International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-5, June 2019

An Efficient Scheduling Algorithm to Improve the Use of Resources in Cloud

Misba Farheen, Santosh Anand, Somnath Sinha

Abstract—Cloud Computing Technology is currently everywhere and the popularity is expanding day by day. It is a stage for providing unique pool of resources and virtualization to the Cloud clients. The most significant challenge is to have an efficient Cloud Scheduling algorithm. The proper use of resources is an elemental aspect of Cloud Computing. In this paper, An Enhanced Particle Swarm Optimization (PSO) protocol is proposed and this study compares the proposed Enhanced PSO strategy with existing standard PSO algorithm by having this algorithm implemented on CloudSim toolkit. The results obtained that the presented Enhanced PSO protocol helps to obtain minimum execution time of jobs, maintains the high Throughput and improves the Resource Utilization compared with the other Scheduling algorithms Evaluated in the present paper.

Keywords— Cloud computing, Scheduling algorithm, PSO, Enhanced PSO, FCFS, RR, CloudSim.

I. INTRODUCTION

Cloud Computing is the improvement of grid, parallel and distributed computing. In this modern era, Cloud Computing is the hottest content available that offers everything in the form of a service. There are three types of service: Software as a Service (SaaS) –It's the top of the cloud layer which provides on demand software. Platform as a Service (PaaS) –permits computing platform, layers are Cloud OS and Cloud Middleware and lastly Infrastructure as a Service (IaaS) – It's the basic Cloud layer that gives access to the resources in a virtualized environment such as Amazon EC2. The main purpose is to provide the services to clients. Cloud is categorized according to the Deployment Model: Public, Private, Hybrid and lastly Community.Job Scheduling or Workflow Scheduling is one among the main and challenging concerns in cloud environment. Scheduling is the allocation of different jobs to the allocation of resources in relation to users' specific constraints.Such constraints could be Execution Time, Throughput, Waiting time, Makespan, Resource utilization and Performance. By considering these parameters various types Metaheuristic Scheduling Algorithms are used to allocate resources. Metaheuristics techniques such as PSO, Ant Colony Optimization, Genetic Algorithm and bee's algorithm. Resource Utilization is a necessary task and it's a main research area. Therefore, the Scheduling [8] plays a vital role in cloud computing environment to efficiently allot resources to each task.

Manuscript published on 30 June 2019.
* Correspondence Author (s)
Misba Farheen, Department of Computer Science, Amrita School of Arts and Sciences, Mysuru
Santosh Anand, Department of Computer Science, Amrita School of Arts and Sciences, Mysuru
Somnath Sinha Department of Computer Science, Amrita School of Arts and Sciences, Mysuru

Job Scheduling can be done in different ways based on different metrics can be statically allocated at compile time to pool of resources or at runtime, its dynamically assigned. In this paper, PSO based Job Scheduling protocol [12] is proposed for Scheduling of jobs while comparing with the other Scheduling methods to reduce the completion time of Jobs, increasing resource usage in an effective manner and also results in high Throughput of the system. The Enhanced Particle Swarm Optimization protocol provides near optimal solution within a reasonable time frame.

CloudSim is the Simulation tool [9] used for the above proposed methodology. A Java environment for cloud modeling, service brokering, provisioning, and policy allocation. It shows the result of the power, time and traffic consumption. CloudSim enables simulation, seamless modelling and experimentation of cloud environment.

The remaining content of this paper is ordered as follows: Literature Survey is presented in Section II. Methodology is presented in Section III. Proposed Methodology is presented in Section IV. Section V covers the Results of Cloud Simulation using a CloudSim Toolkit. And the work is concluded at last Section VI.

II. RELATED WORK

Rajveer Kaur et Al. (2014) proposed Enhanced Genetic Algorithm [1] based on Task Scheduling in Cloud Computing with new fitness function according to its mean and large mean values, the genetic algorithm is enhanced. Parameters are Turnaround time, Throughput, Fairness, Waiting time, Response time. In future work we can test this system by increasing the number of resources, including the number of jobs, so as to escalate the overall performance of the system.

Teena Mathew (2014) proposed Study and Analysis of Various Task Scheduling strategies [2] in Cloud using different scheduling algorithms to improve the metrics such as use of CPU, turnaround time, cumulative output also it presents a detailed study for the cloud environment of different scheduling methods.

Mala Kalra et Al. (2015) proposed scheduling metaheuristic approach [3] in cloud computing it Improves the Communication cost and execution cost by implementing various metaheuristic algorithms with metrics Makespan, Flow time, Cost, Waiting time and Turnaround Time. We can improve heuristic approach by using this efficient metaheuristic algorithms such as ACO, BAT algorithms.

Medhat Tawfeek et Al. (2015) aimed cloud task scheduling based on ant colony optimization algorithm (ACO) [4] with the best values of factors for the optimization algorithm ACO is determined experimentally. Length, Bandwidth, Makespan, Performance are the parameters used in this paper.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

The result of task priorities and load balance will be taken into account and other metaheuristic approaches can be performed in future work.

V. Murali Mohan et Al. (2015) proposed efficient task scheduling strategy [5] for QoS ideal resource usage in cloud using optimal task scheduling algorithm. Metrics such as scalability and robustness. In future work we can improve the utilization of resources.

Nima Jafari Navimipour et Al. (2015) proposed Task Scheduling in the Cloud Computing based on the Cuckoo Search Algorithm [6] the algorithm's speed and coverage become very high. Parameters such as flexibility, availability and many heuristics have been discussed in this work. Ala'a Al-Shaikh et Al. (2016) proposed Resource Utilization in Cloud Computing as an Optimization Problem using greedy algorithm [7] the proposed protocol is based on the problem of 0/1 Knapsack and activity selection. Parameters are time and size. In future work we can obtain an efficient algorithm that can give higher profits. Syed Arshad Ali et Al. (2016) proposed A Comparative Study of Task Scheduling Algorithms [8] in Cloud Computing using different scheduling algorithms such as Cuckoo Optimization algorithm to upsurge better performance of the cloud system. Parameters are response time, throughput, makespan, load balancing, fault tolerance. In future work this work can be used as reference and we can enhance or propose a new scheduling algorithm using this work. Gibet Tani Hicham et Al. (2016) proposed Cloud Computing CPU Allocation and Scheduling Algorithms Using CloudSim [9] Simulator. In this work they have described and discussed about the cloud simulation environment. Also, they showed the number of experiments shown using the CloudSim toolkit to check and estimate the scheduling algorithm performance.

Anshul Ahluwalia et Al. (2016) aimed A Review on Task Scheduling Practices [10] in Cloud Computing. Here they have been discussed survey of various task scheduling algorithms with their pros and cons. Parameters have been discussed in a detailed manner.

Akshay Kumar Gupta et Al. (2017) proposed Efficient Resource Utilization in Virtual Cloud Computing Environment using an Enhanced version of MaxMin task scheduling algorithm [11] that improves the turnaround time. Parameters such as performance, size, memory, bandwidth. Its already enhanced algorithm which gives better results.

Mohd Farhan Ngatman et Al. (2017) proposed A Study on Modified PSO Algorithm [12] in Cloud Computing. It declines algorithm's likelihood of converging to the finest solution. Parameters are makespan, throughput. In future work we can still add more parameters and improve overall system performance.

III. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization algorithm is one among the optimization techniques established in 1995 by Eberhart and Kennedy. It is a nature stimulated algorithm that remained significantly inspired by the popular activities of fish schooling or swarm of birds. In this algorithm there are certain parameters which are Particles, similar to individual objects or it can be referred as potential solutions, also interact with each other. Fitness value is referred as best position in a search space there are two values: one is pBest value (Personal best) a value of each individual particle and second is gBest value (Global best) a value of overall population or particles. Velocity is the position of each particle moving.

There are three steps in the PSO algorithm:
Step 1: Estimate fitness of each individual particle
Step 2: Update the best individually and globally
Step 3: Update the speed and position of each particle
Step 4: These steps will be repeated until there is a stopping state.

IV. PROPOSED METHODOLOGY

The proposed work is to implement a new Enhanced PSO algorithm in one of the Java Environment using CloudSim Toolkit. An Enhanced PSO has proved to be more Effective and fast for solving various optimization problems when compared to the standard PSO algorithm the aim is to improve efficiency of our resources and reducing completion time of tasks with high throughput. A user server design is been made. Where the user submits various jobs and sends them to the server to perform those jobs. As soon as the server receives the jobs and initializes the available resources as well. After it gets initializes it initiates the tasks then the importance is been allotted according to various other factors. Presently the Scheduler assigns the resources matching to their capacity then jobs which are obtainable are set into a collection and the implementation is completed by various algorithms like FCFS, SJF, PSO and Enhanced PSO. Therefore, the outcomes are analyzed based on the duration for executing individuals' jobs.

Formula:

$$V[new] = V[cur] + c_1 * a_1 * (pBest - present[]) + c_2 * a_2 * (gBest[] - present[])$$

$$present[new] = present[cur] + V[cur]$$

Whereas,
V → Particle's velocity
c1 & c2 are the learning factors between 1 & 2
present → Current Particle's position / solution
a1 & a2 are the random numbers between 0 & 1

Algorithm: Enhanced Particle Swarm Optimization
Step 1: Start
Step 2: Population to be initialized in a space search
Step 3: Calculate the fitness criteria
Step 4: The pBest value (the best personal value) is updated
Step 5: The gBest value (the best global value) is updated
Step 6: After finding these two best values, update particle's new position and velocity based on pBest (Personal best) and gBest (Global best or neighborhood) values using given formula.
Step 7: Repeat these steps from 3 to 6 Until it reaches terminal condition or if the target is achieved.
Step 8: Output the ideal solution as the final solution
Step 9: End

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

V. THE EXPERIMENT RESULTS ANALYSIS

CloudSim is the simulation toolkit selected for simulation with programming tools using NetBeans IDE. This section presents results for assessing the performance and efficiency of the newly proposed Enhanced PSO method which can be compared with other different job scheduling methodologies like FCFS, RR, Existing PSO based on the parameters: Execution Time, Throughput and Resource Utilization which have been implemented on cloud environment.

Table `1: Data Used for Experiments

Number of Tasks	300
Number of Data Centers	75
Population Size	25

A. Execution Time

This Parameter shows that the proposed Enhanced PSO Method results in less Execution time (see Fig 1) and the It is also possible to calculate total time to relate the functioning of the various scheduling algorithms. In the graph the x-axis is the number of data centers present and the y-axis is the Execution Time.

Table 2: Execution Time of different algorithms

Algorithms/Methods	Total Execution Time
FCFS	6098.51
RR	6119.10
EXISTING PSO	6115.45
ENHANCED PSO	5549.16

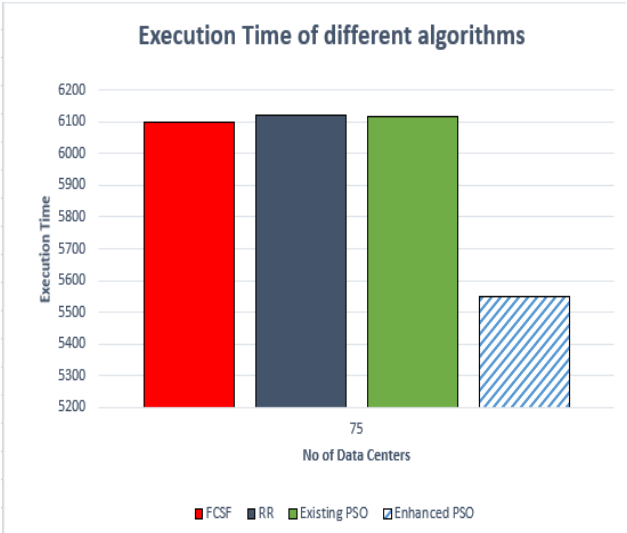

Fig 1. Comparison on the basis of Execution Time

B. Throughput

This Parameter shows that the proposed Enhanced PSO Method provides High Throughput in comparison with the other Scheduling algorithms (see Fig 2). The x-axis is the number of data centers present and the y-axis is the Throughput.

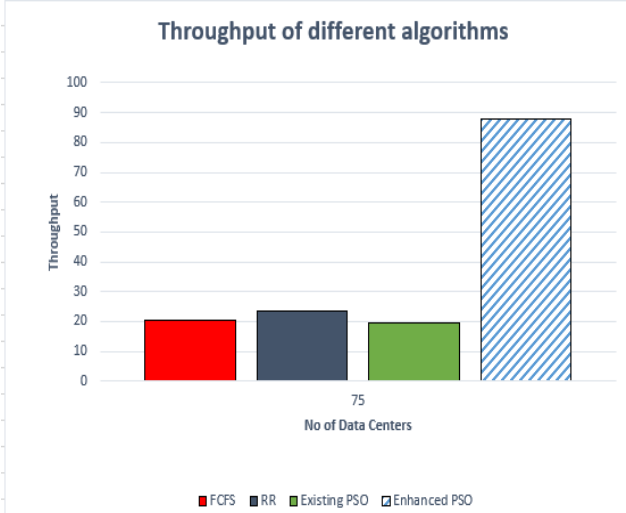

Fig 2. Comparison on the basis of Throughput

Table 3: Throughput of different algorithms

Algorithms/Methods	Throughput
FCFS	20.64
RR	23.73
EXISTING PSO	19.54
ENHANCED PSO	87.89

C. Resource Utilization

Another Parameter shows that the proposed Enhanced PSO Method results in Better Resource Utilization when compared to other Scheduling algorithms (see Fig 3). The x-axis represents the No. of data centers and the y-axis represents the Resource Utilization.

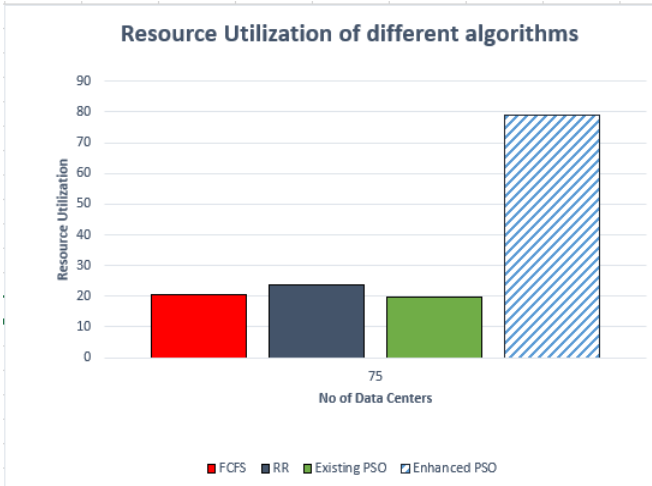

Fig 3. Comparison on the basis of Resource Utilization

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Table 4: Resource Utilization of different algorithms

Algorithms/Methods	Resource Utilization
FCFS	24.76
RR	26.24
EXISTING PSO	19.82
ENHANCED PSO	78.83

VI. CONCLUSION

Cloud computing is the very popular and recent research area and one of the focused areas for improving system performance is job scheduling. The main goal of this proposed work is to establish an effective scheduling algorithm for an efficient resource use of jobs. This Cloud Environment works on various FCFS, RR, PSO and Enhanced PSO scheduling protocols. To evaluate each algorithms' overall performance, it calculates the execution time of jobs, throughput and effective use of resources available. Also, this system can be used to perform comparative analyzes. It is clear from the relative comparison that the introduced methodology is Enhanced PSO which is superior to alternate techniques. This can be demonstrated by computing time analysis as an experimental result analysis.

REFERENCES

1. Kaur, R., & Kinger, S. (2014). Enhanced genetic algorithm based task scheduling in cloud computing. International Journal of Computer Applications, 101(14), 1-6.
2. Mathew, T., Sekaran, K. C., & Jose, J. (2014, September). Study and analysis of various task scheduling algorithms in the cloud computing environment. In 2014 International Conference on Advances in Computing, Communications and
3. Informatics (ICACCI) (pp. 658-664). IEEE.
4. Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. Egyptian informatics journal, 16(3), 275-295.
5. Tawfeek, M. A., El-Sisi, A., Keshk, A. E., & Torkey, F. A. (2013, November). Cloud task scheduling based on ant colony optimization. In 2013 8th international conference on computer engineering & systems (ICCES) (pp. 64-69). IEEE.
6. Mohan, V. M., & Satyanarayana, K. V. V. (2015). Efficient task scheduling strategy towards qos aware optimal resource utilization in cloud computing. Journal of Theoretical & Applied Information Technology, 80(1).
7. Mittal, S., & Katal, A. (2016, February). An optimized task scheduling algorithm in cloud computing. In 2016 IEEE 6th International Conference on Advanced Computing (IACC) (pp. 197-202). IEEE.
8. Ali, S. A., & Alam, M. (2016, December). A relative study of task scheduling algorithms in cloud computing environment. In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I) (pp. 105-111). IEEE.
9. Rani, E., & Kaur, H. (2017, July). Study on fundamental usage of CloudSim simulator and algorithms of resource allocation in cloud computing. In 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-7). IEEE.
10. Gokuldev, S., & Radhakrishnan, R. An Adaptive Job Scheduling with efficient Fault Tolerance Strategy in Computational Grid.
11. Gokuldev, S., Sowntharya, C., & Manishankar, S. (2006). A pro-active fault tolerant deadline hit count based scheduling in computational grid.
12. Rani, E., & Kaur, H. (2017, July). Study on fundamental usage of CloudSim simulator and algorithms of resource allocation in cloud computing. In 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-7). IEEE.

Authors Profile

Misba Farheen,pursuing her MCA in Amrita School of Arts and Sciences, Mysuru and with an overall CGPA of 9. She has completed her Bachelors of Science in St. Philomena's college, Mysuru. Her research interest is in Cloud Computing. Apart from this she did a Software Testing course from one of the best institutes from Bangalore.

Santosh Anand,completed his BE(CSE) from MIT-Manipal, MTech (CSE) from VTU- University and currently doing PhD in Amrita Vishwa Vidyapeetham. He is working as Asst. Professor in the Department of Computer Science, Amrita Vishwa Vidyapeetham, Mysuru campus, India. His research area is Wireless Sensor Network.

Somnath Sinha,completed his PhD in Computer Science from Pacific University, Udaipur, India. He also completed his Master's in Computer Application in IIEST, West Bengal, India. His research interest is in Security and different types of attack detection in MANET. He is presently working as an Assistant Professor in the Amrita Vishwa Vidyapeetham Mysore.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)