Resource Scheduling in Cloud using Bacterial Foraging Optimization Algorithm

Liji Jacob
Department of computer science
Karunya University
Coimbatore

V.Jeyakrishanan
Department of computer science
Karunya University,
Coimbatore

P.Sengottuvelan, Ph. D
Associate Professor, Dept of IT
Bannari Amman Institute of Technology
Sathyamagalam

ABSTRACT
Cloud computing is a form of parallel and distributed computing that consists of interconnected and virtualized computers. Resource scheduling in cloud computing is considered to be a complex task since multiple copies of the same tasks are assigned to different computers. Bacterial foraging optimization algorithm is a global optimization algorithm for the distributed optimization and control. In this paper a bacterial foraging optimization algorithm is used for the scheduling of the resources in the cloud environment. The performance of the algorithm is measured by the use of Cloud Sim. The experimental result shows that the proposed algorithms can reduce the cost, make span and also improve the reliability.

General Terms
Cloud Computing, Scheduling Algorithm

Keywords
BFO, Cloud Computing, Optimization, Scheduling, Virtualization.

1. INTRODUCTION
A cloud computing is a type of network based computing that is evolved from utility computing and grid computing. The cloud computing is developed in such a way that the reliability and the quality of service problems are addressed. The NIST definition for cloud computing is as follows "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction"[4]. In this platform a graphical user interface is being provided to hide the complexity and the details of the underlying infrastructure from the users and the applications. In this computing it makes use of the technology of virtualization which makes it easy to use and manage. This also speed up the information technology operations and increases the utilization of the infrastructure which in turn reduces the cost.

According to NIST, the main essential characteristics of the cloud computing are [4]: 1. On-demand self-service: In this service provider can provide the services based on the demand of the consumer. 2. Broad network access: The various capabilities can be accessed through the network anywhere at any time. 3. Resource pooling: The various services and the resources provided by the resource providers can be pooled together which can be dynamically access by various users. 4. Rapid elasticity: The different capabilities can be provisioned elastically and can be released automatically. 5. Measured services: The services offered by the cloud can be measured.

There are also other characteristics of cloud computing that are agility, cost, virtualization, multitenancy, reliability, performance and maintenance. The main benefits of cloud computing are:

- Reduce runtime and response time: The run time for executing tasks and the response time for obtaining the results is reduced.
- The risks caused by purchasing the physical servers can be minimized by using cloud.
- The initial capital investment is low.
- By using cloud there is no use for the software to be licensed, upgraded or maintained.

Cloud computing types can be classified based on: the types of services and based on the location of cloud hosted. According to NIST, the types of the services offered by the cloud can be classified into three [4]. They are:

1. Software as a service (SaaS): The application software is installed and operated by the cloud providers and the users can access the software from the cloud clients. In this case there is no need for the users to manage the infrastructure and the platform on which the application runs. The SaaS applications are priced based on the monthly or yearly basis for each of the user. Eg: Casengo, Google Apps, Microsoft Office 365.

2. Infrastructure as a service (IaaS): In this type of services the virtual machines, servers, storage, load balancers and network is provided to the cloud users. It offers some additional resources such as virtual machine disk image library file based storage and firewall. Based on the demands of the user the various resources are provided. Eg: Amazon EC2, Azure services platform, ArVM.

3. Platform as a service (PaaS): In this the computing platform is delivered by the cloud providers. The computing platform includes the OS, programming language execution environment, database and web server. By using this developer can run their software services on cloud platform. Eg: ANS elastic beanstalk, Google apps engine, OrangeScape.

According to NIST, the cloud computing type classified based on the location of the cloud hosted are known as deployment models [4]. There are four types in deployment models: 1. Private cloud where the cloud infrastructure is owned and managed by an organization, 2. Public cloud are available to internet user who wants to use based on pay-as-you-go model. 3. Community cloud that shares the properties of both public and private cloud. 4. Hybrid cloud which are coordinated by a broker.

The service that is used for the resource scheduling is the IaaS type of service. In this there is no need for the user to own or manage or operate their own computer instead they can manage some others computer. This means that the virtualization technique is being used here. Virtualization can be implemented in virtual machines which are the abstraction of a physical host machine. Multiple virtual machines can be run on a single physical machine using the virtual machine technology. In IaaS a hypervisor provides virtual hardware resources to the guest OS. The hypervisor monitor the virtual machine execution and the management and configuration capabilities are also provided.

For scheduling the resources in cloud computing there is a need to consider different Quality of service parameters; they are network bandwidth, service completion time, system reliability and cost. These can be used for the measurement of the cloud computing applications. The various algorithms that can be used for the scheduling are the evolutionary algorithms that are based on the biological origin of the species. In the distributed computing the two main groups of job scheduling algorithms are [8]: 1. Batch mode heuristic scheduling algorithm (BMHA) where the jobs are executed as batches eg: FCFS, Min-Min algorithm 2. Online mode heuristic scheduling algorithm where the jobs are executed individually. The online mode is mainly used in cloud environment.

In order to solve the NP complete and NP hard problems heuristic approaches can be used. The various heuristic techniques that can be used are local heuristics, meta-heuristics and hyper-heuristics. The hyper-heuristics can be used to operate at a higher level of abstraction. Meta-heuristic techniques are expensive techniques that require knowledge in problem domain and heuristic technique. Hyper-heuristic technique does not require problem specific knowledge. In order to solve hard computational search problems the hyper-heuristic techniques can be used. The hyper-heuristic techniques can be operated on the search space of heuristics. It can be applied whenever the resources are dynamic in nature [6].

Soft computing is an approach for constructing system which are computationally intelligent and can adapt to the changing environment. The components of the soft computing are fuzzy logic, neural network, machine learning. The evolutionary algorithms are mainly based on the origin of the species. Examples of evolutionary algorithm are Particle Swarm Optimization and Genetic algorithm. PSO is a type of parallel evolutionary computation technique. It is also a heuristic search method which was inspired by the swarming behavior of the biological populations. By using PSO it is able to achieve a good performance.

The rest of the paper is structured as follows: In Section 2, the related works has been presented. In Section3, a description of the resource scheduling in cloud computing is described. Bacterial foraging based hyper-heuristic technique for resource scheduling in cloud is presented in Section 4.Section 5 describes the performance evaluation and results. The comparisons the proposed algorithms with the existing algorithms were done. Conclusion is presented in Section 6.

2. RELATED WORKS

The resource scheduling in cloud computing has become a serious issue that needs to be overcome. This is mainly because of the heterogeneous and dynamic nature of the cloud and due to that it becomes a combinatorial optimization problem. We are using the hyper-heuristic methods for solving the combinatorial optimization problem.

The various heuristic methods such as genetic algorithm, simulated annealing, tabu search, particle swarm optimization, ant colony optimization and bacterial foraging can be used for scheduling of resources in cloud computing.

Genetic algorithm is a search heuristic that mimics the process of natural selection. Elisaveta et al. [7] have developed a genetic algorithm called BASIC to deal with numerous optimization problem and uses real time representation schemes both for real and integer variables. In the first step of BASIC populations of randomly created individuals are initialized. In the next step the offsprings are created. In that a biased selection for reproduction is carried out the by gathering samplings in a sampling pool for crossover and after that the recombination of sampling takes place. Finally the mutation takes place. BASIC uses the generation number as the stopping criterion. If that criterion is met then the obtained best solution is proposed as a problem solution.

The ant colony optimization is a probabilistic technique for solving computational problems which can be reduced for solving computational problems which can be reduced for finding good paths through graphs. Zhang et al. [10] have developed an ant colony based optimization for the support vector machines to attain desired output with an acceptable level of accuracy. The performance of SVM referred to the generalization ability in which the capability of classifying unknown data samples correctly through SVM established by learning from training samples. The generalization ability is influenced by regularization constant and kernel function parameters such as bandwidth of the kernel. The automatic optimization of regularization constant and kernel functions are done here. The central component of ACO algorithm is the pheromone model that includes the state transition rule and updating rule to probabilistically sample the search space. The state transition rule enables the ant to select the parameters by using artificial pheromone trail .In state updating rule the ants are encouraged to produce subsets with fewest classification errors. A meta-heuristic process of artificial ants is implemented for optimizing the parameter values of SVM.

Ghanbari et al. [3] proposed a job scheduling algorithm based on multiple criteria decision. In this the jobs are scheduling based on the priority that is being given to them. The job that needs to be done faster is given the higher priority. The assignment of priority is not only based on the waiting time but also based on certain criteria of the jobs. The multiple decision criteria is known as Analytical Hierarchy process that consists of three levels of priorities. One is the objective level that constitutes the scheduling techniques then the attribute level containing the resources and the last is the alternative level containing the various jobs. Various issues such as consistency, complexity and the finish time of the jobs are needed to be handled while assigning the priority. These issues are rectified to an extend using this algorithm.

Xu et al. [9] considers the main characteristics of the cloud computing such as virtualization and commercialization for scheduling of the various jobs and uses a Berger model. In this model a dual fairness constraint is established. Using the first constraint jobs can be classified based on the quality of service preference and with the second constraint the fairness of resource allocation is judged. By using this algorithm user tasks can be executed effectively and with better fairness.

Thus by using all this algorithms the various QoS parameters the requirements of the user. If the algorithm is able to satisfy the needs of the user then that algorithm is considered to be the best in that case. But in all of the above cases the QoS parameters such as cost, reliability and makespan are not considered together. Therefore in this paper we have considered all of these parameters in the cloud environment and compared with the existing algorithms in cloud environment.

3. CLOUD SCHEDULING MODEL

Cloud scheduling is the core of the cloud resource management. This cloud scheduling implies that there are various jobs which can be mapped into the available resources. This means that different administrative domains are searched to use resources from the cloud infrastructure to can be achieved. The algorithms are selected mainly based on satisfy the requirements of the user. In cloud computing environment the resource scheduling is considered to be a complicated task because there are computers with varying capacities [8]. Cloud computing is a type of distributed system and the main advantage of the scheduling algorithm is to obtain a high performance and high throughput.

Based on the simple classification of the scheduling algorithm the job scheduling algorithms can be classified as batch mode heuristic scheduling algorithm where the jobs can be put into the queue as they arrive eg:FCFS,RR and online mode heuristic algorithms where the jobs are processed immediately as they arrive. In the cloud scheduling online ode is mainly used.

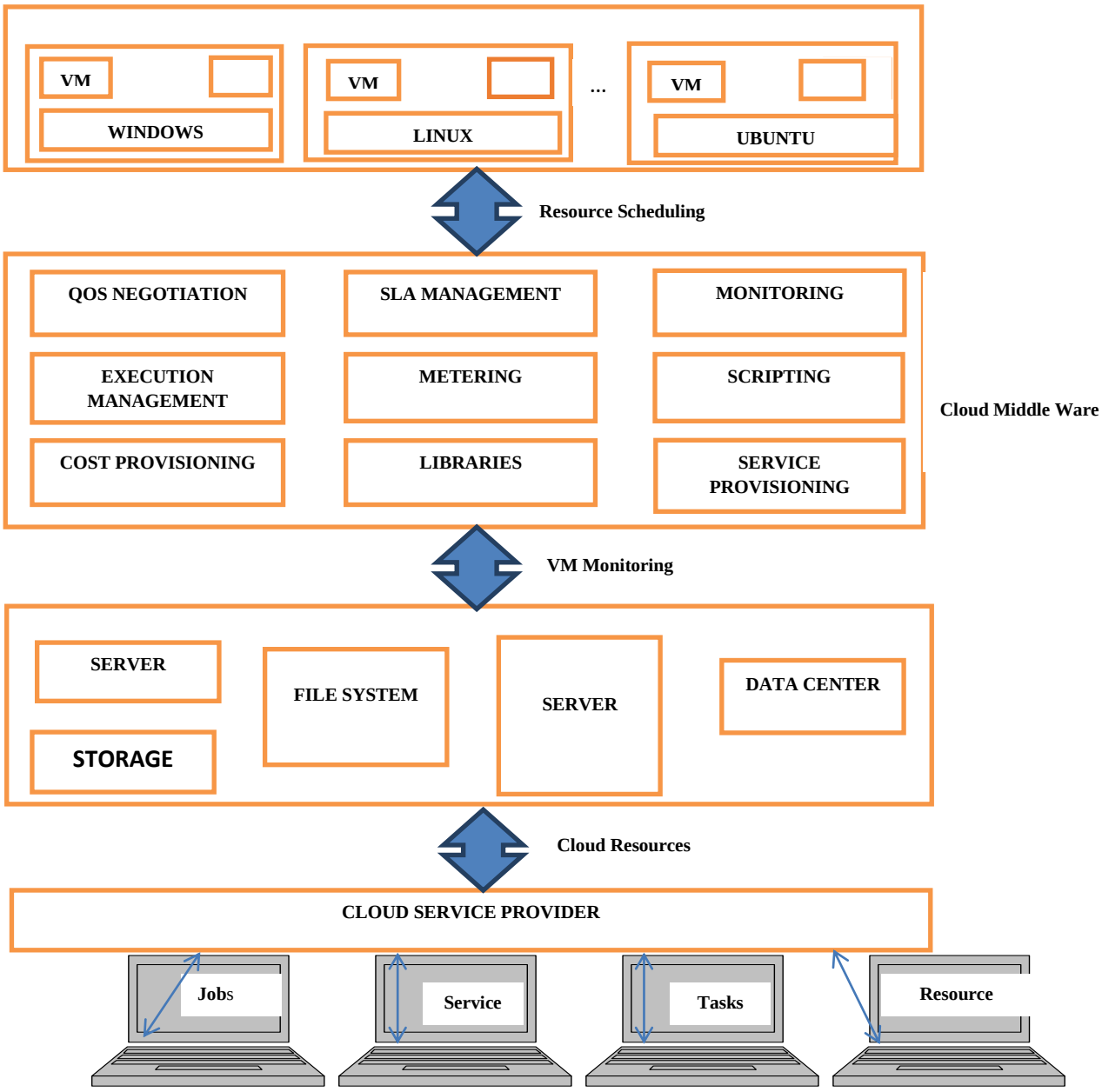

Figure 1:Resource scheduling in cloud

The figure 1 shows the resource scheduling in the cloud computing. The model executes the requests from the user as follows:

- A first the user submits a job to a job scheduler. If there are no free resources then the submitted job will be idle and is put in a queue.
- The cloud scheduler then examines the queue and determines which all jobs are there in the queue without resources.
- Then the cloud scheduler starts the virtual machines, these virtual machines will advertise themselves to the job scheduler.
- The job scheduler sees these virtual machines and starts running jobs on them.

Once all of the jobs are done, cloud scheduler shuts down all the virtual machines that are taking part.

3.1 Problem Statement

The process of scheduling the various tasks to the corresponding resource is a combinatorial problem. The main aim of the Cloud Scheduler is to schedule the resources efficiently. Dynamic allocation of the tasks to the resources is done in cloud and is a very complicated process due to the assignment of multiple copies of the same tasks to different resources. Cloud resources are heterogeneous and dynamic in nature therefore the scheduling in cloud is a NP-hard problem.

3.2 Objective Function

In cloud scheduling the main aim of the user is the minimum cost for cloud applications and also the user aims the application to be reliable in all situations. The main aim of the cloud service provider is to minimize the makespan ie, to increase the productivity of the cloud systems.

Cost =min(C ($r_i j_k$)) for 1≤ i≤ n, 1≤ k ≤ m
(1)

makespan =min (F_{ji}) for $j_i \in$ J
(2)

Fitness Function = αcost +β makespan +γreliabilty (3)

where $0 \le \alpha \le 1$, $0 \le \beta \le 1$ and $0 \le \gamma \le 1$ are weights to prioritize the components of the fitness function.

Cost C($r_i j_k$) is the cost of the job j_k which executes on resource r_i and makespan is the finishing time of the job.

4. THE PROPOSED ALGORITHM

In this a pseudo code of the bacterial foraging based hyper-heuristics for resource scheduling in cloud environment is presented. In this the partial solution is represented by bacterium and the movements of the bacterium as heuristics [1, 6]. The optimization in bacterial foraging algorithm consists of process like chemo taxis, swarming, reproduction, elimination and dispersal. The main objective of the animal that searches the food is in such a way that the energy intake per time (E/T) is maximized.

Chemo taxis: In this process the movement of the E.Coli bacteria through swimming and tumbling through flagella takes place. If the bacterium is moving in the same direction for a period of time then it is swimming and if it is moving in a random direction then it is tumbling.

Suppose that α^p(q,r,t) is the p^{th} bacterium at q^{th} chemotactic r^{th} reproductive and t^{th} elimination and dispersal step.S(p) is the size of the step for tumble then the computation chemo taxis of the bacterium is represented as:

α^p(q+1, r, t) = α^p(q, r, t) + S (p)
(4)

Swarming: The group of bacteria can move towards the nutrients but it can either be attracted or rejected. The bacterium will attract other bacteria in times of stress.
Reproduction: After finishing of the chemo taxis step, the reproduction step is done. In this step the fitness value of the bacterium is calculated and it will be sorted on an ascending order. For maintaining the size of the swarm as constant the less healthier bacteria will die and the more healthier bacteria will then splits asexually into two.

Elimination and dispersal: In order to avoid local optima the elimination dispersal process is done.

4.1. Psedocode of BFOA

[Step 1] Initialization of the parameters: d, N, N_c,N_s,N_{re}, N_{ed}, P_{ed},S (i).

d: dimension of the search space.

N: the number of bacteria in the population.

N_c: chemotaxis step.

N_s: swim length.

N_{re}: the number of reproduction steps.

N_{ed}: the number of elimination dispersal events.

P_{ed}: elimination-dispersal with probability.

S (i): the size of the step taken in the random direction.

[Step 2] Elimination-dispersal loop: t=t+1;

[Step 3] Reproduction loop: r=r+1;

[Step 4] Chemo taxis loop: q=q+1;

{1} For i=1, 2, 3….N, take a chemo tactic step for bacterium p as follows.

{2} Choose the heuristic.

Compute fitness function J (p, q, r, t).

Let J(p,q,r,t)= J(p,q,r,t)+J_{cc}(α^P(q,r,t),L(q,r,t)).

{3} Let J_{last} =J (p, q, r, t).

{4} Tumble: generate a random vector Δ (i)∈ R^n with each Δ_m(p), m=1, 2, 3……d.

{5} Move: let α^P(q+1,r,t) = α^P (q,r,t)+ S(P)(Δ(P)/ $\sqrt{\Delta^T(P)\Delta(P)}$).

{6} Compute J(p,q,r,t) and J(p,q,r,t)+ J_{cc}($\propto^P$(q,r,t),P(q,r,t)).

{7} Swim

i. Let m=0.

ii. While m<N_s

Let m=m+1.

If J(p,q+1,r,t)<J_{last} ,let J_{last} =J(p,q+1,k,l) and α^P (q+1,r,t)=α^P(q,r,t)+ S(P)(Δ(P)/

$\sqrt{\Delta^T(P)\Delta(P)}$ and use this α^P (q+1,r,t) to compute the new J(p,q+1,r,t) as in {6}.

Else let m=N_s.

This is the end of the while statement

{8} Go to next bacteria (p+1) if p≠N.

[Step 5] If J<N_c, go to step 3.Continue chemotaxis step since the life of bacteria is not over.

[Step 6] Reproduction: For the given r and t and for each p=1, 2, 3….N.

Let $J^i_{healt\,h}=\sum_{q=1}^{N_{C+1}} J(p,q,r,t)$.

Sort the bacteria based on the health of each.

[Step 7] if r<N_{re}, go to step 2.

[Step 8] Elimination-Dispersal: For i=1, 2, 3…..N with probability P_{ed},eliminate and disperse each bacterium.

[Step 9] If t<N_{ed}, then go to step 1; otherwise end.

5. PERFORMANCE EVALUATION AND DISCUSSION

The cloud simulation provides the functions of modeling and simulation of the data centers, which are virtualized and cloud based. The cloudsim toolkit has been used for simulation due to the following reasons [5]:

- The cloud based application can be done in less time and effort.
- In heterogeneous cloud environment the application services can be modeled and tested with less effort.
- Large scaling computing environment supported.
- A self contained platform is present.
- A virtualized engine is present in cloudsim.
- A high range of flexibility to switch between space shared and time shared allocation of processing cores.

5.1 Performance evaluation criteria

For evaluating the performance of the bacterial foraging based hyper-heuristic in cloud computing three matrices are considered namely makespan, cost and reliability. The reliability is the likelihood of the failure based on actual measurements, the makespan is the total execution time and the cost indicates the cost per unit resources. The makespan is measured in seconds and the cost is measured in cloud dollars(C$).The reliability is measured based on the cost, bandwidth and complexity. Table 1 shows the characteristics of the resources and cloudlets that are being used.

Table 1: Scheduling parameters and their values.

Parameters	Values
Number of virtual machines	100
Number of tasks	1000
Bandwidth	700-1500 B/S
Cost per job	2.456$
Number of PEs per machine	1-4
Number of machines per resource	1
PE ratings	200-450 MIPS

5.2 Results

To validate the algorithm, 1000 jobs and 100 resources are considered. An average of fifty runs are used to guarantee statistical correctness. To evaluate the performance of the proposed bacterial foraging based hyper-heuristic for resource scheduling in cloud computing various parameters are taken and a number of graphs are plotted based on the parameters of each algorithm. Figure 2 and figure 3 shows the cost comparison and makespan comparison respectively.

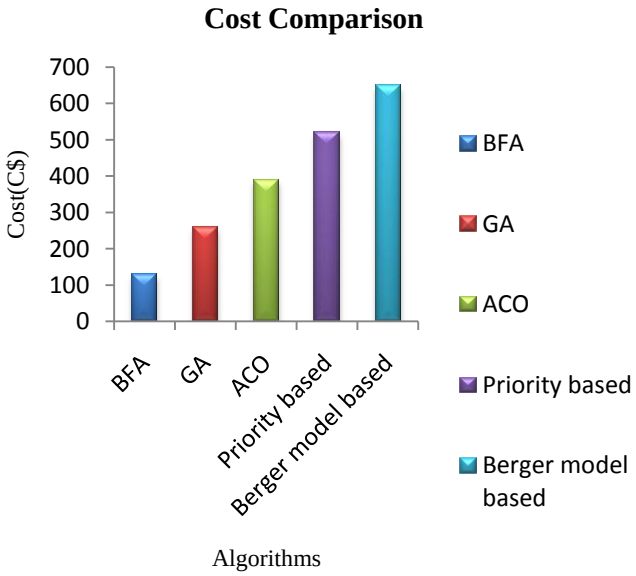

Figure 2.Cost comparison for different algorithms

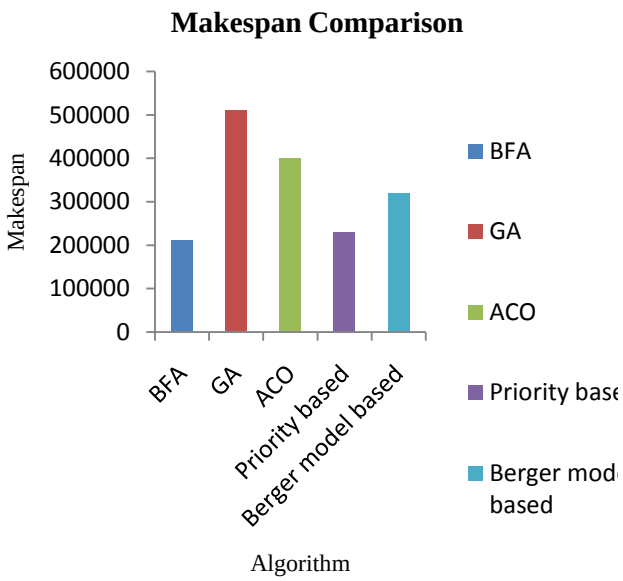

Figure 3: Makespan comparison for different algorithms

Figure 4 and figure 5 shows the effect of the resources on various parameters. From all the comparisons made we know that Bacterial foraging is satisfying the most of the QoS parameters. The performance of the BFO based hyper-heuristic algorithm is compared with GA, ACO, Priority based scheduling and Berger based model.

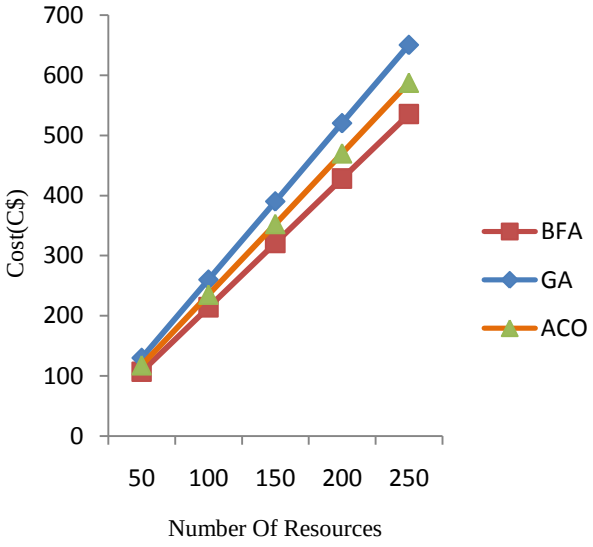

Figure 4: Effect of the number of resources on cost.

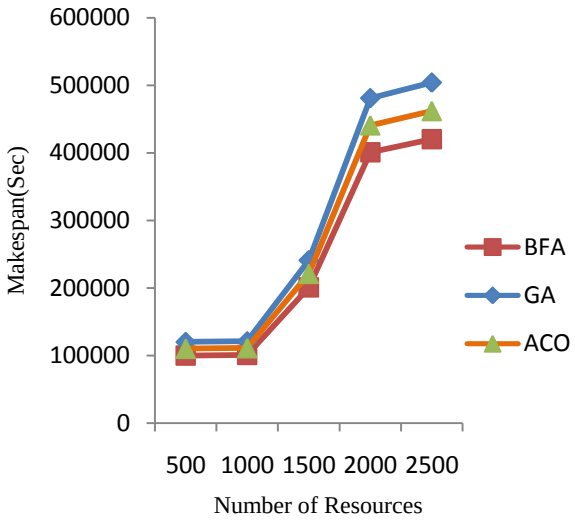

Figure 5: Effect of the number of resources on makespan.

6. CONCLUSION

In the cloud environment multiple copies of the same tasks are assigned to the various computers and therefore the dynamic allocation of resources is a complex task. Therefore there is the need for the scheduling algorithms that will allocate the resources to the tasks automatically. In this paper, a hyper-heuristic based scheduling algorithm is used in the cloud environment. The hyper-heuristic techniques are used so that the best mapping of the resources is done. We have compared the proposed algorithm with the existing algorithm. The proposed algorithm is minimizing the cost and also it minimizes the makespan of the computing resources.

7. ACKNOWLEDGMENTS

We thank Almighty for giving this wonderful opportunity to complete this work successfully.

8. REFERENCES

[1] B.Bhushan., M.Singh., 2011.Adaptive control of DC motor using Bacterial Foraging slgorithm. Applied Soft Computing 11,4913-4920.

[2] E.G.Shopova., N.G.Vaklieva-Bancheva.2006.BASIC –A genetic algorithm for engineering problems solution.Computerand Chemical Engineering 30,1293-1309.

[3] G.Shamsollah., M.Othman.2012.A Priority based Job Scheduling Algorithm in Cloud Computing. Procedia Engineering 50,778-785.

[4] P.Mell., T.Grance.The NIST definition of cloud computing.Special Publication 800-145.

[5] R.N.Calheiros., R.Ranjan., A. Beloglazov., C.A.F. De Rose., R. Buyya.2011.CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software-Practice and Experience .23-50.

[6] Rajini., I.Chana.,2013.Bacterial foraging based hyper-heuristic for resource scheduling in grid computing", Future Generation Computer Systems 29.751-762.

[7] S.G.Elisaveta., V.G.Natasha.,2006.BASIC-A genetic algorithm for engineering problems solution.Computers and Chemical Engineering 30.1293-1309.

[8] V.Vignesh., KS.K.Sendhil., N.Jaisankar., June 2013.Resource management and scheduling in cloud environment. International Journal of Scientific and Research Publications. Volume 3.Issue 6.1-6.

[9] X.Baomin., Z.Chunyan., H.Enzhao., H.Bin.,2011.Job scheduling algorithm based on Berger model in cloud environment.Advances in Engineering Software 42.419-425.

[10] Z. XiaLi., C.XueFeng., H.ZhengJia.2010.An ACO-based algorithm for parameter optimization of support vector machines. Expert Systems with Applications 37.6618-6628.