Comprehensive Study of Task Scheduling in Cloud Computing

Arnab Chattoraj[1] Mohit Sharma[2] Prof. Pallab Banerjee[3] Prof. Biresh Kumar[4] Prof. Amarnath Singh[5]

[1,2,3,4,5]Department of Computer Science & Engineering

[1,2,3,4,5]Amity University Jharkhand, India

Abstract— The basic concept of cloud computing is based on centralized storage where service provider maintains the distributed data centers' computation. So sometimes cloud computing is also called as On-Demand Computing. It is basically an amplified form of client-server, cluster and grid computing. The efficiency and performance of cloud computing services are dependent upon the task submitted by cloud users. Task Scheduling is one of the most essential factors which is responsible for incrementing the performance and efficiency of cloud computing environment by through the maximization of resource utilization. This paper covers the study of different task scheduling algorithms used in cloud computing for the achievement of factors such as quality of services, performance and minimal usage of execution time. This paper aims to describe the comparison of different job scheduling techniques with respect to different parameters like response time, execution time, load balance and timespan of job to discover the best and most efficient task scheduling algorithm with respect to these parameters. In this paper tabular form is used for comparison of the different scheduling algorithms in order to find the best algorithm.

Keywords: Cloud computing, Task scheduling, algorithms, load balancing, resource utilization

I. INTRODUCTION

In the world of scientific community, Cloud computing is gaining a prominent attention due to its ability to facilitate an environment which is even flexible as compared to its counterparts. Cloud computing is defined in terms of a model which provides on demand network access to virtually shared pool of resources(networks, servers, applications, databases and many more) in a way which is simple and ubiquitous to the user along with small amount of management efforts[1]. Thus Cloud

Technology provides variety of services in accordance to the demand of its users such as dynamic network access and rapid elasticity. The popularity which cloud has gained is dependent upon its performance, resources management and various optimum job scheduling techniques.

The main focus of this paper is on the various task scheduling approaches. Task Scheduling is the process of selecting the most suitable and appropriate resources for the proper execution of the task given by the user. The tasks are mainly the user's queries which are send to different servers to be accomplished within a required period of time[8]. Task scheduling is based upon the principle of distributing the task on available resources.

In decentralized environment the main thing of scheduling algorithm is to extend different on servers for proper load balancing which leads to the maximum utilization of the processors and minimum extension of the execution time of user's task. The core objective is the scheduling of the available resources in accordance with the available time needed to execute it to completion. The task usually include a query entry, to process that query, or accessing the software or memory required[1].

The task of a user is fetched to one of the servers available and the result or response of the task is returned to the user. Task by the cloud users are forwarded to the available resources for their proper and timely execution is task scheduling[26]. Various task scheduling algorithms are used for the purpose of improving the performance of cloud and enhance the throughput of the servers. The different parameters of scheduling enhances the overall cloud performance[2]. The scheduling of task has some limitation such as time, throughput, cost, resource utilization and make span[26]. The main purpose behind task scheduling is to reduce cost and time to get an optimal output or outcome which in turn leads to enhanced cloud performance. The upcoming sections of the paper is having the description of classification of scheduling and scheduling processes. Section III comprises of Literature Review and Section IV deals with the working process of different task scheduling algorithms aiding in the comparison of different algorithms as well as discusses the most efficient and best scheduling algorithm and we come to the result of this research paper.

II. CLASSIFICATION OF SCHEDULING

There are three main groups of scheduling methods: Resource scheduling, Task scheduling, Workflow scheduling. Resource Scheduling is the efficient distribution of virtual resources among the Servers (physical machine) available. The workflow scheduling deals with scheduling the entire flow of jobs in the given task in an efficient manner. Task scheduling is the assignment of the task to the available resources for its efficient execution. Task scheduling methods has its application for both centralized and decentralized structure as well as for homogenous and heterogeneous system environment[25]. Several types of task scheduling Algorithms are shown in Figure 1. Task scheduling is the main component of cloud computing and looks after the proper allocation of resources to the tasks. The factors which plays the most significant role in task scheduling is time and its cost required for its completion.

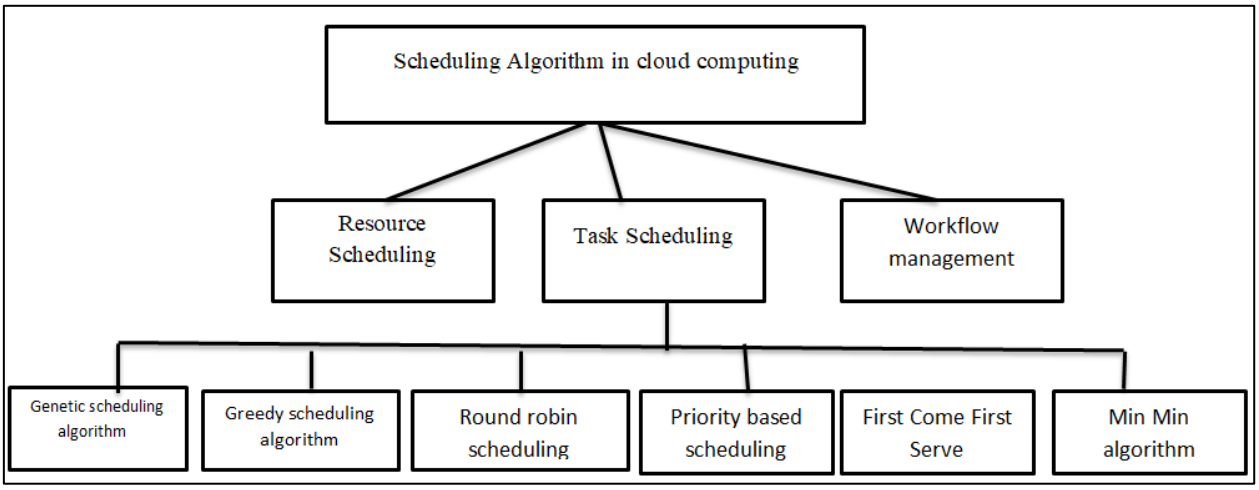


Fig. 1: Classification of Scheduling Techniques [2].

III. SCHEDULING

There are many scheduling Techniques in Distributed Computing Environment and are implemented with applicable parameters. The core objective of these techniques lies in enhancing the performance of the cloud and quick response time of the system[3]. Due to the incapacity of the Conventional Scheduling Algorithms, the overall performance is not increased which leads to increase in the cost and execution time. In this paper we will be discussing Scheduling Algorithms are First Come First Serve, Min-Min ,Round Robin ,Genetic Algorithm, Greedy Algorithm and Priority-based Job Scheduling Algorithm.[4],[24].

A. Scheduling Process

Task Scheduling Process in cloud computing is devided into three stages[7] as shown in Fig 2.

1) Resource Discovery and Filtering: The cloud service provider looks into a specific network to find the list of available resources and also collect and check their working status.
2) Selection of Resources: This stage is of utmost importance and is known as Deciding Stage. Selection of required resources is done on specific parameters and in accordance with requirement of task execution.
3) Submission of Task: As soon as the required resources is selected, the task is submitted to the resource for its execution[34].

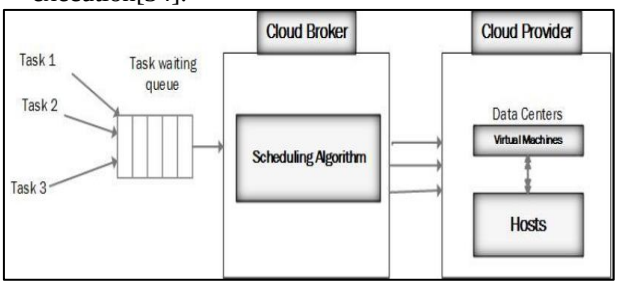


Fig. 2: Task scheduling process[4].

IV. LITERATURE REVIEW

The main issue in the task scheduling is the efficient allocation of the available resources to the new tasks entered by the users. If a number of tasks arrive at the same time, then the process of dynamic resource allocation becomes more complex. To get rid of this problem S. Ravichandran and D. E. Naganathan [7] had proposed a new system which works when a new task arrives which is sent into the queue for waiting and then the job scheduler orders each task and then allocate resources for their execution. In this regard Genetic Algorithm is considered to be the best, all the tasks are sent to the queue, then comes the scheduler and selects the task from the queue and allocate them resources and then execute the task. Thus this system achieves the minimization of the execution time of task and optimization of resource utilization.

In support of this paper, authors V. V. Kumar and S. Palaniswami mainly focused on increasing the efficiency of job scheduling techniques for cloud computing service. They proposed an algorithm which utilizes the turnaround time optimally by giving high priority to different jobs for its completion and less priority for termination issues of real-time task [9].

Siad Bin Alla and Hicham Bin Alla in [22] have described a novel based dynamic task scheduling technique which is based on enhanced genetic algorithm. The working function of proposed algorithm is mainly focused to reduce the execution time, effectively improve the throughput and the scalability of the cloud in task scheduling.

The author proposed a novelistic approach for task scheduling algorithm Quality of Service(QOS) with Genetic Algorithm and Ant Colony "QOS-GAAC" with multi-QOS constraint[33], in this algorithm authors' main focus on expenditures, security, time-consuming and reliability in the process of task scheduling[33]. This algorithm is the result of the combination of genetic Algorithm and ant colony optimization algorithm. The result represent that this algorithm has great performance in both guaranteeing QOS and resource balancing in task scheduling [27].

V. TASK ANALYSIS ALGORITHMS

A. Genetic Algorithm

Genetic Algorithm is the most transformative algorithm and is based on the concept of natural transformation[14].This genetic algorithm has gained popularity in the field of artificial intelligence[3],[6]. In this algorithm the quality of each task is being processed in accordance with the user

requirements until user's satisfaction. It looks after the processing of each task as shown in the Fig3. Darwin's theory proposed the idea of "Survival of the Fittest" which inspired this algorithm and where task allocation to the resources is based on their fitness value function[12].In this the task is not processed as a whole rather it evaluates each parameter of that task on basis of fitness value [4] [20]. The generic terms of this algorithm are as follow:

1) Initial Population

In this algorithm there are several number of individuals which executes the tasks in an iterative way and so a number of solutions are being fixed up, such solutions are termed as populations, in every specific iteration. In that population every solution is termed as chromosome. Ten chromosomes are being carefully chosen from that population [5]. From this initial population, ten chromosomes are selected randomly [6].

2) Fitness Function

The basic purpose of this function is the evaluation of the quality of each individual in the population while depending upon optimization approach. It is oftenly dependent on deadline but in few cases it is dependent on the budget constraints [7].

3) Selection

In this process an operator is used known as proportional selection which evaluates the probability and fitness between two algorithms. It categorizes that either selected probability or next groups are proportional to fitness of each individual [10].

4) Crossover

Purpose of this process is the selection of best fitted pair of individuals for crossing over and this is done with the usage of an operator known as single-point crossover operator. The benefit of crossing over is that both sides' portions can be exchanged [10].

5) Mutation

New individuals are not generated in easily for that purpose; some of gene locus is being exchanged by other gene locus values and it is done in the coding series. A very small value (0.05) is chosen as mutation probability [11].

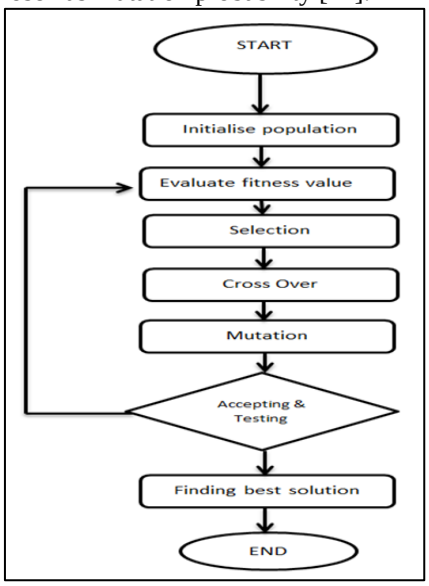


Fig. 3: Working of Genetic Algorithms [5].

B. Greedy Algorithm

Greedy Algorithm is used for problems by making best decision in a particular situation. Working process of Greedy Algorithm is explained in Fig. 4. Optimization issues can be easily solved using this algorithm. Although some problems do not seem easy enough with efficient solutions but they are being solved using of greedy algorithm to get the finest solutions [11]. There are some deviations to the greedy algorithm:

1) Orthogonal greedy algorithms
2) Relaxed greedy algorithms
3) Pure Greedy Algorithms

When some agitations props up such as bad weather and so on, few constraints in above model are being affected due to which the entire schedule become totally impracticable. The core purpose of this algorithm is to solve such problems in each and every step and make the finest decisions. Its main aim was to get the finest solution and keep on working that schedule unless all the problems are being resolved or removed [13]. Due to this optimization of the large problem was further sub divided into small size problems and this helped in identification of solutions in less time [12]. Basic working of Greedy Algorithm is as follows:

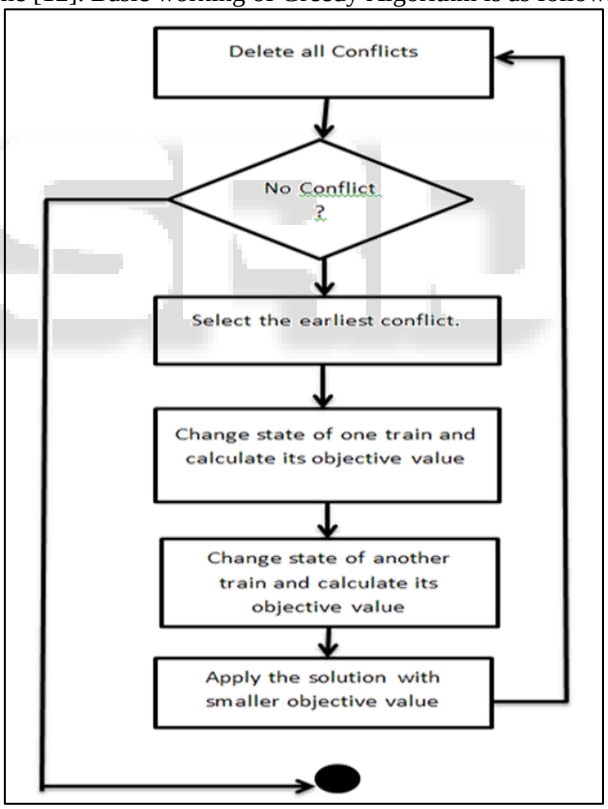


Fig. 4: Working of Greedy Algorithms [12].

Some standards of Greedy Algorithm are as follows [12]:

1) Kruskal's Minimum Spanning Tree (MST)

In this MST is generated by selecting edges not collectively but individually. The greedy choice is always selecting the edge of lightest weight because it will not create a cycle in MST.

2) Prim's Minimum Spanning Tree

MST is being generated again in it but there two sets will be managed: set of vertices which are already being added up in MST and the set of vertices which are not added yet into

it. Those edges are selected which are having less weight [11].

3) Dijkstra's Shortest Path

It is quite similar to Prim's algorithm. In this the shortest path tree is being built up by every single edge. Here we manage two sets: set of vertices which are already being added up in MST and the set of vertices which is not added yet [18]. Greedy choice is selection of the lightest weight path.

4) Huffman Coding

It is a kind of Loss-less compression technique which forms the base of this algorithm. It allocates variable length bit codes to different characters. The Greedy Choice is to assign least bit length code to the most frequently occurred character [11], [12].

C. Priority-Based Job Scheduling Algorithm

In Cloud computing, an innovative approach to solve the programming work is presented by Shamsollah Ghanbari and Mohamed Othman by using mathematical measurements [19]. The importance of the job for programming is considered by this algorithm and is called the Algorithm for priority based job scheduling algorithm "PJSC". It is based on the multiplicative standards decision-making model. There are three levels of priorities in this algorithm that are programming level, resource level and work level which is shown in Fig. 5.

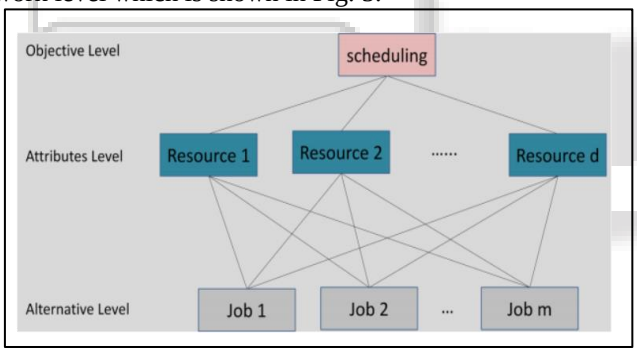


Fig. 5: Three level of priority based scheduling [20].

In this algorithm jobs set are taken as J= {J1, J2, J3, J,…,Jm} that requires assets in a cloud atmosphere and resources set are taken as C= { C1, C2, C3, C4,….,Cd} that is presented in cloud atmosphere as input where (d<<m). Each job set demands a resource with the required priority. Priority of different jobs set is compared independently [28]. Each job is allocated a resource with the specified priority. Hence, the correlation networks of each activity/job set are computed in accordance to the prospects of retrieving the resources, and the matrix of comparison of the resources is also computed. Now the normal matrix of all jobs with the name Δ is calculated by relating the each of the job set matrices and priority paths are also calculated accordingly[21]. Then the normal resources matrix is calculated, and the name of the matrix is given as γ. Now the PVS (priority vector of S) is calculated in this algorithm and S is stated as a set of jobs. The matrix Δ is multiplied with the matrix γ which is results as PVS. Now the highest ranked job is selected, and resource is allocated to that job. Job's list is upgraded with the time, and the programming procedure proceeds until the point that all jobs are planned in a suitable resource [33]. The trial/experiments come about show that the calculation of the algorithm has the rational complexity. There are additional issues which are identified with this calculation, for example, completion time, consistency and complexity [19], [21].

D. Round Robin

The round robin algorithm is a simple example of load balancing technique. A round robin technique was developed to divide scheduling time equally among all scheduled task, in which all tasks get in queue list, and each task gets an equally small unit of time as clearly explained in Fig. 6. The major concern of RR is to emphasis on equal dividing of load to all resources [14]. A cyclic approach is used in round robin. The scheduler selects a task and assign it to the controller and after time expires for the first task, it then move to next task [17]. This is the cyclic approach in which all task assigned to the controller at least once and then scheduler again pick up the first task again.

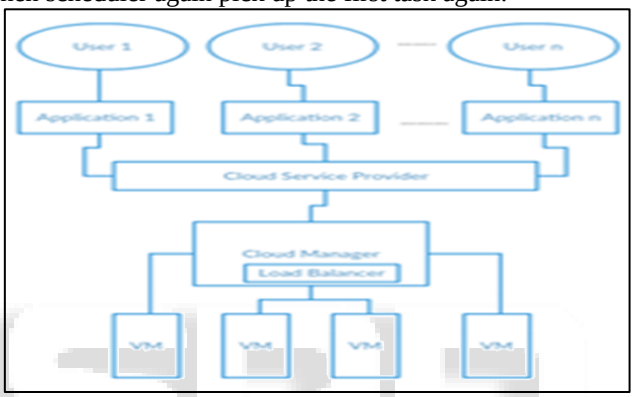
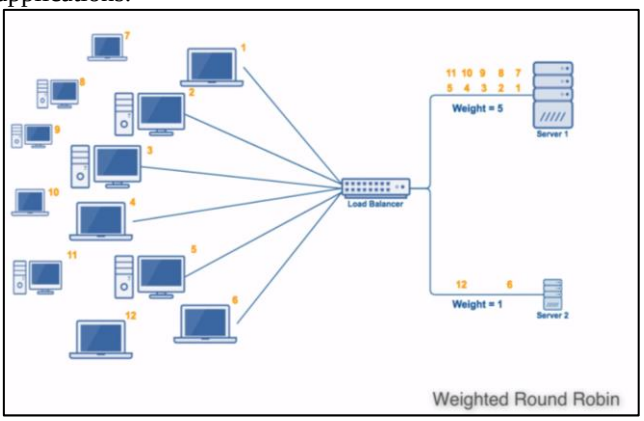


Fig. 6: Load Balancer Round Robin [10].

The load balancing and response time are much better in comparison to other algorithms. The working of Round Robin in cloud environment is same as round robin in process scheduling [15]. In round robin, each task has an equal opportunity to be chosen [28]. There are many variation of Round Robin Algorithms such as :

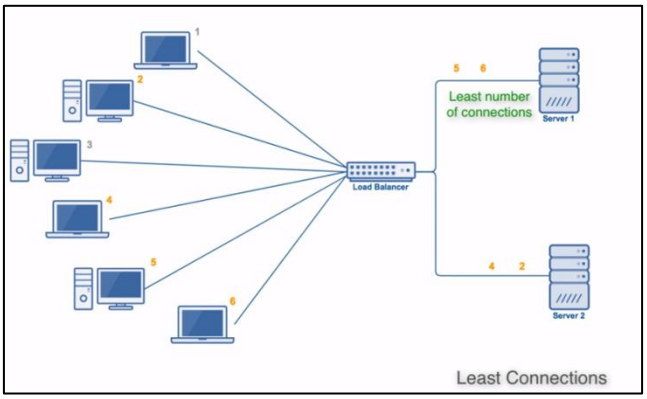
1) Weighted Round Robin:

In this the resources (i.e. servers) do not get easily overloaded and is best suited for business critical applications.



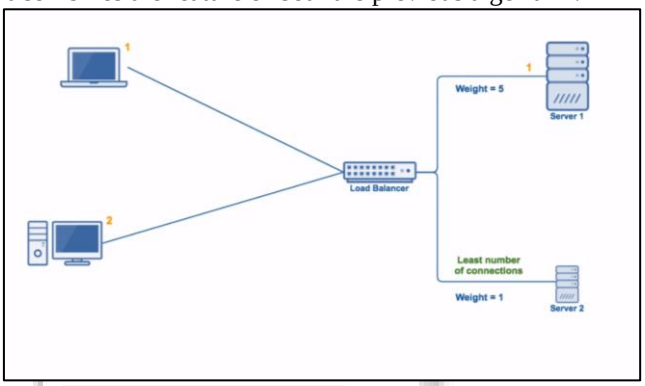
2) Least Connection:

This algorithm takes into consideration the least number of current connection each server has and then the load balancer will try to determine which server has least number of connection and then assign the new connection to that server.



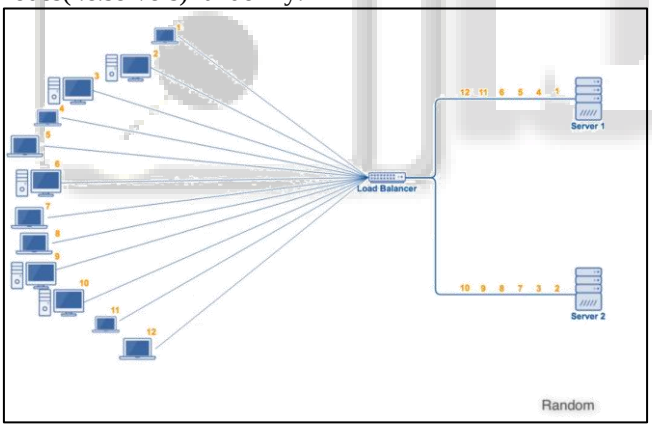
3) Weighted Least Connection:

It combines the feature of both the previous algorithm.



4) Random:

This algorithm distributes the load among the nodes(i.e.servers) randomly.



E. Min-Min Algorithm

This scheduling technique is based on a strategy in which task which has least execution time is selected for all task. This algorithm begins with a set of all jobs which are not assigned and continue to execute until the whole set of the job is empty [16]. In Min-Min Algorithm jobs having the greater time or long task is not to be considered first and the task having greater time will always follow the shorter jobs. In this algorithm completion time of all tasks is calculated then job having least completion time is scheduled over resource [16].

The formula for calculating completion is as follows:

$$T_{finish} = T_{start} + T_{exe}$$

T_{finish} is denoting finish time, T_{exe} is denoting expected execution time, and T_{start} is denoting starting time of the task

$$T_{comp} = \max(T_{finish} = T_{start} + T_{exe})$$

This Algorithm causes all set of tasks which are earlier executed to get a longer time and unbalanced load [23].

F. First Come First Serve Algorithm

It is the simplest and most fundamental technique which schedules the task by utilizing the task arrival time on cloud environment. The execution and scheduling of the task is depending upon which task has arrived first into the queue, thus it is entirely dependent on the arrival time and none other parameter. The task scheduling is done in the correct order of jobs. The task or user request which arrives first to data centre will be sent to the virtual machine first for its processing. The data centre controller looks for free Virtual machine and then provides task to that VM, then remove that task from queue.

If four task arrives on cloud environment having three virtual machines then FCFS scheduler will schedule three task on VM in parallel manner leaving one task until one VM becomes free for first schedule. For second schedule if task 4 has Childs then child can't be executed until their parents gets executed. When task 4 is executing on VM then the two other VM remains idle which cause the less utilization of resources.

VI. COMPARISON OF EXISTING TASK SCHEDULING ALGORITHMS

Task scheduling algorithms that we discussed earlier are compared in by their methodology.

Algorithms	Methodology	Parameters	Merits	Demerits
Genetic Algorithm	Genetic algorithm describes the solution and suitable function to evaluate the solution domain	1.Crossover Probability 2.Population size 3.Mutation Probability	1.Easily Understandable Concepts 2.Accurate Solution of Mathematical and Financial problems	1. Slow Algorithm 2. Inaccurate Solution
Greedy Algorithm	This algorithm tries to discover the global optimum by following the problem-solving heuristic approach for choosing every step.	1. Parameter μ 2. Domain D 3. Population n	1. Easy implementation. 2.Execution and Implementation is fast	1. Global optimization solution is not fulfilled by this algorithm. 2. Very difficult to make changes in parameters.
Priority Based Job	Dependency mode	1. Priority to each queue	1. Priority of the process increases with the increases in the	1. Jobs having lowest priority will be lost when the system

Scheduling Algorithm			time. 2. Easy to use and user-friendly.	crashes. 2. Starvation for resources they need.
Round Robin	The algorithm functions on cyclic approach in which each task has equal chance to be chosen and has an equal small unit of time for execution	1. Arrival time 2. Time slice	1. Response time is good. 2. Load is balanced. 3.less complex	1. Pre-emption causes the process out once time slice expires
Min-Min Algorithhm	This algorithm works on strategy in which task having least execution time is used in relation for all task	1. Makespan	1. Better makespan	1.load imbalance 2.poor Quality of Service
First Come First Serve	This algorithm manages the task scheduling with FIFO queue. Task which comes first will be executed first on VM	1. Arrival time	1. Simple and fast execution	1. Task scheduling is based on arrival time, doesn't consider any other criteria 2.Less utilization of VM

Table 1: Comparison of Mentioned Task Scheduling Algorithms

VII. DISCUSSION

Task Scheduling is one of the challenging components of the Cloud Computing domain. The principle purpose of task scheduling is to distribute the incoming tasks from users to the available virtual machines with respect to the different parameters Load Balancing, execution time, load balance, Quality of service, performance, response time and fairness resource allocation in which task can be executed efficiently. Few of the algorithms consider only load balancing whereas some consider response time. Mostly algorithms works with one or two parameters, due to which good result is difficult to be achieved effectively. Better results can be produced by coupling more scheduling metrics to generate one efficient algorithm as improvement but little bit complexity introduced.

VIII. CONCLUSION

Efficient scheduling algorithm can yield more desirable services and results to users and enhance the performance provided by cloud environment. The core objective of task scheduling in cloud environment is to minimize the execution time of tasks and to increase the resource utilization. A short description of each algorithm methodology has been presented and most algorithms are depended on one or two parameters. More satisfactory results can be obtained by adding more metrics to existing set of algorithms. Table I is constructed upon different scheduling parameters such as execution time, load balance, Quality of service, performance, response time and makespan. The main problem in task scheduling is load balancing, response time, resource utilization and memory storage. Efficient scheduling algorithm can be achieved by amalgamating different parameters to existing algorithms which will increment their overall performance of cloud environment.

REFERENCES

[1] S. Sonia, "Task Scheduling in Cloud Computing," Internation Journal of Advanced Research in Computer Engineering & Technology (IJARECT), vol. 4, no. 6, June 2015.
[2] D. a. S. N. R.Nallakumar, "A Survey of Task Scheduling Methods in Cloud Computing," International Journal of Computer Science and Engineering (JCSE), vol. 2, no. 10, 31 Oct 2014.
[3] P. H.Srimathi, "Survey and Analysis of Task Scheduling in Cloud Environment," Indian Journal of Science and Technology, vol. 9(37), October 2016.
[4] M. B. Yougita Chawla, "A study on Scheduling Methods in Cloud Computing," International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), vol. 1, no. 3, October 2012.
[5] F. A. O. Safwar A. Hamad, "Genetic-Based Task Scheduling Algorithm in Cloud Computing Environment," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 7, no. 4, 2016.
[6] T. Y. K. J. K. K. a. J. S. L. S. H. Jang, "The study of genetic algorithm-based task scheduling for cloud computing," International Journal of Control and Automation, vol. 5, pp. 157-162 , 2012 .
[7] T. G. a. A. Agrawal, "Host Scheduling Algorithm Using Genetic Algorithm In Cloud Computing Environment," International Journal of Research in Engineering & Technology (IJRET), vol. vol. 1 , 2013 .
[8] R. R. a. R. N. C. R. Buyya, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," in High Performance Computing & Simulation, 2009. HPCS'09. International Conference on, 2009.
[9] B. M. P. D., J. K. M. a. S. D. K. Dasguptaa, "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing," International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA), 2013.
[10] S. A. H. a. A. Omara, "Genetic-Based Task Scheduling Algorithm in Cloud Computing Environment," (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 7, no. 4, 2016.

[11] A. S. a. M. V. S. A. Malik, "Greedy Algorithms," International Journal of Scientific and Research Publications, vol. Volume 3, no. Issue 8, August 2013.
[12] Z. HE, "Research on Improved Greedy Algorithm for Train Rescheduling," Seventh International Conference on Computational Intelligence and Security, 2011.
[13] V. K. Z. B. S. B. N. Krivsha, "Greedy algorithms for granular computing problems in spatial granulation technique," XIIth International Symposium «Intelligent Systems», INTELS'16, 5-7 October 2016, Moscow,Russia.
[14] M. Ann Arbor, "J.H. Adaptation in Natural and Artificial Systems," in University of Michigan Press, Holland.
[15] J. P. a. X. F. Gaochao Xu, "A load Balancing Model Based on Cloud Partitioning for the Public Cloud," TSINGHUA SCIENCE AND TECHNOLOGY, vol. 18, no. 1, pp. 34-39, Febuary 2013.
[16] J. L. a. J. X. Gang Lui, "An Improved Min-Min Algorithm in Cloud Computing".
[17] E. a. GibetTaniHicham, "Smarter Round Robin Schedulling Algorithm for Cloud Computing and Big Data," Journal of Data Minning and Hummanities, 23 January 2017.
[18] M. Ann Arbor, "J.H. Adaptation in Natural and Artificial Systems;," University of Michigan Press.
[19] S. a. M.Othamn, "A Priority Based Job Scheduling Algorithm in Cloud Computing," International Conference on Advances Science and Contemporary Engineering , 2012.
[20] W. H. Z. J. Yin H., "An Improved Genet ic Algorithm with Limited It erat ion for Grid Scheduling," Yin H., Wu H., Zhou J., "An Improved Genetic Algorithm with Limited It erat ion for Grid Scheduling", IEEE Sixt h Internat ional Conference on Grid and Cooperat ive Comput ing, 2007. GCC 2007, Los Alamitos, CA.
[21] S. P. a. U.Bhoi, "Priority Based Job Scheduling Techniques in Cloud Computing: A Systematic Review," INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH , vol. 2, no. 11, NOVEMBER 2013 .
[22] S. B. A. A. E. a. A. M. Hicham Bin Alla, "A Novel Architecture with Dynamic Queues Based on Fuzzy Logic and Partical Swarm Optimization Algorithm for Task Scheduling in Cloud Computing," Advances in Ubiquitous Networking 2, Lecture Notes in Electrical Engineering, 2017.
[23] A. I. E.-D. M. F. A.-r. El-Sayed T. El-kenawy, "Extended Max-Min Scheduling Using Petri Net and Load Balancing," International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, vol. 2, no. 4, September 2012.
[24] D. B. J. K. J. B. Shalmali Ambike, "An Optimistic Differentiated Job Scheduling System for Cloud Computing," International Journal of Engineering Research and Applications (IJERA), vol. 2, no. 2, pp. 2248-9622.
[25] M. J., M. B. S. a. M. K. T. Arash Ghorbannia Delavar, ""RSDC (RELIABLE SCHEDULING DISTRIBUTED IN CLOUD COMPUTING)"," International Journal of Computer Science, Engineering and Applications (IJCSEA) , vol. 2, no. 3, June 2012.
[26] Saeed Parsa and Reza Entezari-Maleki, " RASA: A New Task Scheduling Algorithm in Grid Environment," World Applied Sciences Journal 7 (Special Issue of Computer & IT), pp. 152-160, 2009.
[27] Mrs.S.Selvarani1 and D. Sadhasivam, "improved cost-based algorithm for task scheduling in Cloud computing," IEEE, 2010.
[28] A. I. E.-D. M. F. A.-r. El-Sayed T. El-kenawy, ""A Priority based Job Scheduling Algorithm in Cloud Computing," International Conference on Advances Science and Contemporary Engineering 2012 (ICASCE 2012).
[29] M. S. S. Kalra, "A review of metaheuristic scheduling techniques in cloud computing Egypt. Inf. J. 16(3), 275–295 (2015). Elsevier".
[30] J. J. e. a. Durillo, "Multi-objective particle swarm optimizers: An experimental comparison," International Conference on Evolutionary Multi-Criterion Optimization. Springer Berlin Heidelberg, 2009.
[31] A. a. S. K. Verma, "Bi-criteria priority based particle swarm optimization workflow scheduling algorithm for cloud," Engineering and Computational Sciences (RAECS), 2O14 Recent Advances in. IEEE, , 2014.
[32] Y. L. Y. L. X. Dai, "A task scheduling algorithm based on genetic algorithm and ant colony optimization algorithm with multi-QoS constraints in cloud computing," In: 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, pp. 428-431, IEEE (2015).
[33] A. R. M. Beegom, "Beegom, A., Rajasree, M.: A particle swarm optimization based pareto optimal task scheduling in cloud computing," Lecture Notes in Computer Science, p. 79–86, Springer (2014).
[34] P. Salot, "A survey of various scheduling algorithm in cloud computing environment," International Journal of Research in Engineering and Technology, vol. 2, no. 2, pp. 131-135., 2013.
[35] "Parallel Workloads Archive: NASA Ames iPSC/860. http://www.cs.huji.ac.il/labs/parallel/ workload/l_nasa_ipsc/".

