

Rider Optimization Algorithm Based Optimal Cloud Server Selection in E-Learning

R. Soundhara Raja Pandian* and C. Christopher Columbus

Department of Computer Science and Engineering, PSN College of Engineering and Technology, Melathediyoore, Tamil Nadu, 627152, India

*Corresponding Author: R. Soundhara Raja Pandian. Email: rsoundhararaja@gmail.com

Received: 30 January 2022; Accepted: 06 March 2022

Abstract: Currently, e-learning is one of the most prevalent educational methods because of its need in today's world. Virtual classrooms and web-based learning are becoming the new method of teaching remotely. The students experience a lack of access to resources commonly the educational material. In remote locations, educational institutions face significant challenges in accessing various web-based materials due to bandwidth and network infrastructure limitations. The objective of this study is to demonstrate an optimization and queuing technique for allocating optimal servers and slots for users to access cloud-based e-learning applications. The proposed method provides the optimization and queuing algorithm for multi-server and multi-city constraints and considers where to locate the best servers. For optimal server selection, the Rider Optimization Algorithm (ROA) is utilized. A performance analysis based on time, memory and delay was carried out for the proposed methodology in comparison with the existing techniques. The proposed Rider Optimization Algorithm is compared to Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and Firefly Algorithm (FFA), the proposed method is more suitable and effective because the other three algorithms drop in local optima and are only suitable for small numbers of user requests. Thus the proposed method outweighs the conventional techniques by its enhanced performance over them.

Keywords: Optimization; queuing; slot selection; server selection; rider optimization algorithm

1 Introduction

Computers and computing devices-based e-learning technologies play a significant role in the educational, business and industrial sectors. In India, all the educational institutions could not provide quality education uniformly. Few top-class institutions such as the Indian Institutes of Technology (IIT), Indian Institute of Science (IISc), and Indian Institute of Management (IIM's), National Institutes of Technology (NITs) have good teaching and computing facilities to avail high-quality education [1]. Nevertheless, they establish barely 5% of entire organizations. Now a day, modern learning technologies have their advantages and drawbacks, but traditional learning technologies have several advantages. The



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

use of advanced computing, media and Information and Communication Technologies (ICT) is critical [2]. Recent studies estimated that the quantity of technical information available on the web is doubling every two years. However, the syllabuses of higher education institutions have not been updated periodically in practice. Better services at cheaper costs can be obtained by utilizing cloud computing, hence online classes are increasing day by day. The National Programme on Technology Enhanced Learning (NPTEL) is the largest online storehouse in the field of engineering and other related areas [3–6]. Most of the educational institutions move their classes online [7,8] due to unexpected natural disasters time or some pandemic situations like Covid-19.

The need for online education is growing daily [9]. In recent times, functionalities in E-learning platforms are updated drastically such as online discussions, doubt clarifications and engaging students in parallel communication [10]. Most of the premier universities are providing online education by availing facilities from the service providers. To minimize the cost for availing infrastructure from the service providers, universities should have appropriate network and server setup [11]. In online educational platforms, data transmission plays a vital role. During data transmission, traffic and time required for acquiring data are major issues. To avoid these issues, an optimal cloud server selection algorithm is developed and utilized [12,13].

The organizational flow of the paper includes the following section that contributes to the review based on conventional cloud-based e-learning techniques. Consequently in the proposed work, the elaborative demonstration of the proposed work is presented. In the fourth section, the result and simulation were performed and the final section concludes the overall work done.

2 Related Works

This section deals with the evaluation of related works based on the conventional cloud computing-related e-learning methods and their drawbacks Masud et al. [14] have proposed a cloud-based E-learning strategy for the enhancement in the learning process. The working of the proposed system is run by removing the complications present in the learning process. Moreover, the work also deals with adapting cloud security and access control. The needs for the learning were met by utilizing this cloud processing. Cloud-based e-learning results in more benefits in terms of initiating, creation, deployment and more. But the work can be further enhanced by inducing data security and technical requirements with the proposed work.

Elmasry et al. [15] have established e-learning based on cloud computing. To enable the higher studies without terminating despite unavoidable situations such as Covid-19 the study has been proposed. E-learning resides the better path for ensuing higher studies in virtual education without distribution using cloud computing services and mainly helps in distance learning. But the work has to be further enhanced by adding hybrid cloud to the e-learning for better learning efficiency and cost-effectiveness.

Ssemugabi et al. [16] have established a comparative study of web-based e-learning applications. Two Usability Evaluation Method (UEM) with the Heuristics evaluation has been carried out for the profit of the e-learning application. The drawbacks in usability have been clearly emphasized with the UEM. To improve the efficiency in developing web-based learning, an end-user survey of the learner population has been compared to the HE results to evaluate the learning application using evaluation criteria developed for e-learning applications. The evaluation over the Info3Net results in improving the usability of the e-learning application. But Info3Net holds some drawbacks as that must be rectified and the evaluation methods should be redeveloped concerning the application.

Ray [17] has proposed a detailed study of the application and importance of web-based e-learning. The work has been developed to contribute to the importance of e-learning, its objective and has been convinced for higher quality education and to practice for better use. The survey about the different approaches used for

W-Bel and to blow out the benefits acquired by using the web-based e-learning. Further enhancement in the work can be done by utilizing cloud-based techniques.

Barbierato et al. [18] have proposed a design of hybrid cloud-based distance learning architecture. A Vectorial Auto-Regressive Fractional (Integrated) Moving Average-based approach has been utilized for the evaluation and management of e-learning. Also, enables decision-making with cost-efficient and management of the distant learning procedure. Mainly tends to access the material features for the learning process with the information management and security. Further enhancement in the work has to be established by inducing a sophisticated approach for further enhancement in the features in learning.

Kalushkov et al. [19] have implemented an e-learning module for efficiency improvement in the virtual learning process. A pseudo-cloud hosted e-learning strategy has been utilized for the development measure in the e-learning strategy. This methodology has been known to be easier and more profitable for the distant learning process. Also, includes functional benefits and information to be accessed during the learning. The work can be enhanced by utilizing an integration of collaborative learning with the proposed strategy.

Vanjulavalli et al. [20] have proposed an effective tool for the cloud-based e-learning strategy. A web 2.0 and the cloud have been integrated to result in the conventional application in e-learning in connection to the internet technology. But the proposed work has relied on the virtualization procedure with the cloud, application, and service model layers. Mainly has been implemented to enable high performance in improving the learning procedure and for security concerns with storage and precise computing. The drawback of the work relies on improving the performance of the application used for the e-learning procedure.

The following section contributes a collaborative explanation of the proposed methodology.

3 Proposed Methodology

This study focusing the implementation of a cloud-based e-learning educational platform in India. For example, we can use the same methodology for online-based e-learning educational platforms like Massive Open Online Courses (MOOC), National Programme on Technology Enhanced Learning, online degree programs via IIT Madras. MOOC and NPTEL help deliver quality education to students, irrespective of their geographical location. In this regard, the IITs have configured five centralized servers initially with network optimization models which are placed in IIT Bombay, IIT Guwahati, IIT Madras, IIT Delhi, and IIT Roorkee. This project implementation provides rural educational institutions have special significance/ remote area students. It can be possible to establish the same cloud models in major cities in India and server availability can be increased based on the requirements in near future. The above-mentioned IITs are the prime locations to host the center and connect with all places in India. In any cloud computing online platform, server selection is an important issue, and high-end configuration servers are required. To balance data bandwidth utilization, Cloud servers must be linked to load balancers simultaneously. If at all feasible, the servers should be connected in a mirror arrangement, which will increase operational stability and eliminate hazards. A cloud computing system is depicted in Fig. 1 as a network of servers that links different users. Assigning different users distinct slots or priorities for attending different courses is a huge difficulty. It's not unusual for many distant users to connect to the server at the same time in order to access the same apps. In this situation, service will be provided using the first-in, first-out (FIFO) method (first in, first out). Registrations can be enabled ahead of time to aid in the prioritisation of service delivery. We focus on networking optimization for utilizing all e-learning recourses effectively.

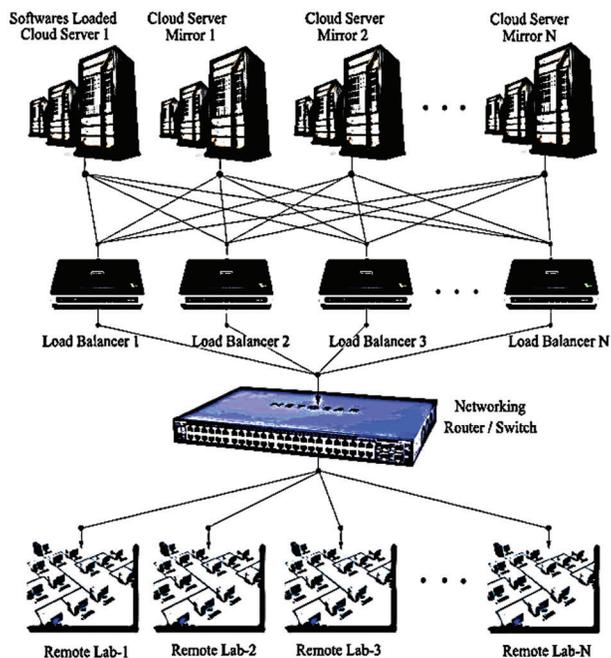


Figure 1: Schematic diagram of network connection in a cloud model

3.1 Rider Optimization Algorithm Based Cloud Server Selection

The attacker, overtaker, follower, and bypass rider are the four different groups adapted in ROA. Each group has a different approach to achieving the goal. The bypass rider's goal is to reach the target by avoiding the leader's path. The following group tries to keep up with the leader in axis. The over-taker group determines its position to reach the target via evaluating the leader's closest places. The attacker group adapts the position of the leader to accomplish the destination with the use of utmost speed. The main objective of ROA is to select an optimal server based on the nearest distance and server usage for online education. In India, due to Covid-19, all the educational institutions were closed and most of them are providing classes in online mode. In the future, most of the education institutes will motivate for online classes; because this mode of study is cost-effective and also have the flexibility to learn anytime anywhere around the world.

Hence, the mandate for web communication propagates continuously, however the accessible assets do not enhance at the similar level. High-end server system and web arrangement are main and instant essentials for educational institutes that are offering numerous webbased uses to their pupils.

The current study uses optimization and queuing techniques to allocate optimal servers and slots to users for accessing cloud applications [12]. The proposed method provides the optimization and queuing algorithm for multi-server and multi-city constraints and also considers the identification of optimal server locations. For optimal server selection process, in this paper rider optimization algorithm [21] is utilized (ROA). The rider optimization algorithm has several applications such as document clustering, enhanced video super-resolution, webpage re-raking and resource allocation [22,23]. The proposed methodology can be implemented for NPTEL and also proposed methodology diagram is exposed in Fig. 2. Initially, the NPTEL registered users can send the request to the cloud interface. Cloud interface transfers the request-to-request buffer. The request buffer collects all the requests and queuing [24–26]. Then, these requests are given to the optimal server selection process. In this stage, optimal servers are selected using ROA based on the nearest distance, server usage and time.

To avoid traffic, several mirror servers can be placed. To establish their judgment, the ROA evaluates a few riding groups that travel to the same location to win the race, to formulate their opinion [27,28]. The overall riders is divided into four sets, with certain riders in the respective group is chosen equal to the overall riders. Fig. 3 illustrates the ROA principle. To attain the goal, each group employs a different strategy.

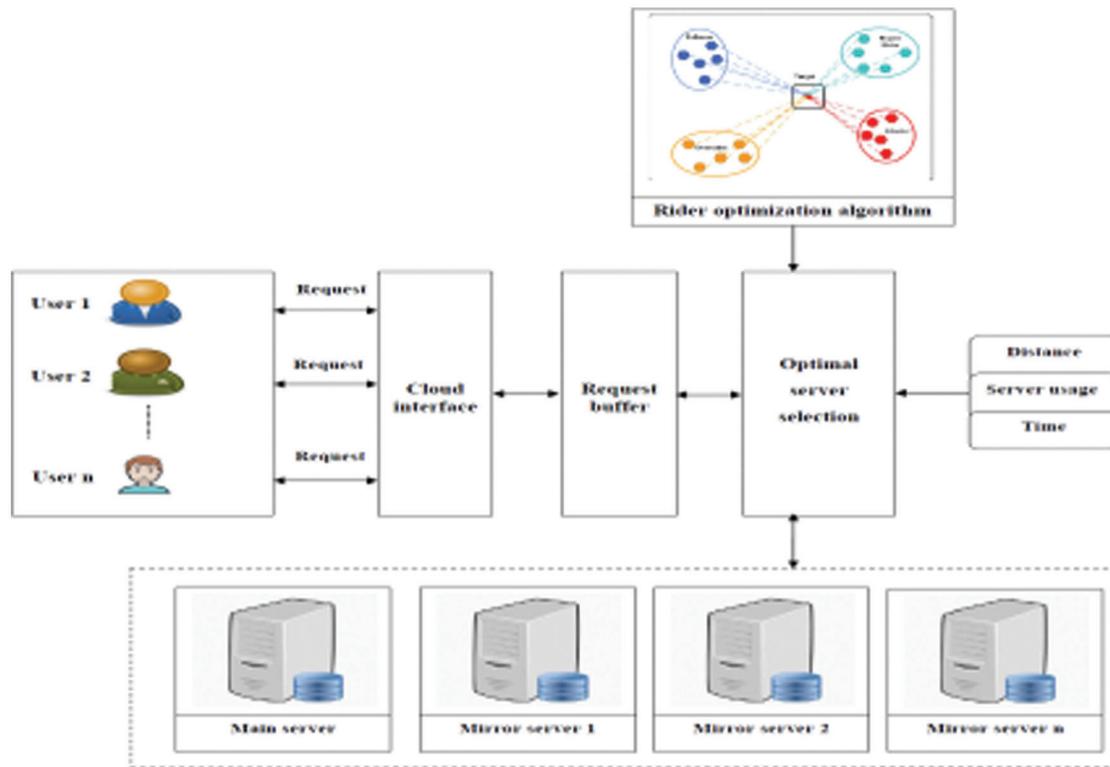


Figure 2: Image of the overall concept of the proposed methodology

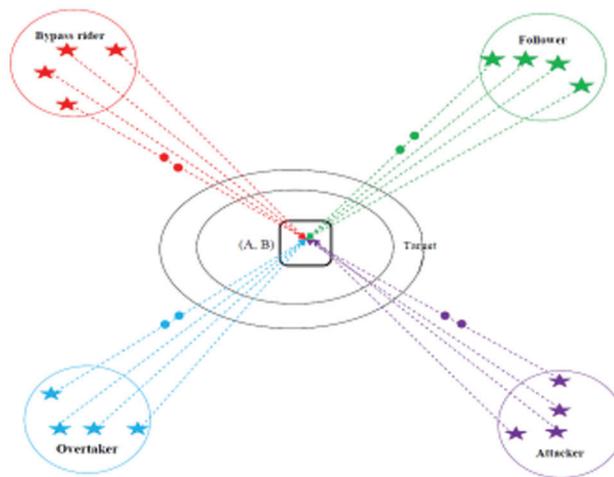


Figure 3: Schematic diagram for the principle of the rider optimization algorithm

Identification of ideal server via ROA: Data communication is a essential practice in online courses. To achieve this, the ideal server is needed to be chosen via ROA. The succeeding steps are involved in the server chosen method.

Step 1: Result encrypting: For all optimization techniques, result encrypting is a significant process. The result contains the amount of servers and the amount of user appeals. During the early phase, the server appeal is selected arbitrarily. The span of the answer is identical as the amount of user appeals. The result

is deliberated as the rider. Subsequent to the initialization of cluster sets, rider factors for instance routing, gear, accelerator, and brake are modified. [Tab. 1](#) provides a model result organization.

Table 1: Sample solution encoding

Cloud server	CS ₁	CS ₂	CS ₃	CS ₄
User request				
U ₁	1	0	0	0
U ₂	0	1	0	0
U ₃	0	0	1	0
U ₄	0	0	0	1
U ₅	0	1	0	0
U ₆	1	0	0	0

In [Tab. 1](#), 4 servers and 6 users are employed. The user 1 appeal is allocated to Cloud Servers CS₁, user 2 appeal is allocated to CS₂ and user 6 appeal is allocated to CS₁. The early result equation is provided as

$$I_t = \{I_t(i,j)\}, 1 \leq i \leq S; 1 \leq j \leq Q \quad (1)$$

where I represent a rider's number, Q represents an optimization problem, t represents a time instant, I_t represent the position of i^{th} rider at time t in the interval of (i,j).

Step 2: Fitness Estimation: The fitness of every rider is estimated while the group riders were primed. Here the multi-objective function is operated with the objectives such as time, distance, and server usage.

$$Fit = \min(\text{time}, \text{distance}, \text{serverusage}) \quad (2)$$

Step 3: Modernise the bypass rider: The Bypass Riders evade the public route, however they will not adopt the prominent riders, therefore the eminence modernise for this is allotted to every base at arbitrary as follows

$$I_{t+1}^{By}(i,j) = \tau[I_t(\gamma,j) * \mu(j) + I_t(\rho,j) * [1 - \mu(j)]] \quad (3)$$

where τ represents the random number [0,1], γ represents range rate between 1 and R, is the random numeral. ρ tends to the number that select a random value among 1 and R at random. Moreover, μ denotes the random number among 0 and 1. It will be subject to the data size, such as $1 \times Q$.

Step 4: Modernise the location of the follower: By chasing the chief rider's position, a follower modernises its location so as to help the chief rider in attaining the endpoint as fast as probable. The organise selector is employed by a follower to alter the position reliant on the Q values.

$$I_{t+1}^F(i,j) = I^L(L,K) + \left[\cos\left(A_{i,k}^t\right) * I^L(L,K) * d_i^t \right] \quad (4)$$

where K is the sector coordinate and I_{t+1}^F is represents the follower's position, S^L signify the spot of the bypass rider and L signifies the index of bypass rider, $A_{i,k}^t$ characterizes the steering angle of the i^{th} and k^{th} coordinates and d_i^t embodies the distance to travel the i^{th} coordinate.

Step 5: Modernise the location of the overtaker: The over-taker is modernized according with 3 vital strictures: route pointer, comparative success level and coordinate sector. The present and the chief rider positions are utilized to modernise the location of the overtaker according with the subsequent equation,

$$I_{t+1}^{OT}(i, k) = I_t(i, k) + [S_t^D(i) * I^L(L, K)] \quad (5)$$

At this point $S_t(i, k)$ denotes the rider location that contains the i^{th} rider and k^{th} director and $S_t^D(i)$ signifies the route pointer of i^{th} rider prompt of t . The route pointer demonstrates the route of the rider; the route pointer is delivered as,

$$S_t^D(i) = \left[\frac{2}{1 - \log(R_t^s(i))} \right] - 1 \quad (6)$$

where $S_t^D(i)$ denotes the directional indicator of i^{th} coordinator instant of t and $R_t^s(i)$ represent the success rate of the i^{th} rider with instant time t .

Step 6: Modernise the attacker location: The attacker try to proceed the location of the chief drives via the identical informing procedure as the follower. Though, instead of modernising the stated values, the attacker modernises all the values in the coordinate. The attacker location is updated as

$$I_{t+1}^{Attacker(i,j)^L(L,j)} [\cos(A_{i,j}^t) * S^L(L,j) * d_i^t] \quad (7)$$

where $I^L(L, j)$ denotes the rider leading position in the j^{th} coordinates and $(A_{i,j}^t)$ denotes the rider steering angle in the i and j^{th} coordinates.

Step 7: Closure conditions: The procedure is performed until the finest server is identified. When the ideal server is identified, at that point the technique will be concluded. A user is allotted to the servers that were selected.

Tab. 2 Displays the Pseudocode of ROA.

Table 2: Pseudocode of ROA

```

initialize the population
initialize the rider parameters
Find fitness calculation using Eq. (2)
While  $t < T_{off}$ 
  for  $i = 1$  to  $R$ 
    update the position of bypass rider with Eq. (3)
    update the position of follower using Eq. (4)
    using the equation, update the attacker's position of the overtaker (5)
    Using the equation, update the position of attacker position (6)
  rank the rider based on the fitness function
  Return XL
   $t = t + 1$ 
  end for
end while

```

4 Results and Discussion

The present study focuses on the effective usage of servers based on the demand from the user. The method combines optimization and queuing concepts to allocate the slots of the server to the respective user. Network optimization is a complex issue in the current network framework. One of the major issues is a balanced allocation of the server to avoid network traffic which slows down the overall performance. In the optimization framework, the algorithm searches the server which is nearer to the user. For optimization, ROA is implemented. In addition, there could be a chance that multiple users may be closer to a particular single server. In this condition, the algorithm has been devised to route the user to the next nearest server where the slots are free to use. In this way simultaneously, the algorithm searches optimal slots based on the distance as well it equally distributes the slots. Most of the time, it is inevitable that several slots in the server are less than the available user. In those circumstances, the algorithm implements the queuing theory concepts. In which users are allowed to stand in a queue where the slots are being allotted based on the first cum first serve bases. Fig. 4 shows a typical step in choosing a server and slot by JAVA action script simulation program, which is developed for simple cloud model multi-server and multi-user constraints. In this step, the server is running at its full capacity, so the user gets the message as shown in Fig. 4. Also, the user is kept on the top of the queue so that he could be allotted immediately when some other user finishes his process. We can use the same output in Hypertext-Markup Language (HTML) based optimization, even though based on my study, the time taken for the calculation to optimize the server slots will take time through this simulation concept.

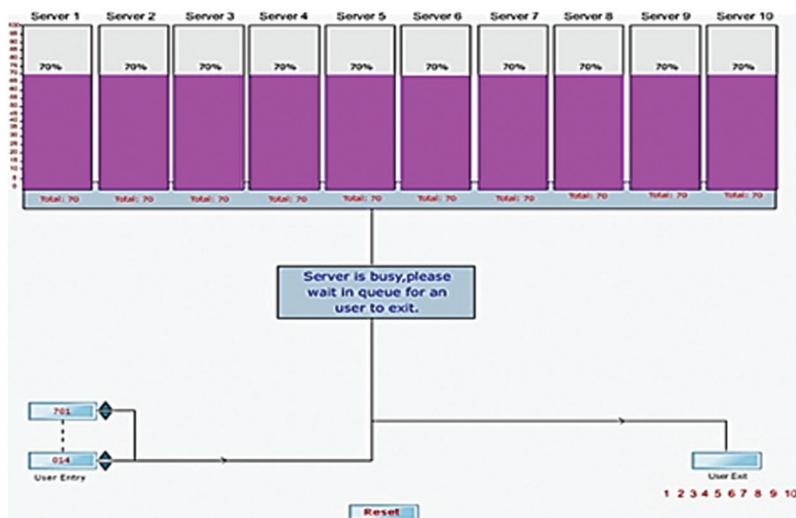


Figure 4: Schematic diagram of selection of server slot

4.1 Comparative Analysis

To demonstrate the proposed methodology's effectiveness, it was compared to other algorithms. In this process, Rider Optimization Algorithm (ROA) is utilized for the optimal server selection process. For comparison, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Firefly Algorithms (FA) are utilized. In this comparative analysis, 100 servers are utilized. The proposed methodology performance is analyzed in terms of memory usage, time and delay.

In Fig. 5, the projected procedure enactment is investigated in terms of execution time by altering number of user appeals. The minimum running time would be a good system. This proposed method was taken a minimum time of 1010 ms for 100 user requests. Here, the optimal server was selected based on

the user characteristics. Compared to PSO, GA and FF, the ROA method was better because the other three algorithms were falling in local optima and were only used for a small number of user requests. In this Fig. 5, the user request is increased; running time also gradually increases.

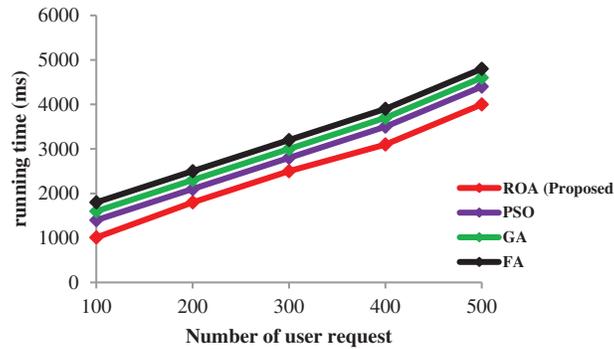


Figure 5: Server performance based on running time

In Fig. 6, the projected procedure enactment was investigated in terms of memory usage. While examining Fig. 6, the projected process acquired only a least delay time of the 20s, 40s, 60s, 80s and 100 s for user requests 100, 200, 300, 400 and 500 correspondingly. Similarly, PSO-based data transmission got the delay time of the 70 s, 140 s, 170 s, 220 s, and 300 s for user request 100, 200, 300, 400 and 500 respectively. As the outcome is clear that got a minimum time delay for different user request-based data transmission.

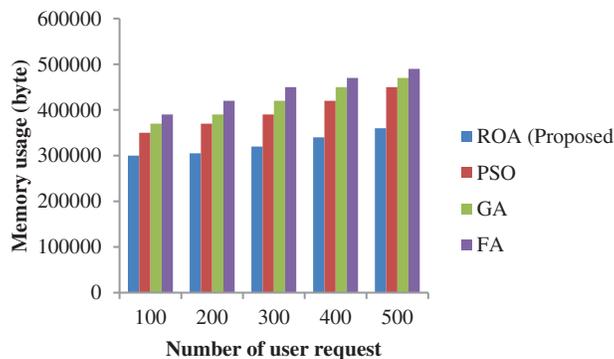


Figure 6: Server performance based on memory usage

It is clear that utilized only minimum memory compared to the existing approach and also in Fig. 7 the performance is analyzed in terms of delay time. This method is introduced due to network traffic. To avoid the traffic, server optimization is introduced.

The overall wordone based on the memory usage, delay and running time the proposed method provides efficient result by utilizing less running time, memory usage and delay in comparison with the conventional techniques. Fig. 8 shows the overall performance graphical representation.

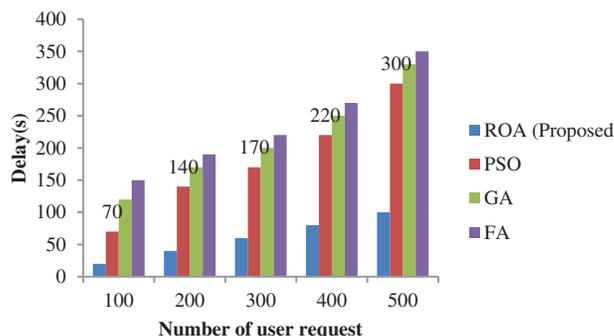


Figure 7: Server performance based on delay time

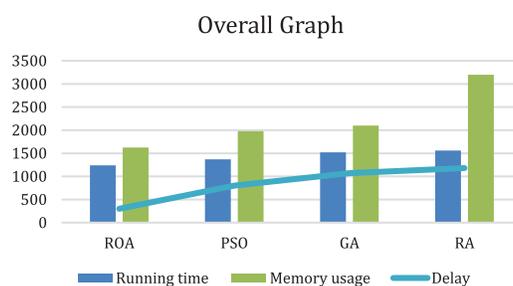


Figure 8: Overall work done representation graph

4.2 Experimental Results Based on Five Servers and Forty Users

In this section, the server is allocated to the user based on the distance and server usage and for optimal server selection, the ROA algorithm is utilized. Tab. 3 shows the server and slot allocation table. So that total slots of the whole setup are twenty-five, and server and slots allocation for 40 users was calculated. The first twenty-five users got the slots and servers based on the optimization concept, and the remaining 26th to 40th users got servers and slots based on queuing concept. The first program checked the lower utility server and then which is nearer to the user location, based on the optimization calculation user got a slot at the right server. Here distances of 26th to 40th can be any because the system understands the user's priority based on the time interval of user arrival to the system.

Table 3: Server and slot allocation table

User No.	Distance to Chennai Server	Distance to Bombay Server	Distance to Delhi Server	Distance to Guwahati Server	Distance to Roorkee Server	Server Usage Time (in s)	Server allotted for User	Slot allotted for User
1	741	474	587	378	306	12118.9227855207	5	1
2	129	267	521	368	501	22475.9886411207	1	1
3	194	748	8	658	700	7501.73789970716	3	1
4	298	169	633	266	608	37100.5005268765	2	1
5	56	490	692	338	364	22229.9753213311	4	1
6	513	454	578	723	568	25709.4876352725	5	2

(Continued)

Table 3 (continued)

User No.	Distance to Chennai Server	Distance to Bombay Server	Distance to Delhi Server	Distance to Guwahati Server	Distance to Roorkee Server	Server Usage Time (in s)	Server allotted for User	Slot allotted for User
7	302	291	33	38	313	36708.3419128680	1	2
8	737	107	284	730	729	10300.3880067398	3	2
9	302	20	529	143	741	42774.2871494359	2	2
10	466	316	547	501	648	61033.8780520177	4	2

Fig. 9 shows an output graph of MATLAB optimization and queuing program for allocation of server and slot to the user based on server utility and distance from user to the server. The X-axis denotes the users then the Y-axis denotes the server number. This algorithm takes care of gradual load sharing from server-side as mentioned earlier.

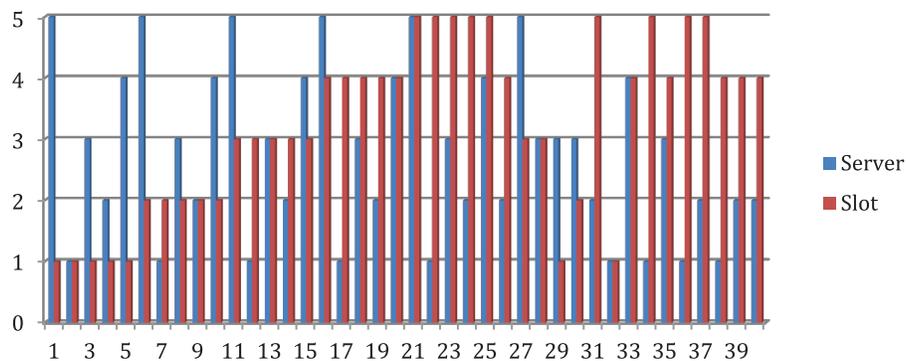


Figure 9: Output for server slot distribution

Fig. 10 shows an output graph for the time taken of each user in the system. This is to avoid a particular server being overloaded and a particular server being idle. This also takes care of new users to take maximum processing capacity server based on availability and comparison logic.

Tab. 3 shows the distance between each user to five servers, which are placed IIT Delhi, IIT Madras, IIT Bombay, IIT Guwahati, and IIT Roorkee. In this calculation, we have taken five servers and mirrors from five cities and each sever had five slots [29,30].

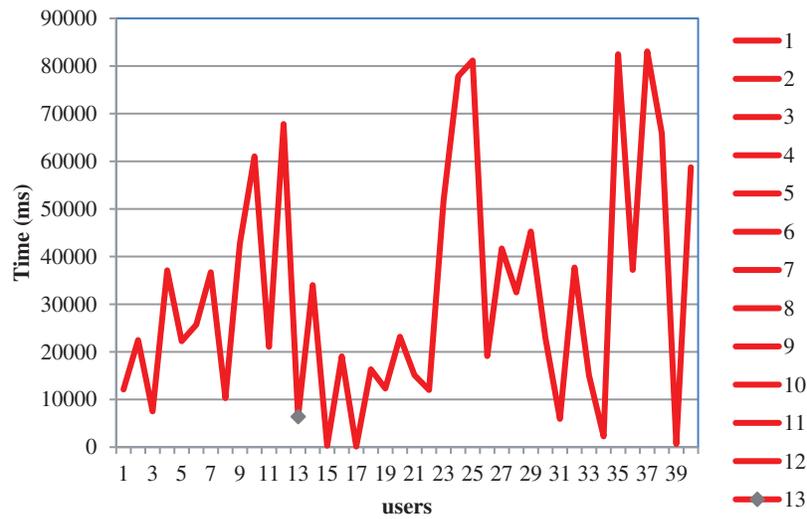


Figure 10: Time taken for each user in the system

5 Conclusion

The network optimization approach based on data transmission in the cloud has been outlined effectively in this article by utilizing ROA. The Rider Optimization Algorithm (ROA) is used for selecting the optimal server with the least delay and memory utilization. The bypass rider's goal is to reach the target by avoiding the leader's path. There are several applications of the rider optimization algorithm, notably document clustering, enhanced video super-resolution, webpage re-raking, and resource allocation, all of which are utilized to gain access to materials to facilitate e-learning. Based on various requests and metrics, we have evaluated the performance of this methodology. Compared to PSO, GA, and Firefly, the present algorithm utilizes minimum memory usage. According to this ROA, the server-side will gradually share the load, as suggested earlier, to prevent overloading and idleness on a particular server. Based on the availability and comparison logic, it ensures new users are assigned for the maximum processing capacity of the server.

Future Scope

This work can be further enhanced by inducing ensemble of ROA with other machine learning algorithm for further performance improvement.

Acknowledgement: The authors would like to express their gratitude to Anna University, as well as anonymous reviewers for their "insights."

Funding Statement: The authors received no specific funding for this study.

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

References

- [1] M. S. Krishnan, "NPTEL: A programme for free online and open engineering and science education," in *Proc. Int. Workshop on Technology for Education*, Bangalore, India, IEEE, pp. 1–5, 2009.
- [2] R. S. R. Pandian and K. S. Kasiviswanathan, "Effective use of cloud computing concepts in engineering colleges," in *Proc. IEEE Int. Conf. on Technology for Education*, Chennai, India, IEEE, pp. 233–236, 2011.

- [3] J. M. Warriem, "NPTEL local chapters: Facilitating mainstreaming of MOOCs in higher education," in *Proc. Int. Conf. on Technology for Education (T4E)*, Chennai, India, IEEE, pp. 89–92, 2018.
- [4] M. S. Krishnan, "NPTEL: A programme for free online and open engineering and science education," in *Proc. Int. Workshop on Technology for Education*, Bangalore, India, IEEE, pp. 1–5, 2009.
- [5] J. Ravi and H. J. Jani, "A critical study of NPTEL," in *Proc. IEEE Int. Conf. on Technology for Education*, Chennai, India, pp. 35–42, 2011.
- [6] M. S. Ananth, "National programme on technology enhanced learning (NPTEL): The vision and the mission," in *Proc. Int. Conf. on Technology for Education*, Chennai, India, IEEE, pp. 8, 2011.
- [7] K. K. Bhattacharjee, "SWOT analysis of NPTEL knowledge portal," in *Proc. 2014 IEEE Int. Conf. on Industrial Engineering and Engineering Management*, Selangor, Malaysia, pp. 1013–1017, 2014.
- [8] A. H. Reddy, "NPTEL: A best e-learning source for engineering students," *International Journal of Library and Information Studies*, vol. 6, no. 4, pp. 248–252, 2016.
- [9] R. Nitonde, "Impact of NPTEL on faculty and students in non-technical colleges," in *Proc. Int. Conf. on Technology for Education (T4E)*, Goa, India, IEEE, pp. 260–261, 2019.
- [10] R. S. R. Pandian, S. Thangalakshmi and S. Saravanan, "Virtual learning system: A conceptual framework of network optimization," in *Proc. of the Int. Conf. on Soft Computing for Problem Solving*, India, Springer, pp. 789–795, 2012.
- [11] J. Lee and L. Martin, "Investigating students perceptions of motivating factors of online class discussions," *International Review of Research in Open and Distributed Learning: IRRODL*, vol. 18, no. 5, pp. 148–172, 2017.
- [12] W. Sun, G. Z. Dai, X. R. Zhang, X. Z. He and X. Chen, "TBE-Net: A three-branch embedding network with part-aware ability and feature complementary learning for vehicle re-identification," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2021.
- [13] W. Sun, L. Dai, X. R. Zhang, P. S. Chang and X. Z. He, "RSOD: Real-time small object detection algorithm in UAV-based traffic monitoring," *Applied Intelligence*, vol. 92, no. 6, pp. 1–16, 2021.
- [14] M. A. H. Masud and X. Huang, "A novel approach for adopting cloud-based e-learning system," in *Proc. IEEE Int. Conf. on Computer and Information Science IEEE*, Shanghai, China, pp. 37–42, 2012.
- [15] M. A. Elmasry and M. H. Ibrahim, "Cloud computing for e-learning: A proposed model for higher education institutions in developing countries," *International Journal of Scientific & Technology Research*, vol. 10, no. 4, pp. 2277–8616, 2021.
- [16] S. Ssemugabi and R. De Villiers, "A comparative study of two usability evaluation methods using a web-based e-learning application," in *Proc. Ann. Research Conf. of the South African Institute of Computer Scientists and Information Technologists on IT research in Developing Countries*, New York, United States, pp. 132–142, 2007.
- [17] P. P. Ray, "Web based e-learning in India: The cumulative views of different aspects," *Indian Journal of Computer Science and Engineering*, vol. 1, no. 4, pp. 340–352, 2010.
- [18] E. Barbierato, L. Campanile, M. Griboaldo, M. Iacono, M. Mastroianni *et al.*, "Performance evaluation for the design of a hybrid cloud based distance synchronous and asynchronous learning architecture," *Simulation Modelling Practice and Theory*, vol. 109, no. 1, pp. 102303, 2021.
- [19] T. Kalushkov, D. Valcheva and G. Markova, "A model for pseudo-cloud hosted e-learning module for collaborative learning," in *Proc. 2018 2nd Int. Symp. on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Ankara, Turkey, pp. 1–5, 2018.
- [20] D. N. Vanjulavalli, S. Arumugam and D. A. Kovalan, "An effective tool for cloud based e-learning architecture," *International Journal of Computer Science and Information Technologies*, vol. 6, no. 4, pp. 3922–3924, 2015.
- [21] A. Alameen and A. Gupta, "Fitness rate-based rider optimization enabled for optimal task scheduling in cloud," *Information Security Journal: A Global Perspective*, vol. 29, no. 6, pp. 310–326, 2020.
- [22] M. Yarlagadda, K. G. Rao and A. Srikrishna, "Frequent itemset-based feature selection and rider moth search algorithm for document clustering," *Journal of King Saud University-Computer and Information Sciences*, vol. 1, no. 11, pp. 23–31, 2019.

- [23] S. Augustine and J. P. Ananth, "A modified rider optimization algorithm for multihop routing in WSN," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 33, no. 6, pp. 2764, 2020.
- [24] K. N. Vhatkar and G. P. Bhole, "Improved rider optimization for optimal container resource allocation in cloud with security assurance," *International Journal of Pervasive Computing and Communications*, vol. 16, no. 3, pp. 235–258, 2020.
- [25] M. Adhikari and S. N. Srirama, "Multi-objective accelerated particle swarm optimization with a container-based scheduling for Internet-of-Things in cloud environment," *Journal of Network and Computer Applications*, vol. 137, no. 4, pp. 35–61, 2019.
- [26] J. López, N. Kushik and D. Zeghlache, "Virtual machine placement quality estimation in cloud infrastructures using integer linear programming," *Software Quality Journal*, vol. 27, no. 2, pp. 731–755, 2019.
- [27] E. J. P. Stamm, M. Zafonte and S. M. Palenque, "The effects of instructor participation and class size on student participation in an online class discussion forum," *British Journal of Educational Technology*, vol. 48, no. 6, pp. 1250–1259, 2017.
- [28] R. K. Poluru and L. K. Ramasamy, "Optimal cluster head selection using modified rider assisted clustering for IoT," *IET Communications*, vol. 14, no. 13, pp. 2189–2201, 2020.
- [29] E. Kim, J. Kim and C. Lee, "Efficient neighbor selection through connection switching for P2P live streaming," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 4, pp. 1413–1423, 2019.
- [30] P. Hao, L. Hu, J. Jiang, X. Che, T. Li *et al.*, "Framework for replica placement over cooperative edge networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 8, pp. 3011–3021, 2019.