. 1-7, 2020. [CrossRef][Google Scholar][Publisher Link]

[24] Sandeep Kumar Polu, "Security Enhancement for Data Objects in Cloud Computing," *International Journal for Innovative Research in Science and Technology*, vol. 5, no. 6, pp. 18-21, 2018. [Google Scholar][Publisher Link]

[25] I. Lakshmi, "A Review On Security In Mobile Cloud Computing," *SSRG International Journal of Mobile Computing and Application*, vol. 6, no. 2, pp. 4-11, 2019. [CrossRef][Publisher Link]

[26] Muzammil H Mohammed, and Faiz Baothman, "Intelligent Workload Management of Computing Resource Allocation For Mobile Cloud Computing," *International Journal of Computer & Organization Trends*, vol. 5, no. 2, pp. 30-39, 2015. [CrossRef][Google Scholar][Publisher Link]

[27] T. Prasuna et al., "A Novel Approach for Improved Data Replication using HDFS," *In 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology*, pp. 854-859, 2018. [CrossRef][Google Scholar][Publisher Link]

[28] K. Rajalakshmi, K. Ramesh, and P. N. Renjith, "Comparative Study of Cryptographic Algorithms in Cloud Storage Data security," *In 2nd International Conference on Intelligent Technologies*, pp. 1-7, 2022. [CrossRef][Google Scholar][Publisher Link]

[29] Richa Kunal Sharma, and Nalini Kant Joshi, "Security and Privacy Problems in Cloud Computing," *International Journal of Computer and Organization Trends*, vol. 9, no. 4, pp. 30-39, 2019. [CrossRef][Publisher Link]

[30] M. Ramanan, and P. Vivekanandan, "Efficient Data Integrity and Data Replication in the Cloud using Stochastic Diffusion Method," *Cluster Computing*, vol. 22, no. 6, pp. 14999-15006, 2019. [CrossRef][Google Scholar][Publisher Link]

[31] Anand Prakash Singh, and Arjun Choudhary, "Approach for Ensuring Fragmentation and Integrity of Data in SEDuLOUS," *In Proceedings of Second International Conference on Computing, Communications, and Cyber-Security*, vol. 203, pp. 857-869, 2021. [CrossRef][Google Scholar][Publisher Link]

[32] M. A. Sivasankari, M. D. Abirami, and M. K. Ayesha, "Division and Replication of Data in Cloud for Optimal Performance and Security using Fragment Placement Algorithm," *International Research Journal of Advanced Engineering and Science*, vol. 1, no. 4, pp. 57-63, 2016.[Google Scholar][Publisher Link]

[33] R. Sugumar, A. Rajesh, and R. Manivannan, "Performance Analysis of Fragmentation and Replicating Data Over Multi-clouds with Security," *In International Conference on Computer Networks and Communication Technologies*, Singapore, vol. 15, pp. 1031-1040, 2019. [CrossRef][Google Scholar][Publisher Link]

[34] Fursan Thabit et al., "A New Lightweight Cryptographic Algorithm for Enhancing Data Security in Cloud Computing," *Global Transitions Proceedings*, vol. 2, no. 1, pp. 91-99, 2021. [CrossRef][Google Scholar][Publisher Link]