EAI Endorsed Transactions
on Internet of Things

Research Article **EAI.EU**

Cloud Computing: Optimization using Particle Swarm Optimization to Improve Performance of Cloud

Nidhi[1,*], Malti Nagle[2] and Vishal Nagar[3]

[1, 2, 3]Department of Computer Science & Engineering, Pranveer Singh Institute of Technology, Kanpur, Uttar Pradesh, India

Abstract

INTRODUCTION: In the contemporary world cloud computing is acknowledged as advanced technology to manage and store huge amount of data over the network. To handle the network traffic and effective task scheduling some efficient load balancing algorithm should be implemented. This can reduce the network traffic and overcome the problem of limited bandwidth. The various research articles represents ample amount of optimization techniques to overcome the transfer of data with limited bandwidth. Among all, few solutions has been chosen for current research article such as – optimization of load distribution of various resources provided by cloud.
OBJECTIVES: In this paper, Comparative analysis of various task scheduling algorithms such as (FCFS, SJF, Round Robin & PSO) have been proposed in current research article to accumulate the outcome and evaluate the overall performance of cloud at different number of processing elements (pesNumber) .
METHODS: Overall performance of task scheduling is significantly enhanced by PSO Algorithm implemented on cloud in comparison of FCFS, SJF and Round Robin. Outcomes of optimization technique has been implemented and tested over the CloudSim simulator.
RESULTS: The comparative analysis conducted based on scalability for increasing the number of processing elements over the cloud. The major insight of proposed algorithm has shows that results are still better when number of VMs is increased and it successfully minimizes waiting time and turnaround time and completion time by 43% which is significantly high than outcomes of existing research articles.
CONCLUSION: To optimize the task scheduling in cloud computing, comparative analysis of various task scheduling algorithms has been proposed, including Particle Swarm Optimization algorithm.

Keywords: Fog Computing, FCFS, SJF, Task Scheduling, Cloud Computing, Round Robin, PSO, CloudSim

Received on 09 September 2023, accepted on 30 November 2023, published on 12 December 2023

doi: 10.4108/eetiot.4577

1. Introduction

In recent days, IoT devices are increasing exponentially and become part of daily lives, so as it is essential to manage and store data at cloud. Data is transmitted through bandwidth to the cloud, and since the distance between cloud and devices are large, it causes delay in transmission [25]. Novel approach to manage the data at middle layer i.e. fog layer is appreciated but also introduced new challenges such as network traffic and load balancing in polynomial time [2]. IoT can be assumed as network connection of large number of sensors and actuators that collect the data and forward to cloud [3]. Latency required to store and forward the sensors data is very low. Various researches present the different algorithms to improve the performance of cloud computing addressing edge

*Corresponding author. Email: nagle.malti083@gmail.com

computing. According to the literature survey many challenges have been noticed in the area of edge computing, such as: latency delay, response time, computational cost, support for mobility and geological transformation of data over cloud [4]. These challenges can be addressed by efficient cloud computing [5].

The cloud receive the data from various fog devices, this increases the latency and task scheduling is also a key issue in cloud computing. The management and storage of real-time fog device's data is tedious task for cloud. The architecture shown in figure1 represents overall structure of edge and cloud computing. Previously cloud acknowledged and resource provider but after novel approach of fog layer, new responsibility of managing and storage of fog data at cloud. This also introduced the new challenges of communication technology [2][8][9][10]. Variation like latency delay and other computational parameters address an important issue in the edge computing [11].

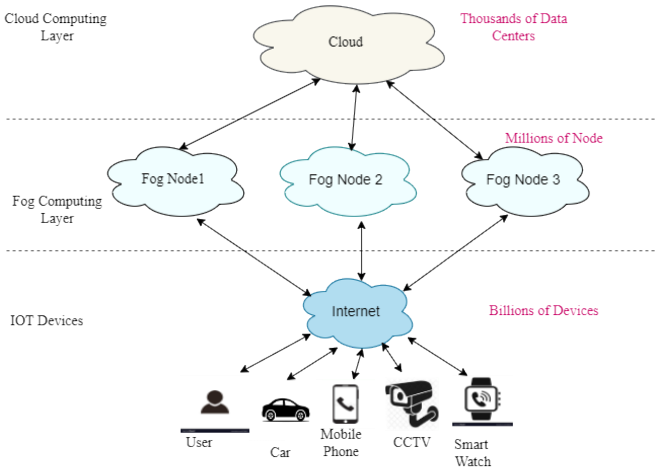

Figure 1. Cloud Computing Architecture

There are many challenges in a cloud computing environment. Biggest challenge is load balancing using task scheduling. Tasks are generally classified into two parts, dependent tasks and independent tasks. When planning tasks in the fog, the task category plays a crucial role. Depending on the suitable task to schedule, different scheduling algorithms can be used. The scheduling algorithms can take advantage of better execution efficiency and maintain system load balancing. The suitable scheduling algorithm may be useful for load balancing. While in- preemptive scheduling algorithm, task execution could be slowed down as per requirement. There are too many factors to evaluate the efficiency of scheduling algorithms, including both waiting time and processing time [5]. Turnaround time refers to the total time for completing the task, while waiting time is the total time a task has been waiting in the ready queue.

According to survey researchers found that if numbers of users are increased in cloud computing, the tasks to be scheduled in the cloud also increased. Therefore, in these system, there is a requirement of better scheduling algorithms. Different preemptive and non-preemptive task scheduling algorithms are proposed by many researchers, standard scheduling algorithms like Shortest Job First (SJF), Round Robin (RR), or hybrids thereof like FCFS (First-Come First-Serve) and RR and SRTF (Shortest-Remaining-Time-First) and RR. So, in this paper, outcome produced based on FCFS, SJF, Round Robin (RR) and PSO (Particle Swarm Optimization) algorithms by increasing the number of processing elements (pesNumber) to evaluate their Average Waiting Time (AWT) and Average Turnaround Time (ATAT).

2. Literature Review

The tasks to be planned in cloud have become more and more frequent as the number of users of cloud computing systems increases. In this context, more efficient algorithms for scheduling tasks on these systems need to be developed. In this section, researchers are focused on many load balancing and task scheduling algorithms.

The author in [1] proposed Multi-Objective Particle Swarm Optimization (MOPSO) algorithm for task scheduling in cloud computing. The objective of the proposed algorithm is to minimize the waiting time and maximize the throughput, when tasks are assigning to virtual machines.

Various existing scheduling algorithms (such as, FCFS, PSO, Genetic Algorithm, one optimization Algorithm), Simulated Annealing Algorithm) are presented in Kabirzadeh et al. [12], [13] and propose a hyper-heuristic algorithm for the evaluation of fog computing. These algorithms are used in centralized cloud architecture.

In article [6], In terms of time delays and response times, the comparison between fog and cloud computing. The performance of fog computing is superior to cloud computing, as long as the delay and response times are reduced.

There are few studies focused on optimization of resources using scheduling techniques. Jalalietal.[14] suggested two models for shared and non-shared network devices for reducing the power consumption, flow-based and time-based, respectively. The author in [14] used the Q-learning algorithm for allocating the task to the virtual machine in cloud computing. FIFO algorithm, Greedy, Random and Mix algorithm are compared with Q learning algorithm. Their model has three parts; tasks transmission, assigning tasks and executing tasks.

In cloud computing, the most significant tasks are resource management and task scheduling. In classical scheduling algorithms, minimum utilization of resources and maximize response. There are six machine learning scheduling algorithms (such as, FCFS, priority scheduling, priority-based consolidation, aggressive migration supported backfilling, and conservative migration supported backfilling) are used. The choice of these algorithms based on the environment and task by means of machine learning classification [15].

There are two reinforcement learning algorithms for resource scheduling. These algorithms are; Offline Resource Scheduling DeepRM_off and Online Resource Scheduling DeepRM2. So that, DeepRM and the heuristic algorithms are compared with these scheduling algorithms. For CPU and memory there are two resources are considered and input taken as image for training process [16].

The author in [17] proposed three task scheduling algorithms and compared them. These algorithms are PSO, modified PSO and Genetic Algorithm. Classical PSO is merged with SJFP to generate the initial population to reduce the makespan, named as modified PSO Algorithm. The comparison of these algorithms demonstrates that the modified PSO exceeds the other two algorithms.

The author in [18] proposed a new algorithm to schedule the tasks or jobs in Big Data clusters. The fact that this proposed algorithm is essentially focused on resource utilization and type of scheduled task makes it unique. Homogeneous clusters are used for experiment purpose. On the basis of the type of the job and resource load for data node, the proposed algorithm will assign the tasks to that node.

K-means clustering is used to cluster the virtual machines and tasks [19]. The virtual environment is categorized based on the application availability in each machine. There are four parameters for task selection are; length of task, deadline, cost and priority of user.

The author in [20] proposed a method which is critical task, to increase the performance of the task scheduling in cloud computing. A frame work has been proposed for this problem. The author suggested using a machine learning algorithm to determine which task scheduling algorithm is best for the specific task.

The fundamental idea in [21] is allocating tasks to various fog nodes. The performance of proposed algorithm is compared with PSO and Genetic algorithms. The given algorithm divides the whole task into two parts; food source foraging behavior and reproductive behavior. BLA is implemented in C++ language. The proposed algorithm exceeds as to CPU execution time, allocation of memory and hence, the cost functions. The disadvantage of proposed algorithm was it does not provide the result for dynamic job scheduling.

The author in [22] proposed a test and choice algorithm to choose the best scheduling algorithm. Training phase and Testing phase are two parts of hyper heuristic algorithm. The main motive of proposed wok is to determine the best algorithm for work flow scheduling.

The author in [23] proposed an Ant Colony scheduling algorithm. The tasks and priority of the tasks are based on two criteria, minimal cost and minimal end-time service. To choose the optimal virtual machine to perform the task, Ant colony algorithm is used. The key feature of fog computing system is to achieve extremely low task latency and multi-resource fairness.

Ghosh et al. (2023) embarked on a comprehensive study to assess water quality through predictive machine learning. Their research underscored the potential of machine learning models in effectively assessing and classifying water quality. The dataset used for this purpose included parameters like pH, dissolved oxygen, BOD, and TDS. Among the various models they employed, the Random Forest model emerged as the most accurate, achieving a commendable accuracy rate of 78.96%. In contrast, the SVM model lagged behind, registering the lowest accuracy of 68.29%[26].

Alenezi et al. (2021) developed a novel Convolutional Neural Network (CNN) integrated with a block-greedy algorithm to enhance underwater image dehazing. The method addresses color channel attenuation and optimizes local and global pixel values. By employing a unique Markov random field, the approach refines image edges. Performance evaluations, using metrics like UCIQE and UIQM, demonstrated the superiority of this method over

existing techniques, resulting in sharper, clearer, and more colorful underwater images [27].
Sharma et al. (2020) presented a comprehensive study on the impact of COVID-19 on global financial indicators, emphasizing its swift and significant disruption. The research highlighted the massive economic downturn, with global markets losing over US $6 trillion in a week in February 2020. Their multivariate analysis provided insights into the influence of containment policies on various financial metrics. The study underscores the profound effects of the pandemic on economic activities and the potential of using advanced algorithms for detection and analysis [28].

3. Problem Statement

It has been concluded from the literature review that cloud encounters different novel challenges after enhancement of fog layer. The heterogeneous real-time data increases latency delay, network traffic and also task scheduling. Among of all challenges some of the key challenges have been presented in current paper. Task scheduling is one of the main key issues for cloud computing when it has to deal fog devices' large data. Also transfer of fog data over the cloud has become more complex due to limited bandwidth. To analysis the overall performance of cloud it is essential to identify such key issues and rectify for better performance of cloud. In current article some of the key challenges have been notified with the help of parameters (such as-load balancing at cloud end; decrease the completion time and resource scheduling). The scalability of cloud is also need to be tested to find the maximum number of processing elements that can be accommodated in limited network capacity. This could help to identify the maximum limit of resources, which can be assigned to cloud.

4. Proposed Methodology

The proposed algorithms are implemented on CloudSim simulator (which is an open-source cloud computing library) for Java on a computer with Intel Core i3-3110M CPU @ 2.40 GHz and 2 GB RAM. CloudSim was chosen to test the algorithms as it allows simulating multiple cloud components such as data center, host, broker, virtual machine and cloudlet.
Cloud computing setup required four basic components. This environment is installed to analysis the effectiveness of task scheduling algorithms. The components are as follows; Data Center, Data Center Broker, Cloudlet and Virtual Machines. The responsibility of data center is to provide the hardware-level services to cloud users. Datacenter Broker creates and destroys VMs as per the requirements of the tasks and provide virtual environment to the user. The VMs processes tasks according to the policy can be provided by the cloudlet scheduler. Cloudlet is the running task on the virtual machine.

4.1. Experimental Setup (CloudSim):

CloudSim 3.0 is the newest version of cloudsim with the fixes all the bugs. It was released on Jan11, 2012 [24].

Table 1. Host Parameters

Number of Host	1
Processing Speed (MIPS)	1000
Memory(MB)	4096
Storage	1,000,000
Bandwidth	60,000

The experimental setup for cloud consists of one cloud hosts with processing speed 1000MIPS each.10000 MB is the size of virtual machine with 2048MB memory and bandwidth as 3000.Cloudsim system architecture is “x86” with OS “Linux”. Number of tasks (cloudlet) is 40 for experiment. 05 datacenters and 20 VMs for each datacenter are considered.

Table 2. VM Parameters

VM's size(MB)	10,000
Memory(MB)	2048
Bandwidth	3000

Table 3. CloudSim System Parameters

SA (System Arch.)	x86
OS	Linux
Time Zone (Resource)	10.0
Processing Cost	3.0
Memory	0.05
Storage Cost	0.1
Bandwidth Cost	0.1

Table 4. Cloudlet Parameters

No. of Cloudlets	40
Cloudlet Length	1000
Long File Size	300
Long Output Size	300
pesNumber	From1to32

Table 5. Datacenter Parameters

Total no. of Datacenters	5
Total no. of Virtual Machine	20

The parameter for QoS is latency, robustness, time, power consumption, bandwidth and cost. The algorithms are used in this experiment were consider based on the work performed in [18]. The selected algorithms were successfully applied to the cloud computing linear programming problem. The parameters are;

- **Bandwidth:** Total amount of data, which is transferred from and to network over given period of time.
- **Latency:** Delay in data transfer to and from cloud server to client.
- **Makespan:** Completion time to execute the algorithm.
- **W T:** The total amount of time spent by the process in the ready queue waiting for CPU.
- **Burst Time (BT):** The total time required by the process for its complete execution on CPU. Hence,

Waiting Time =Turn Around Time (TAT)–Burst Time (BT)

- **Turnaround Time:** Average time elapsed from the submission of a process to its completion. Hence,

Turn Around Time =Completion Time–Arrival Time

5. Experimental Results

In this research article distinct task scheduling algorithms (such as FCFS, SJF, RR and PSO) have been implemented over cloud. Task Scheduling Algorithms are;

- **FCFS (First Come First Serve):** FCFS is the first come first serve scheduling algorithm. It executes the process in the order of their arrival.
- **SJF (Shortest Job First):** SJF preemptive task scheduling it execute processes based on shortest time to complete the execution of respective processes and queuing them for execution in an ascending order. Therefore, this particular algorithm does the sorting first. SJF algorithm has contribute in reducing completion time and TAT in comparison of all scheduling algorithms.
- **Round Robin (RR):** Round Robin is preemptive scheduling algorithm, therefore it execute all processes by allocating fix time slice to all processes in rotation till all processes finish their execution. Each task is given equal time of accessing resources. The time allocated to complete the task or job is called quantum.
- **PSO (Particle Swarm Optimization):** PSO is a meta-heuristic algorithm for scheduling the allocation of resources between the resources, in order to increase the utilization of resources at cloud. The key feature of PSO is random initialization of particles (processing element) over the search space.

Comparative analysis has been accumulated and presented in table (6, 7, 8) given. Table6 shows the, Average Waiting Time of each algorithm (FCFS, SJF and Round Robin) along with different number of processing elements (pesNumber) given in the table.Table7 shows the Average Turnaround Time of each algorithm (FCFS, SJF and Round Robin) along with different number of processing elements (pesNumber). In this experiment the performance of the Round Robin (RR) is increased after pesNumber 32, i.e.; the performance of RR increased when pesNumber is 64. Table 8 shows the Makespan value (Completion time) of each algorithm (FCFS, SJF, Round Robin and PSO) along with different number of processing elements (pesNumber).

Table 6. Average Waiting Time of each algorithm with different pesNumber

pesNumber	FCFS	SJF	Round Robin
1	33.9552	32.2507	31.9833
4	37.3962	33.5462	34.728
8	38.6942	34.1487	35.7815
16	36.7815	33.9707	32.6725
32	37.0377	30.3572	32.3843

Table 7. Average Turnaround Time of each algorithm with different pesNumber

pesNumber	FCFS	SJF	RoundRobin
1	35.8269	34.2623	33.9675
4	39.3710	35.6305	36.8277
8	40.7914	36.2443	37.9415
16	38.6593	36.0355	34.6972
32	39.0183	32.2699	34.3910

Table 8. Overall Computational Cost of cloud using makespan value

pesNumber	FCFS	SJF	Round Robin	PSO
1	6110.253	4437.714	4575.028	3237.94
4	6523.459	4450.064	4279.171	2619.431
8	6310.214	4574.448	4699.436	3253.539
16	5780.14	5077.321	5142.504	3079.626
32	5067.721	4860.412	4819.729	2859.205

The graphical representation of Average waiting time over different pesNumber is shown in figure2.

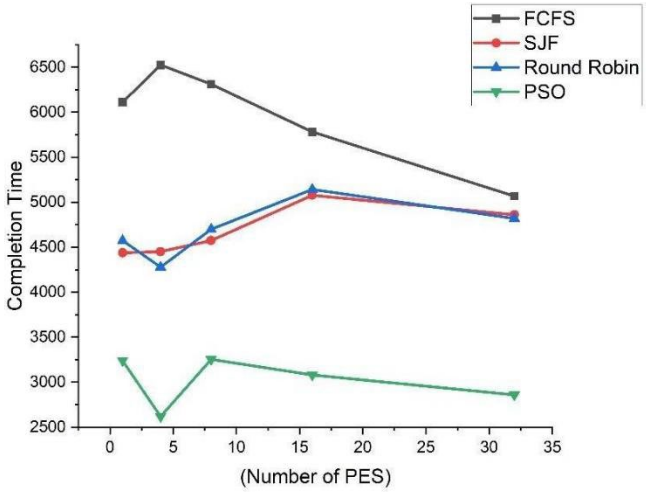

Figure 2. Average Waiting Time over different number of processing elements (pesNumber)

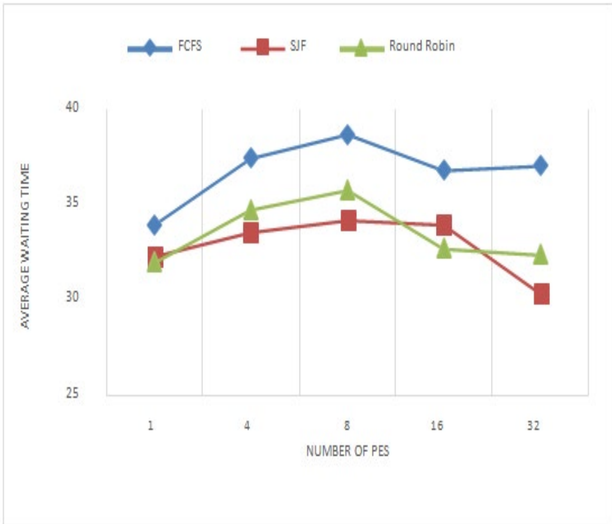

Figure 3. Average Turnaround Time over different number of processing elements (pesNumber)

The graphical representation of Average Turnaround Time over different number of processing elements (pesNumber) is shown in figure 3.

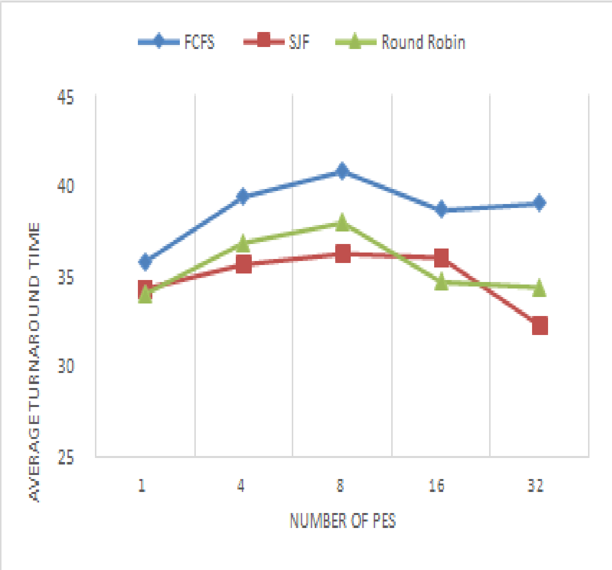

Figure 4. Average Completion Time over different number of processing elements (pesNumber)

The result shown in figure 2 represents the average waiting time of task scheduled at cloud which has been evaluated for different number of VMs and processing elements. Insight of figure 2 seeks attention to the scalability. It has been observed that if number of processing element increases than after 16 Pes average waiting time start increasing for RR. The similar outcomes explored for average turnaround time using round robin scheduling process. Figure 3 shows the graphical representation of average turnaround time of different algorithms. To overcome this scalability issue PSO has been analyzed and tested for different pes ranges from 1 to 32. Completion time is decreases when number of pes are increases till 32. In figure 4 shows the result of completion time.

5. Conclusion and Future Work

The Task Scheduling has a big impact on how well cloud computing performs. The best way to allocate jobs among resources to maximize the QoS metrics is one of the fundamental challenges in task scheduling. This study compares the First Come First Serve (FCFS), Shortest Job First (SJF), and Round Robin (RR) and Particle Swarm Optimization task scheduling algorithms, which are the most widely used in cloud computing. The CloudSim simulator toolbox has been used and comparison study was carried out to evaluate the Average Wait Time (AWT) and Average Turnaround Time (ATAT) for various processing element counts (pesNumber). The Analysis of cloud has been conducted on the basis of Makespan value and overall computational cost has been evaluated. The analysis states the improvement of 43% in comparison of existing proposed work in [1] which is significantly high. The further enhancement of current research is to achieve the effective outcomes at edge computing using suitable and best load balancing algorithm. The major insight of proposed algorithm has shows that results are still better when number of VMs is increased and it successfully minimizes waiting time and turnaround time.

References

[1] Entisar S. Alkayal, Maysoon F. Abulkhair, Nicholas R. Jennings, Efficient Task Scheduling Multi-Objective Particle Swarm Optimization in Cloud Computing, IEEE 41st Conference on Local Computer Networks Workshops, 2016.

[2] Z. Hao, E. Novak, S. Yi, Q. Li, Challenges and software architecture for fog computing, IEEE Internet Compute.21, 2017, 44–53.

[3] S. Sharma, H. Saini, A novel four-tier architecture for delay aware scheduling and load balancing in fog environment, Sustain.Compute.Inform.Syst.24,2019, 100355.

[4] S. Bitam, S. Zeadally, A.Mellouk, Fog computing job scheduling optimization based on bees swarm, Enterpr. Inform. Syst.12, 2018, 373–397.

[5] D. Tychalas, H. Karatza, A scheduling algorithm for a fog computing system with bag-of-tasks jobs: simulation and performance evaluation, Simul.Modell.Pract.Theory98, 2020, 101982.

[6] L. H. Kazem, Efficient resource allocation for time-sensitive IoT applications in cloud and fog environments, Int. J. Recent Technol. Eng. 8, 2019, 2356–2363.

[7] San Jose, Cisco Knowledge Networking, Cisco Global Cloud Index: Forecast and Methodology, 2015- 2020, white paper, Cisco Public, 2016.

[8] P. Hu, S. Dhelim, H. Ning, T. Qiu, Survey on fog computing: architecture, key technologies, applications and open issues, J. Netw. Compute. Appl. 98, 2017, 27–42.

[9] R. K. Naha, S. Garg, D. Georgakopoulos, P. P. Jayaraman, L. Gao, Y. Xiang et al., Fog computing: survey of trends, architectures, requirements, and research directions, 2019, 47980–48009.

[10] Y. Liu, J. E. Fieldsend, G. Min, A frame work of fog computing: architecture, challenges, and optimization, 2019, 25445–25454.

[11] H. Rafique, M. A. Shah, S. U. Islam, T. Maqsood, S. Khan, C. Maple, A novel bio-inspired hybrid algorithm (NBIHA) for efficient resource management in fog computing, 2019, 115760–115773.

[12] Rahbari D., Nickray M., Scheduling of Fog Networks with Optimized Knapsack by Symbiotic Organisms Search, Conf. Open Innov. Assoc. Fruct, 2017.

[13] Kabirzadeh S., Rahbari D., Nickray M., A hyper heuristic algorithm for scheduling of fog networks, Conf. Open Innov. Assoc. Fruct, 2018, pp.148–155.

[14] Mostafavi S, Ahmadi F, Sarram M., Reinforcement-Learning-Based Foresighted Task Scheduling in Cloud Computing, 2018.

[15] Pathak GR., Task Scheduling in the Cloud Using Machine Learning Classification, 2015.

[16] Yufei Ye,Jin Wang, Lingxiao, Wenxia Guo, Wenqian Huang, Wenhong Tian, A new approach for resource scheduling with deep reinforcement learning, 2018.

[17] Abdi S, Motamedi S, Sharifian S., Task scheduling using modified PSO algorithm in cloud computing environment, Int. Conf. Mach. Learn. Electr. Mech. Eng., 2014, pp.37-41.
[18] Hadjar K, Jedidi A., A new approach for scheduling tasks and/or jobs in big data cluster, 4th MEC International Conference on Big Data and Smart City (ICBDSC), 2019, pp.1-4.DOI:10.1109/ ICBDSC.2019.8645613.
[19] Sharma V, Bala M., A credits based scheduling algorithm with K-means clustering, First International Conference on Secure Cyber Computing and Communication (ICSCCC), 2018, pp.82-86. DOI:10.1109/ICSCCC.2018.8703201.
[20] Shetty C, Sarojadevi H., Framework for task scheduling in cloud using machine learning techniques, 2020 Fourth International Conference on Inventive Systems and Control (ICISC), 2020, pp. 727-731. DOI: 10.1109/ ICISC47916.2020.9171141.
[21] Bitam S, Zeadally S, Mellouk A., Fog computing job scheduling optimization based on Bees Swarm. Entrepreneurship, 2017, 12:1-25. DOI:10.1080/17517575.2017.1304579.
[22] Kabirzadeh S, Rahbari D, Nickray M., A hyper heuristic algorithm for scheduling of fog networks, 21st Conference of Open Innovations Association (FRUCT), 2017, pp.148-155.DOI:10.23919/FRUCT.2017.8250177.
[23] Ghaffari E., Providing a new scheduling method in fog network using the ant colony algorithm, 2019.
[24] R. Buyya, R. N. Calheiros, A. Beloglazov, and S.Garg. GRIDS Lab, cloudsim 3.0 package download, updates from version 2.0 to version 3.0, Jan11, 2012.
[25] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the internet of things, Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing (MCC), 2012, pp.13–16.
[26] Ghosh, H., Tusher, M.A., Rahat, I.S., Khasim, S., Mohanty, S.N. (2023). Water Quality Assessment Through Predictive Machine Learning. In: Intelligent Computing and Networking. IC-ICN 2023. Lecture Notes in Networks and Systems, vol 699. Springer, Singapore. https://doi.org/10.1007/978-981-99-3177-4_6
[27] Alenezi, F.; Armghan, A.; Mohanty, S.N.; Jhaveri, R.H.; Tiwari, P. Block-Greedy and CNN Based Underwater Image Dehazing for Novel Depth Estimation and Optimal Ambient Light. Water 2021, 13, 3470. https://doi.org/10.3390/w13233470
[28] G. P. Rout and S. N. Mohanty, "A Hybrid Approach for Network Intrusion Detection," 2015 Fifth International Conference on Communication Systems and Network Technologies, Gwalior, India, 2015, pp. 614-617, doi: 10.1109/CSNT.2015.76.