FOREX Publication
Open Access | Rapid and quality publishing

International Journal of Computer Science Research (IJCSR)
Survey Article | Volume 2, Issue 3 | Pages 89-95 | ISSN XXXX-XXXX

Multi-Objective Optimization and its Application in Cloud Computing

Sigma Nayak[1], Subhashis Mishra[2], Debashis Mishra[3] and Utpal De[4]

[1] *P.G Department of Statistics, Utkal University, Vanivihar, India, sigma1994.nayak@gmail.com*
[2] *P.G Department of Statistics, Utkal_University, Vanivihar, India, mishrasubhashis999@gmail.com*
[3] *Technology Analyst, Infosys Limited, Bhubaneswar (SEZ), India, debashis.sw@gmail.com*
[4] *School of Computer Application, KIIT University, Bhubaneswar, India, deutpal@gmail.com*

*__Correspondence:__ *sigma1994.nayak@gmail.com*

ABSTRACT- As an emerging technology, cloud computing which processes a huge amount of data as well as requires a large amount of resource cost and a large part of the user's budget cost. Job arrangement is significant in cloud computing, since it openly disturbs a systems burden and enactment. An actual job scheduling technique needs not only conference the user's wants but also improves the output of the entire system. So, scheduling mechanism works as a dynamic part in the cloud computing. Thus, my protocol is planned to minimize the switching time, increase the resources' utilization and also increase the server performance and throughput. Here we allocate the import to the job which gives better performance to the computer and try the best to reduce the waiting and switching time. The cloud services comprise of altered functionalities at variable costs, and changeable reliability. So the customer's main purposes are to maximize their usefulness, and decrease their costs and risks. For job scheduling problems in cloud computing, a multi objective optimization technique is required to be projected here. An effective job scheduling not only minimizes the load of the resources but also eliminates the budget cost constraints of the users completing multi objective optimization of both performance and cost. We proposed ant colony optimization algorithm here to resolve this tricky. The ant colony algorithm is a probabilistic and indeterminate world-wide optimization algorithm, so it's easy to find an over-all optimal solution. Moreover, it does not depend on accurate optimization and organizational features of the tricky itself. Two constraints roles were used to calculate and view the utility performance and cost. Based on the feedback these two restriction functions ready the algorithm correct the scheduling to process in a timely manner.

General Terms: Optimization, Scheduling, Cloud Computing
Keywords: ACO, Make Span.

ARTICLE INFORMATION
Author(s): Sigma Nayak, Subhashis Mishra, Debashis Mishra and Utpal De;
Received: 17/08/2023; **Accepted**: 25/09/2023; **Published**: 30/09/2023;
e-ISSN: XXXX-XXXX;
Paper Id: IJCSR-030106;
Citation: 10.37391/IJCSR.030106

crossref member
CROSSREF.ORG THE CITATION LINKING BACKBONE

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

Cloud computing delivers on-demands setup entrée to shared pool of resources. It make sure to contact IT resources which are existing in the data centre and shared by others. The data kept in cloud are humble to use and paid for the usage and can be a cessed over the internet. The users using cloud need not to know about who it is construed and cloud also enable the consumer to think of computing as effectively limitless and of minimal cost. The clod schedulers finds out the good-quality resources for an exact job and summit the job to certain system. Cloud schedulers does not have switch the resources and also the summited the job. In cloud job can be executed by any machine but the time of execution can be different. When the jobs were running in the allocated resources the actual time and the expected execution time may vary, so the job sharing has been resolute according to the scheduling. Scheduling algorithms are used to reduce the resource starvation. Job scheduling is vital in cloud computing as it is openly moves the systems load and performance. This scheduling technique wants not only summit the customers but also enlightening the productivity of the entire structure. Many scheduling algorithms are now existing but Ant Colony Optimization (ACO) algorithm is one of the best appropriate algorithms for arranging jobs in cloud computing. In cloud computing some jobs have great demand for the CPU, where some need extra storage. In different resources the costs may differ so the costs of the jobs are also different. So Ant Colony Optimization is co-operative for reproduce the job costs. So we use a resources cost model to solve these problems. Resource cost model give back the burdens of the jobs for the resources. Multi-Objective optimization scheduling method takes two limitations of performance and cost. This problem solves by using Ant colony optimization algorithm. The ant colony optimization algorithm can calculate and modify the value of the result by using two compels performance and cost.

2. RELATED WORK

Ant Colony optimization was capable to solving the job scheduling problems. Were the job scheduling is related to the scheduling performance, response time and the completion time. Several Multi-Objective optimization scheduling also contains the completion time, commercial cost and energy consumption. Ant Colony Optimization permits fast near optimal results to be found. It is useful in industrialized surroundings where computational resources and time are limited. Scheduling method used Ant colony optimization method to solve this problem.

2.1 Scheduling Performance

The best essential features for schedulers is the entire processing time of cloud job. The main target of scheduling performance is scheduling time. Some researcher uses intelligent optimization algorithms to optimize job scheduling. The focus in on decreasing time, response time, completion time related with scheduling performance. The scheduling problems divided into two portions: the variety discovery and advance discovery. To explain addiction of cloud jobs, the scheduling problems can be interpreted into the directed acyclic graph (DAG).

2.2 The Multi-Objective Optimization Scheduling

Multi-objective optimization comprises the completion time, Quality of service (QoS), energy consumption and economic cost. In cloud computing there are many elements to be used for job scheduling. Multi-Objective optimization way has been projected difference metrics like cost, resource utilization. Energy consumption also the major problem for maintenance of cloud datacenters. The main target of multi-objective optimization algorithms are execution time, economic cost, and system performance. The use of a multi objective optimization in job scheduling is minimizing make span and cost in a multi cloud environment. It proposed a multi-objective optimization method to exploit the income of cloud providers.

2.3 Scheduling Method Based On ACO Algorithm

Italian scholar M.Dorigo proposed the Ant Colony Optimization algorithm to resolve the ideal result of combinatorial optimization problem. The optimization scheduling problems comprises several targets of performance and cost. Firstly, this scheduling method focused on scheduling productivity, such as accomplishment time. It is generally emphases on optimizing the total implementation time by ant colony optimization algorithm using guesstimate method. The scheduling method grounded on ant colony optimization method. By using pre-execution time that works set of pheromone threshold to avoid the local optimum solution. Secondly the system performance works not only reduces make span but also achieve the load balancing by decreasing the time. The ant colony optimization algorithm schedules the jobs of cloud customers to virtual machines in cloud computing environment in an effective way. The ACO algorithm solve the over-all optimization problem with ant by evading the extended paths whose pheromones are incorrectly collected by leading ants.

Thirdly the scheduling method consider the cost. For example an ant colony optimization algorithm solves the tricky by using big workflow arrangement and various quality of service factors. Mainly the scheduling methods estimate the quality of the result and the performance and cost.

3. SYSTEM MODEL

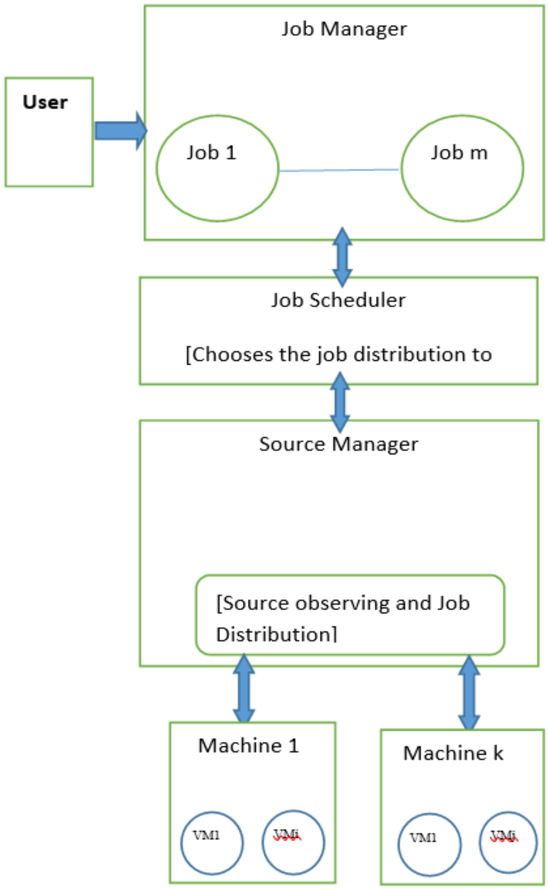

Figure 1: System Model

The scheme structure model is shown in fig. 1; when customer submits a set of cloud jobs, the job chief admits jobs and manage them with job schedulers. After user submit, then submits the information to the scheduler. The job scheduling algorithm exist in, is answerable for determining the job distribution to resources. The job scheduler communicates with the resource chief which is capable to display the valid resources and goes of job allocation from the job scheduler. The resource executive monitors resources and stores the data about CPU utilization. Then the storage information can be used on behalf of job scheduler. By gathering and informing the system's data from the resource executive the job scheduler maintains up-to-date resource information. The universal resource leader collects and updates the data from limited resource managers. The native resources manager observers the time running jobs and then submits that time to global manager. Then universal manager analyses source cost by resource model.

The simple rules for scheduling the cloud jobs are; first collects the data about cloud job and resources from the supply chief. Secondly the resource R_j meets the requirement of jobs $J_{i.}$. The necessities comprise the time limit, cost and the real requirements. At lastly the scheduler assigns the resource R_j to the job $J_{i.}$

4. DEFINITION OF JOB AND RESOURCES

Firstly, assumed that there are k no of jobs

J= $\{J_1, J_2, \dots \dots, J_i, \dots .. j_K\}$ and N number of assets

To the virtual resources. The jobs $J_i = (C_i, M_i, D_i, B_i)$, the two parameters C_i *and* M_i are CPU practise and memory usage. And other two parameters D_i *and* B_i are deadline of the jobs and cheap cost of the users. These limitations arise from the job leader abs are submitted by consumers. Resource cloud is well-defined by the key limitations of its CPU and memory.

Every resource cloud is defined by CPU and memory. That is resource R_j= (C_j, M_j). Where C_j is the CPU utilization and M_j is the memory usage.

According to the above definition we are expected that data submitted by customer is reliable.

5. RESOURCE COST MODEL

Resource cost system reproduce the connection among resource and customer low-priced. In cloud computing, resource and jobs are different to each other. Example; several jobs demand various CPU resources, however others want more loading. Different resources are different price, so the job costs are too dissimilar. So, we consider the alterations in job demands for assets. So, we suggest a resource price model that reflects connection among the user resource cost and cheap cost. Resource cost system distributes the assets price into two parts of CPU and memory.

The cost of CPU is defined as:

$$C_{cost(J)} = C_{base} * C_{ij} * t_{ij} + C_{Trans} \qquad \dots\dots (1)$$

When the source is used by lowermost application C_{base} is the base cost. t_{ij} Is the interval time of the job job T_i goes in resource R_j. C_{Trans} is the rate related with the CPU program.

The cost of memory is defined as:

$$M_{cost(J)} = M_{base} * M_j * t_{ij} + M_{Trans} \qquad \dots\dots\dots (2)$$

M_{base} Is the base price while memory is 1 GB t_{ij} is interval time of the job T_i running in resource R_j . M_{Trans} Is the cost related with the memory program.

Established on above source cost model CPU and memory, the cost function defined as;

$$C_j = \sum_{j=1}^{N} C_{cost}\,(j)$$

$$M_j = \sum_{j=1}^{N} M_{cost}\,(j)$$

6. SCHEDULING OPTIMIZATION MODEL

Scheduling optimization system is created on the source cost mode because in cloud computing, Scheduling is definite by not only scheduling performance but also the user budget rate. In cloud computing system first we assumed that there are K number of jobs and N number of resources are there where jobs J= $\{ J_1, J_2, J_3, \dots \dots J_K\}$ and resource R= $\{R_1, R_2, R_3, \dots R_N\}$. The aim of the scheduling optimization method is to do the ideal result by scheduling the jobs K to resource N. At the similar period it is essential to chance the target and customer budget prices. This is a multi-objective optimization problem which is defined as:

Minimize $\sum_x H(x)$ = P(x), O(x)
O(x) = C(x) +M(x)
O(x) ≤ $\sum_{i=1}^{K} B$i
P(x)≤ $\sum_{i=1}^{K} Di$

Where X is the possible result, and P(x) is the objective functions that refers the make span.

O(x) is the objective function that refers customer budget prices which contain job demands for CPU and memory that equivalent to the rate function in the source rate system. This is a multi-objective optimization problem which is difficult to explain so we propose an Ant Colony Optimization to solve this problem.

7. SCHEDULING BASED ON ANT COLONY OPTIMIZATION

The optimization scheduling is the combinatorial optimization problem that include the performance, Cost, deadline and cheap cost. Due to the combinatorial Optimization problem, it is hard to explain the problem and also hard to get the ideal result. The Ant colony optimization has a Benefit to solve the combinatorial optimization problem. The Ant colony optimization has been used to solve verities of Scheduling problems and to achieve the decent outcomes. Therefore, we can Proposed an Ant Colony algorithm which calculates the quality of result and available feedback. Ant Colony algorithm avoids the local optimal solution.

7.1 ACO Algorithm

The ant colony optimization algorithm (ACO) is a circulated procedure that is used to resolve combinatorial difficulties. By simulating the searching process of ants this algorithm complete the scheduling process. At first each Ants starts from a randomly selected path. Each Ants keeps a memory called pheromone of that path. When the Ant reaches its target, it calculates its path of capability and at that point the ants sets the pheromone on the path permitting to its fitness. At last, the Ants focus on the great suitability path and to achieve the best explanation. At last, it is essential to inform the pheromone and behavior.

7.1.1 Real Behaviors of Ant Colony Algorithm

According to the Ant colony algorithm, first we give the input of the number of jobs, the jobs goal and cheap costs, the number of resources. Secondly, every jobs is allocated an ant. When the jobs j_i is positively assigned to source R_j the jobs j_i will recorded through the table. All the jobs are scheduled completely according to the repeated above steps. This method of allocating jobs to resources copycats the process of ant developing track. This process has the essential connection with pheromone and experimental data. So in order to reach the best result this system was used to achieve behavioral sets:

$$P_k(j_i,R_j) = \begin{cases} \frac{[\tau(J_i,R_j)]^{\delta}\ [\eta(J_i,R_j)]^{\beta}}{\sum_{h\epsilon gk(J_i,R_j)}\ [\tau(J_i,R_j)]^{\delta}\ [\eta(J_i,R_j)]^{\beta}} & R_j\ \epsilon\ gk(J_i,R_j) \\ 0, otherwise \end{cases}$$

……….. (9)

We assumed that $g_k(J_i,R_j)$ is the source set which meets the target and cheap cost of jobs $J_i$in K^{th} iteration.

Here $\tau(J_i,R_j)$ is the pheromone of the job J_i that allotted to the resource R_j in the route $\eta(J_i,R_j)$ is the Heuristic information of the jobs J_i . $\alpha\ and\ \beta$ Are the weight influences of the experiential data and pheromone.

7.1.2 Fitness Function

When an Ant go across all the jobs, at that times Ants procedure a path, that route is the possible solution to explain the problem. A Fitness function is used to calculate the excellence of possible results. The fitness function wants to be based on the optimization problems. According to the scheduling optimization, there are two scheduling aims make span and minimized the prices. The formula of fitness function is:

Fit(x) = $\rho e^{(-P(x))} + \sigma e^{-O(x)}$ ……… (10)

$\rho\ and\ \sigma$ are the load aspect of the performance and rate. $\rho > \sigma\ and\ \rho, \sigma\ \epsilon (0,1)$. Here P(x) is the performance and O(x) is the cost objective purposes, individually.

7.1.3 Informing pheromone

If suitability of a route is high, the pheromone of the route should allow more ants to finds the path.it is needed to inform for each side of the path. The rule of informing pheromone is:

$\tau(J_i, R_j)$ = (1- γ). $\tau(J_i,R_j)$ +$\Delta\tau(J_i,R_j)$ ... ... (11)

Where the pheromone vaporescence factor is γ , and the incremental amount of the pheromone is $\Delta\tau(J_i,R_j)$. The suitability of the route is greater and the incremental amount is better.

$$\Delta\tau(J_i,R_j) = \begin{cases} Q(\rho e^{(-f(x))} + \sigma e^{(-b(x))} & (J_i,R_j)\in path \\ 0, otherwise \end{cases}$$

….. (12)

Where Q is the constant and the value is taken as 100. The performance P(x) and the cost objective function O(x) are lesser and the incremental amount of pheromone is upper. The decent result will be upgrading and by the using of informing the pheromone the poor result will be reduced. The pheromone vaporesence factor is used to prevent the optimal solution.

7.1.4 Algorithm

Input: $J_1, J_2, \ldots J_i, \ldots J_k, R_1, R_2, \ldots R_j, \ldots R_N$,itr_{max}
Output:
Map (J_i, R_j)

I. START
II. Set ants' dispersal among R_j
III. SEE TO
IV. For each ant see to
V. For each J_i see to
VI. Choice next path
VII. Finish For
VIII. Estimate fitness of discrete route by Formula 10;
IX. IF r_j encounters the optimization problem Then
X. Output the map (J_i, R_j);
XI. Inform pheromone along its route by formula 11 and 12;
XII. Finish IF
XIII. Finish For
XIV. Till itr_{max}
XV. FINISH

Where $itr_{max}$is extreme number of repetitions which has been 100.

7.2 Difficulty Analysis of System

Time Difficulty has been separated into two fragments. First one is to discovery the ideal result and the algorithm difficulty is 0(K). Second one is to optimization judgment to meet the price and performance which are constrains. The difficulty is 0(KN). So, the inclusive difficulty algorithm is 0(KN). The sum of jobs, the sum of resources and the sum of ants are constants where the complexity is on space. At the same time optimization scheduling system does not involve active variables. So, this one does not essential additional space and the difficulty of the space is O (1).

8. REPRODUCTION

Two types of reproduction experimentations are there, one is reproduction experimentations and the second one is actual request instance experimentations. By using of Cloud some reproduction experiments were designed to prove the performance of PBACO.

8.1 Experimentation and Constraints Setup

By using cloud experiment generated a data center. On each host were 100 clouds and 10 computer-generated machines. The parameters setup computer-generated machines and the jobs in data center are shown in the *table 1* and *table 2.*

Table 1: The constraint setup of virtual machine in data center

Parameter	Value
CPU computing ability	1860 MIPs, 2660 MIPs
RAM	4096 MB
Band Width	100M/s
Storage	10G

Table 2: The constraint setup of jobs in data center

Parameter	Value
Length (CPU)	[400,1000] MIPs
Folder size	[200,1000] MB
Production size (Memory)	[20,40] MB
The number of jobs	[100,600]

FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Computer Science Research (IJCSR)
Survey Article | Volume 2, Issue 3 | Pages 89-95 | ISSN XXXX-XXXX

The total ants in ant colony (ACO) algorithm and in original ant colony (ACO) algorithm is 10 in the experimentations. Sum of repetitions is 100. We verified and related the performance of 10 sets of dissimilar α, β, ρ limitations. After we designated the greatest set of limitations as the limitations in experiments. The value of ρ has been fixed as 0.01.

8.2 Examination Metrics

There are four performance-evaluation are used in the experiments: First one is makes pan, second one is rate of user, destruction rate of target, and the fourth one is the resource utilization. Make span is the overall time of complete jobs which are used to assess the arranging performance. Rate of customer which are different for the diffident resource method and resources; destruction rate target which is the feedback outcome of Qos to prove its performance.

The price was designed by using source price and the completion of the jobs. 0.17/hour, 0.05/GB/hour. Programme charges for microprocessor (CPU) and memory have been fixed as the target destruction level is too shown as the arranging performance. If the reply time and the achievement time and the job is greater than the dead line then the job is considered to disrupt the target constrains. The deadline destruction rate is designed as:

$$V=\frac{n_d}{K}*100$$

Where n_d is the sum destruction the deadline time in K jobs.

8.3 Make Span

There are 100 to 600 jobs were submitted for 10 times. To get the mean value make span was calculated after the result output the execution time of each jobs.

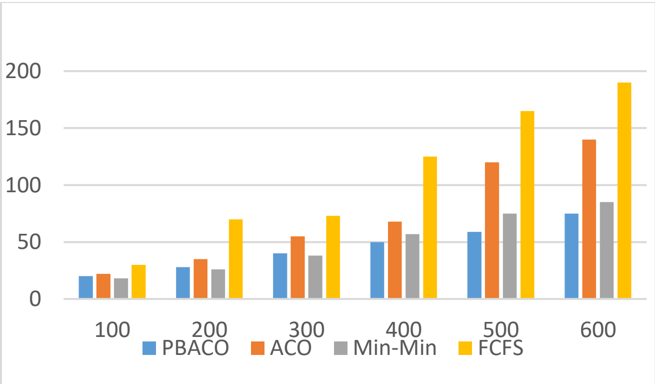

Figure 2: Make span, the arrival rate=10

Make span of four systems were diffident coming rates. First the diffident coming rates alter the make span marks. When the coming time is higher and the reply time is larger thus make span is higher when the onset level is 80 per second. Next the make span is diffident from the diffident methods. In the PBACO the number of jobs are lower but the performance of the PBACO is best. The make span PBACO is related to the Min-Min technique. PBACO method has a great advantage because the number of jobs increased. PBACO method is the best because it increases closely 56.6% comparative to FCFS algorithm. This technique uses to access good quality of solution. It is cooler to discover best result than unique Ant colony (ACO) algorithm.

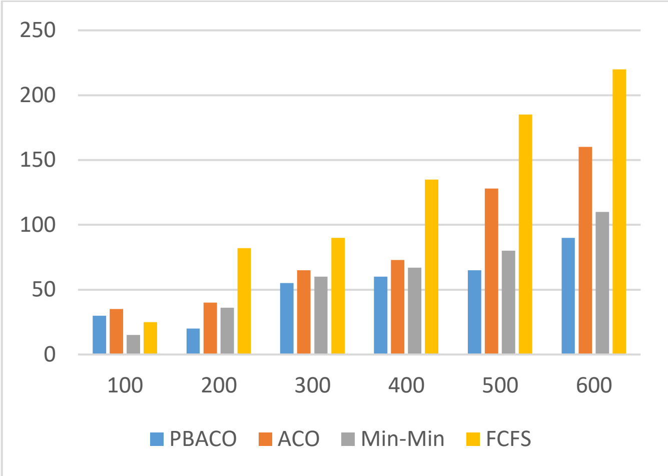

Figure 3: Make span, the arrival rate=80

8.4 Costs

In the second experiment there are 200 to 600 jobs were submitted tat verifies the costs at different deadlines. In this paper PBACO was used for resource cost model and cost constraints where the other three methods were used to considered only the deadline.

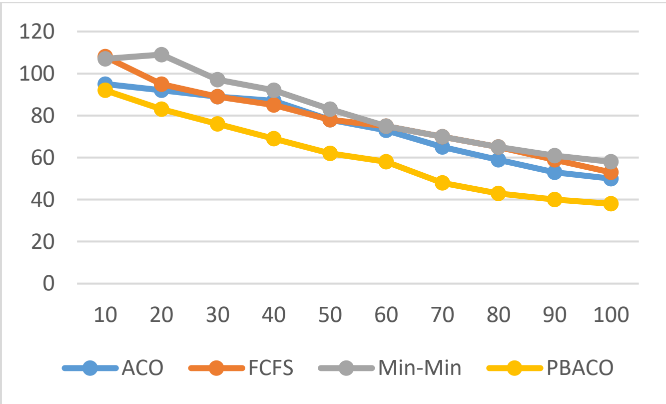

Figure 4: Cost with different deadline, K=200

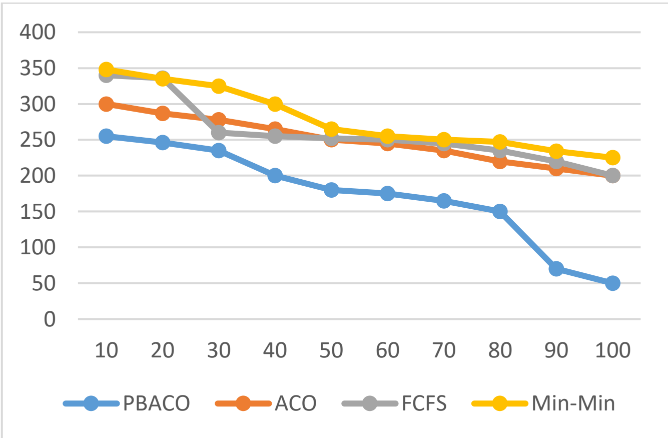

Figure 5: Cost with different deadline, K=600

When the sum of jobs is fewer, the prices of several systems are not very different. Here also PBACO is best. Compared to other method the reducing range is form 7% to 23%. PBACO rises the price limitations comparative to other method. These two figures reproduce the connection between the source price and the targets.

8.5 The Destruction Rate of Target

Through the target destruction rates this experiment verified the scheduling QoS. This experiment selected 200 and 600 job and set some of longer jobs on the target. In the target destruction rates for dissimilar jobs deadlines, the target violation rates have been lesser. Where few jobs have more resource to choice at the same time the target destruction rates are high for all four systems. Still the PBACO is the greatest method of all four systems.

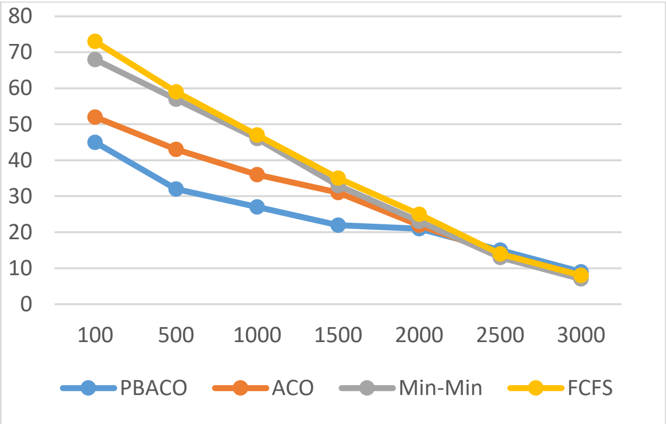

Figure 6: The destruction rate of target, K=200

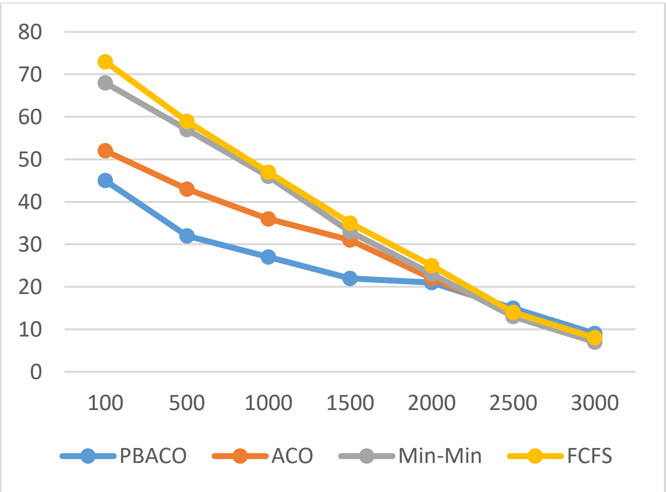

Figure 7: The destruction rate of target, K=600

8.6 The Source Use

Scheduled 100 times. For every planning each resource was used as the resource utilization. Deadline is tight the jobs the performance of source utilization and the load balancing were Shown in the graph. When the jobs plan in several resources. Therefore, the load balancing and the resource utilization are worse. The CPU computing capacity for these incomes is 2660 MIPS and the residual incomes are 1860 MIPS. Here, PBACO is still the greatest system of the other four systems. The dissimilarity for the source Operation of PBACO is fewer than the other three method. PBACO evaluates the performance and adjust the quality of the result.

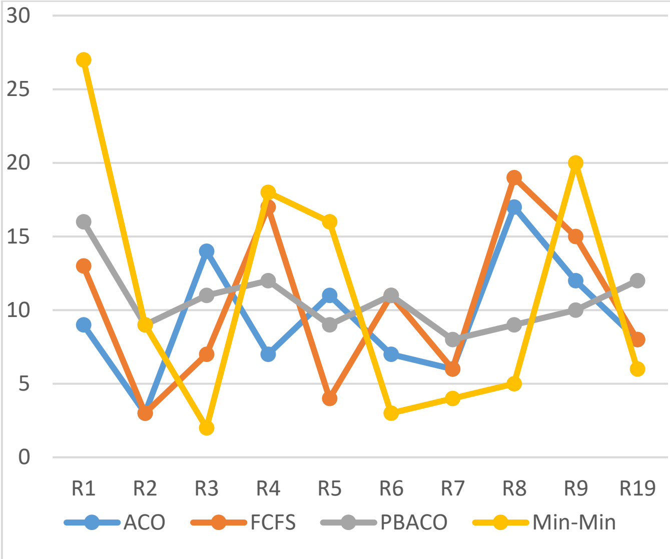

Figure 8: The source operation, the deadline≥100

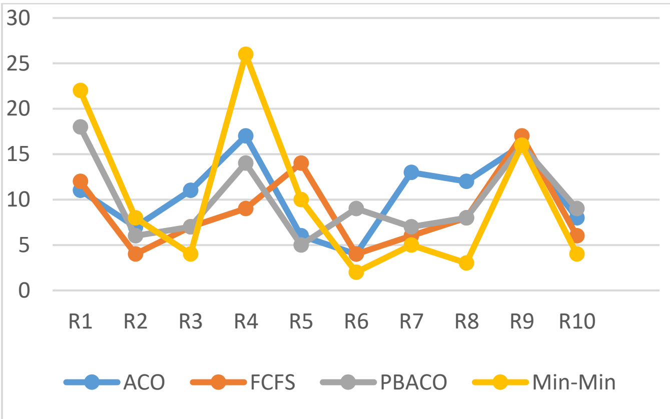

Figure 9: The source operation, the deadline<100

9. CONCLUSION

A multi objective optimization scheduling (MOOS) technique (PBACO) was used established on Ant Colony (ACO) Algorithm. PBACO source price system is used which explain the job request for incomes. This System reproduces the connection among the source rate and jobs prices. Then a multi objective Optimization technique was used which main goal was to optimize the scheduling routine and the Customer prices. At last, an ant colony (ACO) algorithm was used to explain optimization problem. This System used the performance and user cheap limitation function to evaluate the cost and give Feedback. Due to the experimental result, it is clear that the PBACO technique has great benefits in the terms of make span also the PBACO method was almost equivalent to Min-Min algorithm.

International Journal of Computer Science Research (IJCSR)
Survey Article | Volume 2, Issue 3 | Pages 89-95 | ISSN XXXX-XXXX

REFERENCES

[1] Y. P. Dave, A. S. Shelat, D. S. Patel and R. H. Jhaveri, "Various Job Scheduling in Cloud Computing: A Survey," IEEE (ICICES), pp. ISBN NO.978-1-47799-3834-6/14, 2014.

[2] S. Ghanbari and M. Othman, "A Priority based Job Scheduling Algorithm in Cloud Computing," Elsevier (ICASCE), pp. 778-785, 2012.

[3] M.Vijayalakshmi and V. Kumar, "Investigations on Job Scheduling Algorithms in Cloud Computing," IJARCST, vol. 2, pp. 157-161, 2014.

[4] E. WEINTRAUB and Y. COHEN, "Multi Objective Optimization of Cloud Computing Services for Consumers," IJACSA, vol. 8, pp. 139-147, 2017.

[5] L. ZUO, L. SHU1 and C. ZHU, "A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing," IEEE Access, 2015.

[6] S. K. Panda and P. K. Jana, "A Multi-Objective Task Scheduling Algorithm for Heterogeneous Multi-Cloud Environment," IEEE, 2007.

[7] LIYUN ZUO1, I. LEI SHU1 (Member, S. DONG2, C. Z. M. IEEE) and I. TAKAHIRO HARA4 (Senior Member, "A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing," IEEE, vol. 3, p. 13, 2015.

[8] Ewa Figielska, "An Ant Colony Optimization Algorithm for Scheduling Parallel Machines with Sequence-Dependent Setup Costs",IEEE,p.15,2011

[9] Mohan B.C., Baskaran R, "A survey: Ant Colony Optimization based recent research and implementation on several engineering domain", 2012, Vol. 39, s. 4618-4627

[10] Xuan Chen and Jun Chen," Research in the Cloud Calculation Database Based on Improved Ant Colony Algorithm",IJDTA,vol.9,p.71-78,2016

[11] Subhashis Mishra, Debashis Mishra, Dr Madhabananda Das,"A Survey Paper on Swarm Intelligent Techniques and Applications", vol. 8, pp.23-26, 2017

[12] SrinivasanSelvaraj, Jaquline.J"Ant Colony Optimization Algorithm for Scheduling Cloud Task", vol 7(3), pp. 491-494.

[13] Guo. L, Zhao.S., Shen, S., Jiang, C., "Task Scheduling Optimization in cloud computing based on heuristic algorithm", pp. 547-553, 2012.

[14] Mao,J., "Task Scheduling of Parallel Programming system using Ant Colony Optimization" pp. 179-182, 2-010.

[15] Goo G, Ting-Lei H, Shuai G. "Genetic Simulated Annealingaalgorithm for Task Scheduling based on Cloud Computing Environment", p. 60-63, 2010.

[16] S. Shin, Y. Kim, and S. Lee, ''Deadline-guaranteed scheduling algorithm with improved resource utilization for cloud computing,'' in Proc. 12th Annu. IEEE Consum. Commun. Netw. Conf. (CCNC), Jan. 2015, pp. 814–819.

[17] M.A.RodriguezandR.Buyya,''Deadlinebasedresourceprovisioningand scheduling algorithm for scientific workflows on clouds,'' IEEE Trans. Cloud Comput., vol. 2, no. 2, pp. 222–235, Apr./Jun. 2014.

[18] B. Tripathy, S. Dash, and S. K. Padhy, ''Dynamic task scheduling using a directed neural network,'' J. Parallel Distrib. Comput., vol. 75, no. 5, pp. 101–106, Jan. 2015.

[19] F. Ferrandi, P. L. Lanzi, C. Pilato, D. Sciuto, and A. Tumeo, ''Ant colony heuristic for mapping and scheduling tasks and communications on heterogeneous embedded systems,'' IEEE Trans. Compute.-Aided Design Integer. Circuits Syst., vol. 29, no. 6, pp. 911–924, Jun. 2010.

[20] Iosup, A., Ostermann, S., Yigitbasi, M.N., Prodan, R., Fahringer, T. and Epema, D.H.J, "Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing", IEEE Transactions on Parallel and Distributed Systems, Vol. 22, No. 6, 2011, pp. 931-945.

[21] S. He, L. Guo, and Y. Guo, ''Real time elastic cloud management for limited resources,'' in Proc. 4th IEEE Int. Conf. Cloud Comput., Washington, DC, USA, Jul. 2011, pp. 622–629.