International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Priority-Based Virtual Machine Selection Algorithm in Cloud Computing

Manoj Kumar, Suman

Abstract*: *Cloud computing technology has gained the attention of researchers in recent years. Almost every application is using cloud computing in one way or another. Virtualization allows running many virtual machines on a single physical computer by sharing its resources. Users can store their data on datacenter and run their applications from anywhere using the internet and pay as per service level agreement documents accordingly. It leads to an increase in demand for cloud services and may decrease the quality of service. This paper presents a priority-based selection of virtual machines by cloud service provider. The virtual machines in the cloud datacenter are configured as Amazon EC2 and algorithm is simulated in cloud-sim simulator. The results justify that proposed priority-based virtual machine algorithm shortens the makespan, by 11.43 % and 5.81 %, average waiting time by 28.80 % and 24.50%, and cost of using the virtual machine by 21.24% and 11.54% as compared to FCFS and ACO respectively, hence improving quality of service.

Keywords* : *Cloud, Cost, Makespan, Quality of Service

I. INTRODUCTION

Information and Communication Technology (ICT) has played a vital role in today's modern life to make it as simple as possible. The era of Fifth Generation Mobile Technology (5G) has already started in some counties, giving connectivity to IoT devices, smart devices, mobile cloud, giving upswing to more users [1]. Cloud computing consists of a large number of virtualized machines that run on various host or Physical Machines. The services of cloud computing are Infrastructure as a Service, Platform as a Service and Software as a Service [2][3]. Users can avail these services using a variety of devices like smartphones, PCs, tablets, etc. over the internet. Cloud computing is a very proficient paradigm due to its reliability, availability, on-demand, easy to maintain, pay per -use, and scalability [4][5]. Quality of Service in cloud computing as an essential issue on which many researchers are working. Various algorithms on task scheduling, load balancing, resource management, VM provisioning has been proposed to provide a better quality of service to users by Cloud service providers [6].

The growth of the internet and demand for resources is increasing at high speed. Every user wants to get a better quality of service from its provider. Growing demands for resources and services in the cloud degrade the quality of service. Quality of service depends on various scheduling strategies like task scheduling, virtual machine allocation and placement, provisioning, task selection, and scheduling. Virtual machine scheduling is an essential part of cloud computing that affects many parameters of quality of service in cloud computing. It is a three-step process: Finding the appropriate physical machines, selection of virtual machines from VM pool, and submission of the tasks to virtual machines [7]. Scheduling in cloud computing is a critical problem, and researchers have proposed several techniques to solve this [8] and tried to optimize makespan, waiting time, cost of execution, resource utilization and many more. Scheduling problems become more complicated for heterogeneous data centers where resources have different parameters like capacity, processing power, CPU speed, RAM, number of CPU cores, and size of the disk. So, a scheduling algorithm is required that can efficiently and effectively work on heterogeneous resources and can optimize various parameters of quality of service [9].

In this paper, a priority-based scheduling model has been presented that will select the best virtual machine from the virtual machines pool for the task. In this model, tasks submitted to cloud broker are marked with a priority tag. Based on this tag, the algorithm will search best available virtual machine and finally, the task is bound to a particular virtual machine. Rest of the paper is organized in the following manner: Section 2 presents related work done by researchers. Section 3 presents a problem definition. Section 4 presents the proposed priority-based algorithm to select the best available virtual machine on which the requested task is to be bound. Section 5 presents the results of simulations and a comparison of proposed technique with some existing one. Finally, the paper is concluded in section 6 with some valuable suggestion for future research directions.

II. RELATED WORK

Many researchers and scientist have proposed various algorithms and optimization techniques to improve the quality of service. Ding et al. [10] had proposed a virtual scheduling technique to save energy of the cloud data center with deadline constraints so that all jobs could be executed within the deadline. Haque et al. [11]

Manuscript published on 30 September 2019
* Correspondence Author
Manoj Kumar*, CSE Department, DCR University of Science and Technology, Murthal, Sonepat, India. Email: manojsarasyiya@gmail.com
Prof. Suman, CSE Department, DCR University of Science and Technology, Murthal, Sonepat, India. Email: suman.cse@dcrustm.org

has proposed a priority-based scheduling technique based on multiple criteria decision-making model. The author has defined priorities at three levels namely: scheduling level, resource level and job level in order to complete the task on time. The algorithm lacks in calculating makespan of tasks submitted and cost of execution.

Mokhtar A. Alworafi et al. [12] proposed a cost priority approach to minimize the cost of bandwidth, memory, and storage while taking budget as a constraint for each task submitted. The author considered three clusters for small, medium and large VM and then sending them according to the priority of the task. The author has not described the traditional approach and makespan of each task. The work lacked in the processing of task when all clusters were busy in executing their assigned tasks. Meriam & Mediatron et al. [13] had proposed a multiple QoS based scheduling in cloud computing to ensure less execution time, less delay and low cost. The author considers first 40 % task as high, next 40 % as medium and rests 20 % as low however in the real cloud computing environment, it is not applicable and did not act with the real-time workload. Balagoni & Rajeswara Rao et al. [14] proposed a scheduling algorithm to minimize the cost of execution with minimization of SLA violation. M Linda & Ananthanarayana et al. [15] proposed an Ant Colony Optimization based scheduling algorithm to minimize energy consumption by 22%. Stavrinides & Karatza, et al. [16] proposed a scheduling technique to finish all workflows within deadline limit at high-quality results. They managed to minimize the execution time for each workflow applications and cost charged to the user.

III. PROBLEM DEFINITION

Suppose there is N number of users in the cloud environment and each user has sent the request to access cloud service at any instance of time. The request of the user (task) is considered with one additional tag called priority. The high priority task should be completed first cost-effective task should be completed with minimum cost of execution. Suppose each task T_i has a priority $P \in \{1,2,3\}$. Suppose ET_j is the execution time T_i^{th} task and WT_j is waiting time for T_i^{th} task, then makespan of all task can be computed as follows:

Waiting Time of Task T_i, WT_i= Start Time T_i - Submission Time T_i …………………... (1)

Execution Time of Task T_i, ET_i= Finish Time T_i - Start Time T_i…………………………………………...……….…… (2)

Makespan $=\sum_{j=0}^{n} ETj + \sum_{j=0}^{n} WT$……....………….… (3)

Cost of execution of each task can be computed by finding its execution time and price of virtual machine which has executed the task. So cost of using VM can be formulated as:

Cost of VM = $\sum_{j=0}^{n} VMkp * \sum_{j=0}^{n} ETi$ ……….....… (4)

where VM_{kp} is price of using VM_k in seconds and ET_i is the execution time of task T_i. So the execution time of all task having priority P=1 should be minimum and execution cost of task having priority P=3 should be minimum. So mathematically we can formulate the following equations as objectives to minimize:

$min \sum_{i=0}^{j} Makespan$ …………………………….…. (5)

$min \sum_{i=0}^{j} WT_i$……………………………………….…. (6)

$min \sum_{i=0}^{j} CostVM$……………………………….…. (7)

IV. PROPOSED ALGORITHM

In this section, the proposed methodology is described along with the algorithm. Basic scheduling process in cloud sim along with proposed modifications is shown in Fig. 1.

First of all, all libraries need to be initialized containing the cloud sim functions, packages, classes, and methods. The next step is the creation of datacenters that contains the number of hosts in particular datacenter. The broker is created in the next step which contains all the information like the number of required virtual machines, the number of requests per second, scheduling policies, etc. In proposed model, the default broker is extended and a new broker is created with a new scheduling policy to decide how a virtual machine is selected for a particular task. The next step is creation of virtual machines and these virtual machines are allocated to hosts as per default policy of the cloud sim. When tasks are submitted to broker for execution, MyBroker calls the proposed VMSelect method to display results on screen.

The proposed algorithms work in the following manner. The cloud datacenter default class is extended to Mydatacenter and a new scheduling function is inserted to submit jobs and list of virtual machines. When a task comes to the broker, the value of the priority tag is stored into variable. By default, index of virtual machines is set to -1, indicating that no virtual machine has been selected yet. Then from the pool of virtual machine, first virtual machine is selected and its status is checked. If this virtual machine is busy, then next virtual machine is selected from pool and this process is repeated until all virtual machines are checked. If any machine is free, then its type is found out, based on value of priority tag of task and virtual machine type. A virtual machine is selected and its index value is sent to broker, where task is bind to it for execution and status of this virtual machine is set to busy state. If status of all virtual machines is busy and there is more task to be bind, then status of all virtual machines is set to free, and algorithm will be called again to select virtual machine. This will add task in waiting state for the virtual machine. The process will be called again until all tasks are bind to the virtual machines as per their selection policy.

Algorithm 1(a)

1. Initialize the variable i=0
2. Repeat for all $Task_i$ i=0 to i=n
3. {
4. int index=Select_VM($Task_i$)
5. VM=GetVM(index)
6. If(Status_VM=VM_NOT_BUSY)
7. {
8. Set Status of VM to VM_BUSY
9. }
10. }

Algorithm 1(b)

1. Initialize the variable index=-1 and size = SizeOfVMlist;

Retrieval Number: B3752078219/19©BEIESP
DOI:10.35940/ijrte.B3752.098319
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

2. Repeat for i=0 to i=size
3. {
4. Get the first virtual machine VM=getVM(i).
5. If(VM_Status==Not_Busy)
6. {
7. If((P->$Task_i$==High) and (VM_Type)==Heavy))
8. {
9. Index=getVMId
10. }
11. Else if ((P->$Task_i$==Medium) and (VM_Type)==Medium))
12. {
13. Index=getVMId
14. }
15. Else if ((P->$Task_i$==Low) and (VM_Type)==Light))
16. {
17. Index=getVMId
18. }
19. }
20. Else
21. {
22. Set VM_Status=VM_Not Busy
23. Set i=i+1.
24. Go to step 2.
25. }
26. }

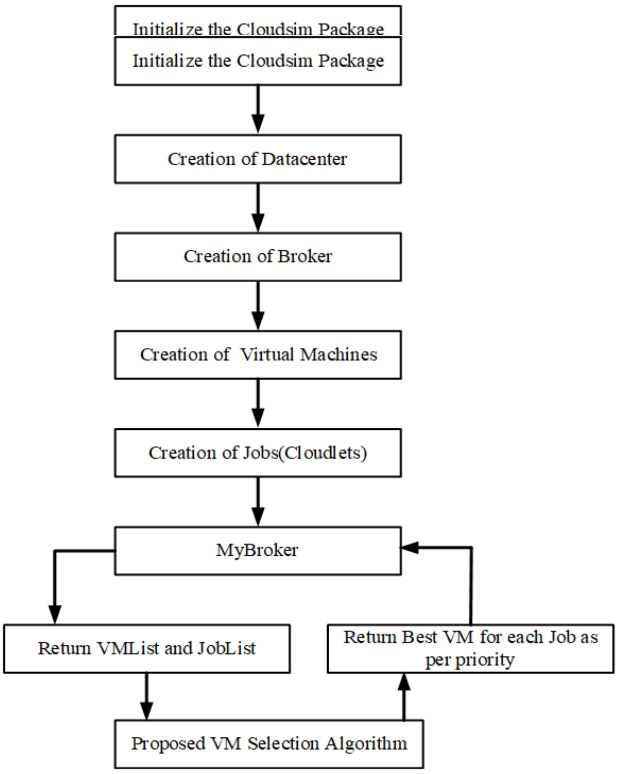

Fig. 1 Proposed Methodology

V. SIMULATIONS

Simulation is carried out using cloud sim toolkit 3.03 [17]. The cloud sim is worldwide most usable tool to evaluate performance of new algorithms. It supports customization and modeling of user-defined policies for allocation of host to virtual machines, selection of virtual machines, and modeling of energy-aware computational resources. The cloud-sim toolkit is a java based simulator that allows researchers to extend seamless modeling for in-built classes of various components to configure as per as user requirement. Cloud sim toolkit 3.03 was configured on 64 bit Windows 10 professional installed on HP Prodesk having 16GB RAM, i7 Processor 3.60 GHz 8 cores. One datacenter is created with six physical machines and three virtual machines namely; VM_High, VM_Medium and VM_Light, same as configured as Amazon EC2. A total of six of virtual machines are created in datacenter and two of each type were considered in order to perform simulations. A task generator can generate any number of cloudlets in the range of [0-30] and the length of cloudlet is defined between [20,000-50,000]. The RAM is set to be as 512 MB, 1024 MB, and 2048 MB for light, medium, and high virtual machine respectively. Similarly, the CPU speed is also set as 1000MIPS, 1500 MIPS and 2000 MIPS for all three types of virtual machine respectively. A total of 10 simulation run are performed on simulator and results are calculated by taking an average of 10 simulations. The simulations are set to perform for different number tasks. The configurations of cloudlets and virtual machine are configured in table 1(a), and 1(b) respectively.

Table I (a)

Cloudlet (Task) Configuration	
Length	20,000-50,000
Priority	1-3
No./sec	0-30

Table I (b)

Virtual Machine **(**VM) configuration	
Number of Virtual Machines	6
RAM	512MB-2048MB
MIPS	1000MIPS-2000MIPS
Number of CPU	1
Scheduler	Space-shared

VI. RESULTS

This section discusses the results obtained from simulations have three parameters namely; makespan, average waiting for time and cost of virtual machine and compares them with some existing ones. The default selection policy built-in with cloud-sim is FCFS and ACO based scheduling method are taken for comparison. The performance of our proposed algorithm is compared with this method as proposed in [15].

Table- II: Makespan (seconds)

No. of Cloudlets	Proposed Priority based Algorithm	FCFS	ACO	Proposed vs FCFS (%)	Proposed vs ACO (%)
6	965.04	1150.23	1050.78	16.10	7.45
12	1918	2115	2035.45	9.31	5.55
18	2550.88	2836.45	2625.15	10.06	2.61
24	3425.85	4021.2	3856.5	14.80	10.70
30	3830.94	4115.25	3945.15	6.90	2.77

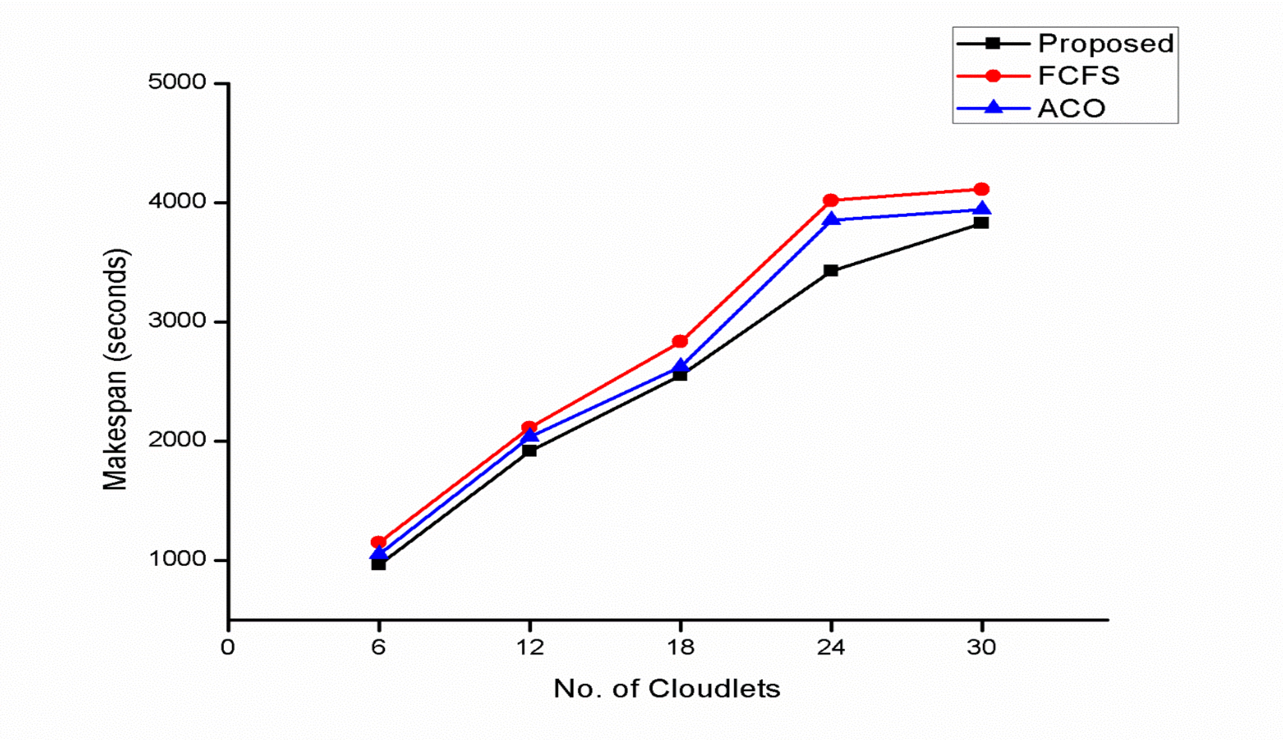

Fig. 2 Makespan vs No. of cloudlets

(a)

Table 2 shows the calculated makespan in seconds when simulations run for different number of tasks submitted to broker. The percentage improvement in makespan calculated using priority-based scheduling is 16.10% more than FCFS and 7.45% more than ACO- based scheduling when number of tasks are 6. When number of tasks are increased to 12, it shows 10.06% more than FCFS and 8.45% more than ACO based scheduling. This percentage improvement is reached maximum to 14.80 % more than FCFS and 10.70% more than ACO based scheduling when number of cloudlets reached to 24. The makespan is less than FCFS because when high priority tasks submitted to broker, it sends to the virtual machine which has high MIPS, so its execution time is low, while it may not get a highly configured machine in FCFS. The ACO algorithm has to calculate a specific path to select the virtual machine, however this time can be minimized by selecting virtual machine from VM list to send it directly. The average percentage improvement in makespan is 11.43% more than FCFS and 5.81% more than ACO.

Table- III: Waiting Time (seconds)

No. of Cloudlets	Proposed Priority based Algorithm	FCFS	ACO	Proposed vs FCFS (%)	Proposed vs ACO (%)
6	0	0	0	-	-
12	519.35	750.325	653.45	30.78	20.52
18	756.35	1010.25	953.45	25.13	30.16
24	1378.35	1635.45	1515.8	25.44	20.67
30	1845.25	2187.5	1985.3	33.87	26.67

Retrieval Number: B3752078219/19©BEIESP
DOI:10.35940/ijrte.B3752.098319
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

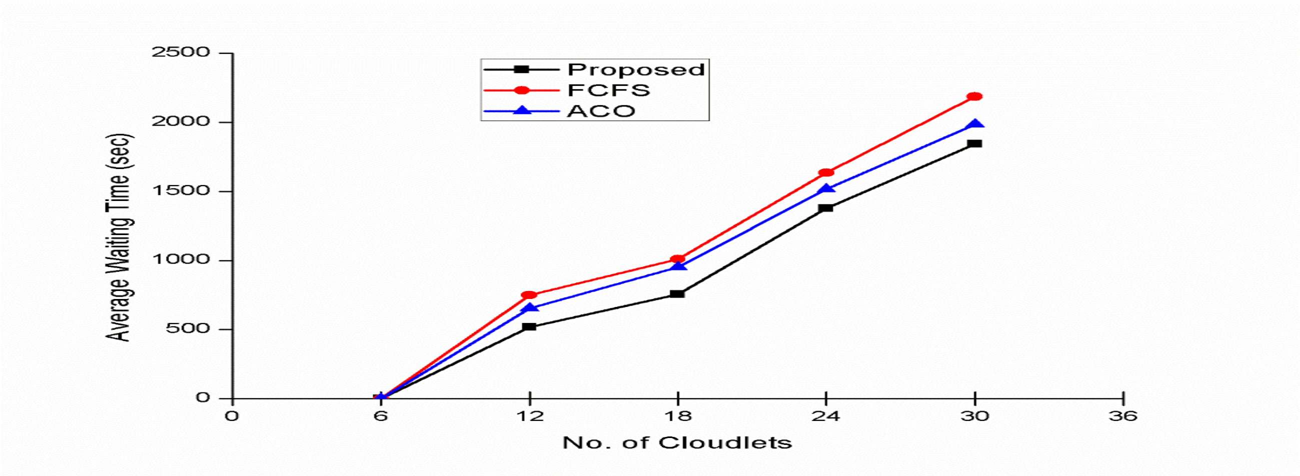

Fig. 3 average waiting time of cloudlets

(b)

Average waiting time calculated by three algorithms are shown in table 4. Figure 2 shows the calculated average waiting time for different number of tasks in all three algorithms. The waiting time is 0 when number of tasks are 6 in all three algorithms as each task starts execution as they are submitted. When the number of tasks increases, waiting time increases in each algorithm. It has been established from the figure 2 that percentage improvement in average waiting time for proposed method is 30.78% and 20.78 % as compared to FCFS and ACO for number of tasks are 12, 25.13 % and 30.16 % when number tasks are 18, 33.87 % and 26.67 % when number of tasks are 30. So, overall average waiting time in our proposed method is 28.80% and 24.50 % as compared to FCFS and ACO based scheduling, hence improving the quality of service.

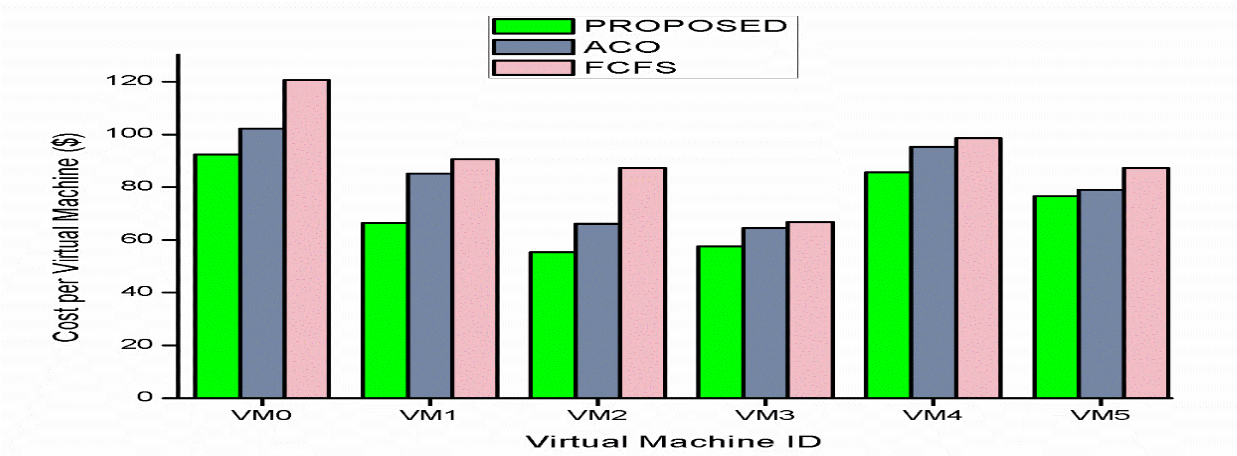

Fig. 4 Cost of Virtual Machine calculated in different algorithm

VM-ID/	Proposed Priority-based Algorithm	FCFS default algorithm	ACO based Scheduling	Proposed vs FCFS (%)	Proposed vs ACO (%)
VM0	92.418	102.2	120.68	27.65	22.91
VM1	66.411	85.24	90.54	28.30	31.26
VM2	55.367	66.21	87.23	48.12	55.16
VM3	57.603	64.512	66.75	14.17	21.24
VM4	85.682	95.25	98.612	13.57	13.76
VM5	76.517	78.915	87.235	13.58	15.56

(c)

Table 4 presents the cost of using each virtual machine which is selected by our proposed algorithm. From figure 4, it can be easily seen that the cost of using virtual machines is less than FCFS and ACO based scheduling. It can be seen from table 1 and figure 4 that % improvement in our algorithm is 21.24 % and 11.54 % more than FCFS and ACO in terms of cost of VM as pay-per-use.

VII. CONCLUSION AND FUTURE WORK

This paper presents a scheduling method based on the selection of virtual machines as per priority of tasks submitted by user.

Retrieval Number: B3752078219/19©BEIESP
DOI:10.35940/ijrte.B3752.098319
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Presented paper helped cloud service providers in selecting appropriate virtual machine after categorization virtual machine into three priority level. It reduced makespan, average waiting time and cost of using virtual machine, giving better quality of service. Thus, cloud service provider can sign more service level agreements to users by promising better quality of service. The cost minimization may help cloud service provider to increase infrastructure. The results justify that proposed priority-based virtual machine algorithm shortens the makespan, by 11.43 % and 5.81 %, average waiting time by 28.80 % and 24.50%, and cost of using the virtual machine by 21.24% and 11.54% as compared to FCFS and ACO respectively, hence improving quality of service. In future, proposed algorithm can be optimized with some artificial intelligent techniques. The demand of users can be used to predict number of users in future, so that cloud service provider can manage their virtual machines in datacenter.

ACKNOWLEDGMENT

This research work is supported by National Fellowship Scheme, UGC, and Govt. of India grant no. NF-2017/18/29146. The author is also acknowledged to reviewers to give their valuable suggestion on this research work and research lab of computer science and engineering department to carry out this work.

REFERENCES

1. A. Gupta and R. K. Jha, “A Survey of 5G Network: Architecture and Emerging Technologies,” IEEE Access, vol. 3, no. 0, pp. 1206–1232, 2015.
2. M. Kumar, S. Sangwan, and S. Singh, “A survey on virtual machine scheduling in cloud computing,” Int. J. Comput. Sci. Eng., vol. 6, no. 3, pp. 485–490, 2018.
3. Sangeeta and S. Sangwan, “Load Balancing in Cloud Computing: A Review Sangeeta,” Int. J. Adv. Res. Comput. Sci. Rev., vol. 9, no. 2, pp. 760–763, 2018.
4. K. R. Remeshbabu and P. Samuel, “Review of the quality of service scheduling mechanisms in cloud,” Int. J. Eng. Technol., vol. 7, no. 3, pp. 1677–1695, 2018.
5. S. Suman, “Implementing Multiple Security in the Cloud Environment,” Int. J. Adv. Res. Ideas Innov. Technol., vol. 2, no. 5, pp. 1–7, 2015.
6. A. K. Bardsiri and S. M. Hashemi, “QoS Metrics for Cloud Computing Services Evaluation,” Int. J. Intell. Syst. Appl., vol. 6, no. 12, pp. 27–33, 2014.
7. L. Liu and Z. Qiu, “A Survey on Virtual Machine Scheduling in Cloud Computing Li,” in 2nd IEEE International Conference on Computer and Communications, 2016, vol. 14, no. 2, pp. 2717–2721.
8. A. R. Arunarani, D. Manjula, and V. Sugumaran, “Task scheduling techniques in cloud computing: A literature survey,” Futur. Gener. Comput. Syst., vol. 91, pp. 407–415, 2019.
9. A. Abdelmaboud, D. N. A. Jawawi, I. Ghani, A. Elsafi, and B. Kitchenham, “Quality of service approaches in cloud computing: A systematic mapping study,” J. Syst. Softw., vol. 101, pp. 159–179, 2015.
10. Y. Ding, X. Qin, L. Liu, and T. Wang, “Energy efficient scheduling of virtual machines in cloud with deadline constraint,” Futur. Gener. Comput. Syst., vol. 50, pp. 62–74, 2015.
11. M. Haque, R. Islam, M. Rubayeth Kabir, F. Narin Nur, and N. Nessa Moon, “A priority-based process scheduling algorithm in cloud computing,” Adv. Intell. Syst. Comput., vol. 755, no. January 2018, pp. 239–248, 2019.
12. M. A. Alwora, A. Al-hashmi, A. Dhari, and A. B. Darem, “Task-Scheduling in Cloud Computing Environment: Cost Priority Approach,” in Proceedings of International Conference on Cognition and Recognition, 2018, vol. 14, pp. 99–108.
13. E. Meriam and N. T. Mediatron, “Multiple QoS priority based scheduling in cloud computing,” in 2016 International Symposium on Signal, Image, Video and Communications, ISIVC 2016, 2017, pp. 276–281.
14. Y. Balagoni and R. Rajeswara Rao, “A cost-effective SLA-Aware scheduling for hybrid cloud environment,” in 2016 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2016, 2017.
15. J. Linda and V. S. Ananthanarayana, “Energy-aware Task scheduling in cloud using Ant-colony Optimization,” in International Conference on Emerging Research in Computing, Information, Communication and Applications, 2014, pp. 1–7.
16. G. L. Stavrinides and H. D. Karatza, “A Cost-Effective and QoS-Aware Approach to Scheduling Real-Time Workflow Applications in PaaS and SaaS Clouds,” Proc. - 2015 Int. Conf. Futur. Internet Things Cloud, FiCloud 2015 2015 Int. Conf. Open Big Data, OBD 2015, pp. 231–239, 2015.
17. R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. Rose, and R. Buyya, “CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms,” Softw. - Pract. Exp., vol. 41, no. 1, pp. 23–50, 2010.

AUTHORS PROFILE

Mr. Manoj Kumar pursued Bachelor of Engineering in Computer Science and Engineering from DCE, Gurgaon, affiliated to MDU, Rohtak in the year 2009. He pursued Master of Technology from CSE Department of DCR University of Science and Technology, Murthal, Sonepat in the year 2011. He is currently pursuing Ph.D. from the CSE Department of DCR University of Science and Technology, Murthal, Sonepat under National Research Fellowship program of University Grant Commission, Govt. of India. He is the member of International Association Engineers.He has published seven research papers in International Journals and presented two papers in National Conference. He is having 5 years of teaching experience.

Professor (Dr.) Suman has been into teaching and research for about 16 years. She did her Ph.D. from Deenbandhu Chhotu Ram University of Science and Technology, Murthal(Haryana)INDIA. Her research areas include Network Security and Heterogeneous Wireless Networks. She received her M.tech. deegree in Computer Science & Engineering from Kurukshetra University,Kurukshetra, INDIA. She has published more than 20 papers in various journals and conferences of repute.