IJCSN International Journal of Computer Science and Network, Volume 4, Issue 2, April 2015
ISSN (Online) : 2277-5420 www.IJCSN.org
**Impact Factor: 0.417**



# Performance Evaluation of Task Scheduling in Cloud Environment Using Soft Computing Algorithms

[1] **R. Jemina Priyadarsini,** [2] **Dr. L. Arockiam**

[1] Department of Computer science, St. Joseph's College, Trichirapalli,
Tamil Nadu, 620002, India

[2] Associate Professor, Department, of Computer science, St. Joseph's College, Trichirapalli,
Tamil Nadu, 620002, India

**Abstract -** Cloud computing is a new computing technology which is developing drastically. Scheduling becomes more crucial and essential in this pay as you go model. Analyzing and evaluating the performance of various heuristics and Meta heuristics scheduling algorithms is a crucial work in this large scale distributed systems. Though various scheduling algorithms exist, the paper exposes a comparative analysis and performance of 2 soft computing algorithms in cloud computing. The algorithms considered are Bee Colony Optimization (BCO), and Particle Swarm Optimization (PSO). The algorithms performance is evaluated using cloudsim simulator to provide Quality of Service (QoS) in this task to resource mapping. The measures considered for evaluation are makespan and resource utilization.

**Keywords -** *Cloud Computing, Task Scheduling, Makespan, Resource Utilization, Bee Colony Optimization (BCO), Particle Swarm Optimization (PSO).*

## 1. Introduction

Cloud computing is a distributed system which has centralized server resources to provide on demand network access. It can be accessed at anytime from anywhere. It offers services like Software as a Service (SaaS),Platform as a Service (PaaS), Infrastructure as a Service(IaaS) respectively [1]. It has various deployment models such as public, private ,community and hybrid cloud. Scheduling can be classified on the basis of resource requirement as static and dynamic. Also Virtual Machine (VM) scheduling plays a vital role because cloud computing is a collection of heterogeneous resources that are distributed on different places. Thus there arises the need to assign user request to appropriate physical machine with minimum time and high resource utilization [2]. The major issue in cloud computing is scheduling of task over the available resource.

Task scheduling helps to improves resource utilization and gains maximum profit by improving reliability and flexibility of the system. Proper resource utilization is required to satisfy user constraints and to get maximum usage of resource. In brief scheduling is related with allocation of task to resource to optimize total completion time (TCT), quality of service (QoS) etc. Since it's difficult to fulfill the customer's needs when the number of users increases the need for optimized task scheduling algorithm arises [3].

Scheduling in the cloud environment is an NP-hard problem. With the increased number of user's, the size of associated computing, and sometime the tasks to be scheduled also proportionally increases, the existing task scheduling strategies cannot fulfill its requirements. For these reason better algorithms for task scheduling is needed to reduce computation time and the cost associated with that computing. An efficient task scheduling algorithm directly affects the system performance [4]. In this paper Section 2 describes the related works on heuristics, evolutionary and swarm algorithms, Section 3 explains the performance evaluation of BCO and PSO, Section 4 gives the results of the performance evaluation and concludes with Section 5.

## 2. Related Works

There are various optimization algorithms to solve tasks scheduling problems. MinMin and MaxMin are popular heuristics technique . Some of them are based on soft computing techniques like Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Bee Colony Optimization

(BCO) algorithm, etc. Some existing works in the above given algorithms are explained below:
Chen et al, proposed a user priority guided MinMin scheduling algorithm for load balancing in cloud. The author introduced a load balancing algorithm using MinMin to reduce the makespan and increase the resource utilization. The proposed approach produce better output [5]. A Resource Aware Scheduling Algorithm (RASA) proposed by Mohana Priya et al., combines MinMin and MaxMin and makes use of active monitoring of load balancing and RASA to achieve improved resource utilization and load balancing in cloud systems[6].

SHIS a synthetic method based on genetic approach for independent task scheduling in cloud computing systems was proposed by Arash et al., the aim of the proposed SHIS is to have a goal oriented operations with optimized initial populations . The method achieves improved resource load balancing with minimum makespan [7]. Sourav Banerjee et al. [8] has proposed Genetic algorithm (GA) to schedule the task for cloud service provider. This heuristic search method minimizes the waiting time of the overall computing system. In the proposed technique users send requests to the service provider who stores them in a queue, and then GA select the best job from that queue. In this way the effectiveness of GA minimizes the waiting time. GA based scheduling techniques are used to search optimize solution from a set of probable solution. This technique increases the system's throughput.

Pop et al studied the meta scheduling issues for grids and clouds [9], the work, details the fundamental issues for developing an effective interoperable meta scheduler for e-infrastructures in general and Inter Cloud in particular ,also it describes a simulation and experimental configuration based on real grid workload traces to demonstrate the interoperable setting as well as provide experimental results as part of a strategic plan for integrating future meta schedulers.

Linan Zhu et al ., proposed a ACO algorithm that chooses the target path through the pheromone strength . The task amount spent is less than the other algorithms. ACO achieves QoS requirements and shortest path. Thus ACO gives more efficient results for node distribution and load balancing [10].

Pinal salot [11] studied various task scheduling algorithms and found that disk space management is a critical issue is virtual environment. Bitam proposed a Bee Life Algorithm (BLA) for optimal job scheduling by assigning task to relevant datacenters in an optimal way. The parameters considered is makespan [12]. Mizan et al,[13] proposed a modified BLA with greedy method to gain optimistic value of service and proper resource utilization in hybrid cloud and the proposed method achieves minimum makespan and to get a positive reply at the user's end.
Pandey et al, proposed a PSO based scheduling strategy to minimize total cost of execution. The author compared PSO with Best Resource Selection (BRS) and found that PSO achieves 3 times cost savings and best distribution of workload on resource compared to BRS [14]. Shaobin Zhan &Huo proposed an improved PSO by combining PSO and Simulated Annealing (SA) algorithm and found that it reduced the average operation time, increased resource utilization and proper supply of resource to user request [15].

## 3. Performance Evaluation of BCO and PSO Algorithms

### 3.1 Bee Colony Optimization (BCO)

BCO algorithms solve problems of various domains, like routing problems, Travelling Salesman Problem and NP hard problems. Some problems are solved with BCO concept and others with ABC algorithms [16]. There is a very thin distinction among variants of the Bee system as the agent in all algorithms is a bee. BCO was also framed comprising of initialization, forward pass and backward pass steps. Forward and backward passes are performed till a stopping criterion is met. Then the bees search for an optimal solution. The steps in BCO:

***Initialization:*** Determine the number of bees $B$, and the number of iterations $I$.

**Select the set of stages** $ST = \{st1, st2,\ldots,stm\}$. Find any feasible solution $x$ of the problem. This solution is the initial best solution.
Set $i: = 1$, until $i = I$, repeat the following steps:
Set $j = 1$, until $j = m$, repeat the following steps:

**Forward pass:** Allow bees to fly from the hive and to choose B partial solutions from the set of partial solutions $Sj$ at stage $stj$.

**Backward pass**: Set $j: = j + 1$.
If the best solution $xi$ obtained during the *ith* iteration is better than the best-known solution, **update the best known solution** $(x: = xi)$.
The BCO is simple, flexible and robust for finding optimal solution. It is easy to implement and has less control parameters when compared to other optimization methods.

### 3.2 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a self-adaptive global search based optimization technique introduced by Kennedy and Eberhart. It is similar to population based algorithm like GA. The movement of each particle is coordinated by a velocity which has both magnitude and direction. Particles position is influenced by its best position (local best) and position of the best particle (global best) in the problem space. The performance of the particle is measured by a problem specific fitness value . PSO algorithm has 3 steps, repeated till a stopping condition is met [17]:

1. Evaluate the fitness of each particle
2.Update individual and global best fitness and positions
3. Update velocity and position of each particle

The average time required for each and every task on all the resources is computed. It is generally observed that the time reduces as the cost of communication increases. All the tasks are mapped in the workflow. PSO finds global minima quickly and also attain balanced distribution of workload onto resources. Fitness evaluation is conducted by supplying a candidate solution to an objective function. Individual and global best fitness and positions are updated by comparing newly evaluated fitness against earlier individual and global best fitness, and replacing best fitness and positions as necessary. Velocity and position update step is responsible for PSO algorithm's optimization ability. PSO algorithm is summarized as follows [18].

1. Initialize the swarm $X_i$, the position of particles are randomly initialized within the feasible space.
2. Evaluate the performance $F$ of each particle, using its current position $X_i(t)$.
3. Compare the performance of each individual to its best performance so far: if $F(X_i(t))<F(P_{ibest})$:
   $F(P_{ibest})=F(X_i(t))$
   $P_{ibest}= X_i(t)$
4. Compare the performance of each particle to the global best particle:
   if$F\ (X_i(t))<F\ (P_{gbest})$:
   $F\ (P_{gbest})=F\ (X_i(t))$
   $P_{gbest}= X_i(t)$
5. Change the velocity of the particle.
6. Move each particle to a new position.
7. Go to step 2, and repeat until convergence.

Table .1 Parameters Used

| *Parameters* | *Values* |
|---|---|
| Resources Used | 4 |
| Jobs | 40,80,160,320,640 |
| CPU used | 1 |
| RAM Size | 1 GB |
| Task Size | 1-7 Units |

Table .2 Makespan (In seconds)

| No of Tasks | MinMin | MaxMin | PSO | BCO |
|---|---|---|---|---|
| 40 | 47 | 44 | 44 | 43.9 |
| 80 | 94.4 | 88.4 | 87.7 | 86.5 |
| 160 | 190.2 | 177.7 | 176.2 | 175.3 |
| 320 | 382.9 | 357.1 | 352.1 | 348.9 |
| 640 | 769.6 | 718.7 | 716.2 | 708.8 |

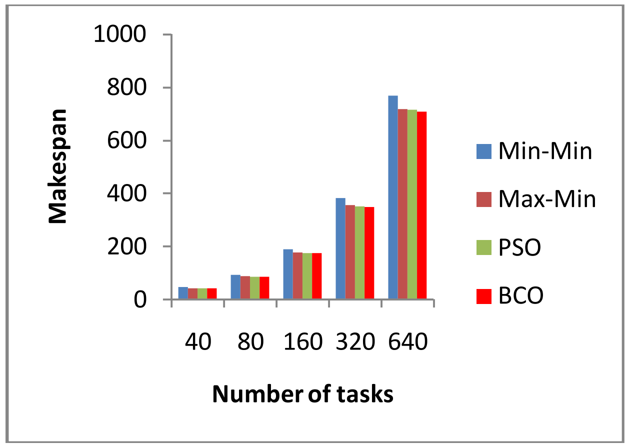


Fig.1 Task Vs Makespan (In Seconds)

Table .3 Resource Utilization

| No of tasks | Min-Min | Max-Min | BCO | PSO |
|---|---|---|---|---|
| 40 | 79.5 | 81 | 81.2 | 81.3 |
| 80 | 79.9 | 78.7 | 79.5 | 80.7 |
| 160 | 81 | 78.8 | 78.8 | 81.9 |
| 320 | 80.4 | 82.2 | 81.6 | 80.5 |
| 640 | 79.7 | 80.7 | 80.7 | 81.3 |

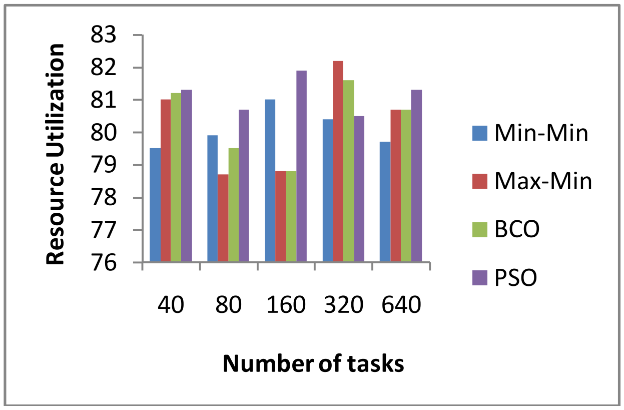


Figure 2. Resource Utilization

Table .2 and Table .3, gives the makespan and resource utilization for job size ranging from 40 to 640. Also Figure.1 and Figure.2 shows the values graphically. BCO reduces Makespan than Min-Min, Max-Min and PSO. Experimental results show that the Makespan is measured for various tasks. At task 40, BCO achieved better Makespan by lowering its value by 6.82% than Min-Min, by 0.23% than Max-Min and by 0.23% than PSO. At task 640, BCO achieved better Makespan by lowering its value by 8.23% than Min-Min, by 1.4% than Max-Min and by 1.03% than PSO. Similarly resource utilization is measured for various tasks. At task 40, PSO achieved a better way by utilizing resources by 2.24% than Min-Min, by 0.37% than Max-Min and by 0.12% than BCO. At task 640, PSO achieved in a better way by utilizing resources by 1.98% than MinMin, by 0.74% than Max-Min and by 0.74% than BCO.

## 5. Conclusion

As the number of cloud users increase with their increase in needs, a good scheduling algorithm is needed to improve the performance. For performance evaluation, we have considered 2 task scheduling soft computing algorithms namely BCO and PSO. The results were also compared with other two heuristics algorithms namely MinMin and MaxMin. The algorithms performance have been evaluated using cloudsim simulator . We found that the Bee Colony Optimization (BCO) gives optimized makespan with better resource utilization. This leads to a need for further optimization and improvement of the solution by providing proper fitness criteria. Also hybridization may lead to better performance. Thus our future work focus on hybrid optimization for efficient Meta task scheduling.

## References

[1] Mayur S Pilavare and Amish Desa i, "A Survey Of Soft Computing Techniques Based Load Balancing In Cloud Computing", International Journal Of Computer Applications (IJCA),(0975-8887), Vol. 110,No 14,January 2015,pp.22-25

[2] Vijaypal S R., Pateriya R R., Rajeev K G,"Survey on load balancing through virtual machine scheduling in cloud computing environment", international journal of cloud computing and services science, (IJ-CLOSER), Vol.3, No.1,February 2014,pp.37-43.

[3] Gunvr Kaur and Sugandha Sharma, "Research Paper on Optimization of Resources Using PSO and Improived Particle Swarm Optimization (IPSO) Algorithms in Cloud Computing", International Journal of Advanced Research in Computer Science & Technology (IJARCST), Vol.2, June 2014 ,pp. 499-505.

[4] L. Guo, S. Zhao, S. Shen and C. Jiang, C, "Task Scheduling Optimization in Cloud Computing Based on Heuristic Algorithm", Journal Of Networks, Vol. 7, No. 3, March 2012, pp. 547-553.

[5] Chen, H., Wang, F., Helian, N., and Akanmu, G. (2013, February)." User-priority guided Min-Min scheduling algorithm for load balancing in cloud computing", National Conference on Parallel Computing Technologies (PARCOMPTECH), 2013 pp. 1-8. IEEE.

[6] S. Mohana Priya and B. Subramani," A New Approach For Load Balancing In Cloud Computing" International Journal Of Engineering And Computer Science( IJEACS ),ISSN:2319-7242,Volume 2 Issue 5 May,2013 pp.1636-1640.

[7] Arash Delvar and Yalda Aryan , "A scheduling heuristics algorithm for independent task scheduling in cloud systems ", (IJCSI) International Journal of Computer Science Issues, Vol. 8, Issue 6, No 2, November 2011,pp.289-295.

[8] Sourav Banerjee, Mainak Adhikari and Utpal Biswas, "Advanced Task Scheduling for Cloud Service Provider Using Genetic Algorithm", IOSR Journal of Engineering, Vol. 2, No. 7, 2012, pp. 141-147.

[9] Pop, F., Cristea, V., Bessis, N., and Sotiriadis, S., "Reputation guided Genetic Scheduling Algorithm for Independent Tasks in Inter-Clouds Environments", In Proceedings of 27th IEEE International Conference on Advanced Information Networking and Applications Workshops, 2013,pp.772-776.

[10] Linan Zhu ,Qingshui Li, and Lingna He, "Study on Cloud Computing Resource Scheduling Stratergy Based On Ant Colony Optimization Algorithm ", International Journal of Computer Science Issues (IJCSI),Vol 9, Issue 5, September 2012.

[11] Pinal Salot , "A survey of various scheduling algorithm in cloud computing environment", *International Journal Of Research Engineering and Technology (IJRET),*February 2013,Vol 2,pp131-135

[12] Bitam, S., "Bees Life Algorithm for Job Scheduling in Cloud Computing", In Proceedings of the Third

International Conference on Communications and Information Technology, 2012, pp. 186-191.

[13] Mizan, T., Masud, S.M.R.A., Latip, R., “Modified Bees Life Algorithm for Job Scheduling in Hybrid Cloud”, International Journal of Engineering and Technology Volume 2 No. 6, June, 2012, 974-979.

[14] Pandey, S., Wu, L., Guru, S.M., and Buyya, R, "A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments", In Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications,2010, pp.400-407

[15] Shaobin Zhan and Hongying HUo, “Improved PSO –based Task Scheduling Algorithm in Cloud Computing”, Journal of Information and Computational Science, 2012 November, pp.3821-3829.

[16] Salim Bitam, "Bees Life Algorithm for Job Scheduling in Cloud Computing", Proceedings of The Third International Conference on Communications and Information Technology, 2012, pp. 186-191.

[17] J. Kennedy and R. Eberhart. Particle swarm optimization.In IEEE International Conference on Neural Networks, volume4, pages 1942–1948, 1995.

[18] W. Qing, and Z. Han-Chao, "Optimization of Task Allocation And Knowledge Workers Scheduling Based-on Particle Swarm Optimization, "In Proceedings of IEEE International Conference on Electric Information and Control Engineering, 2011, pp. 574-578.

[19] Jemina Priyadarsini R and Arockiam L, “Performance Evaluation of Min-Min and Max-Min algorithms for job scheduling in federated cloud”, International Journal of Computer Applications (IJCA),Vol. 99,Number 18 August 2014, ISSN: 0975-8887, pp. 47-54.(IF 0.824)