# An Enhanced Job Scheduling in Cloud Environment using Improved Metaheuristic Approach

Kowsigan. M.
Department of Information Technology
Sri Krishna College Of Technology
Coimbatore-641042, India

Seenivasan. P
Department of Information Technology
Sri Krishna College Of Technology
Coimbatore-641042, India

Rajkumar S.
Department of Information Technology
Sri Krishna College Of Technology
Coimbatore-641042, India

Vikram Kumar C.
Department of Information Technology
Sri Krishna College Of Technology
Coimbatore-641042, India

***Abstract*: Cloud computing is a type of internet based computing which stores and access the information through the internet rather than using our own computer hard drive, where cloud is a metaphor for the internet. Job scheduling is a process of allocating available resources to various tasks by an efficient cloud scheduler and it is one of the major challenging part in the cloud-computing paradigm.The main problem that arises in job scheduling is allocation of resources, complexity in scheduling algorithm and waiting time for each job. To resolve these issues in optimal way we are going for an improved metaheuristic approach called memethic algorithm, which is the combination of genetic algorithm and a local search algorithm. In this research, we are going to produce the comparatively best optimal solution by means of implementing the improved memethic algorithm.**

***Keywords— Memethic algorithm, mutation, crossover, makespan, cloud sheduling***

## I. INTRODUCTION:

Cloud computing is an emerging technology that enables the user to share the resources in the cloud and process the task based on the cloud services provided by the cloud service provider. It enables the users to access the shared pool of resources, which can be rapidly provisioned and released with minimal management effort. It enables the user to store and process the data in both private or public manner and thus the user can access the data anywhere across the world. In cloud, computing Job scheduling plays a vital role. Scheduling is the process of allocating available system resources to many different jobs. An efficient job-scheduling algorithm was proposed in Cloud environment using Auto Associative memory network [3]. An Improved job-scheduling algorithm was implemented using soft computing techniques in cloud environment [4]. The system priories the jobs and thus the completion time of the job is reduced by allocating the resources efficiently among the jobs. The job scheduling processes performed using the job schedulers. In current scenario, many algorithms have been used for job scheduling but they are concentrating only on the allocation of resources to the jobs. However, in those proposals, there exists a saunter in optimal solution i.e. optimised resource utilization. In the proposed system, memethic algorithm is used along with the hill climbing algorithm to allocate the resources to the jobs to obtain the Optimized solution and reduced makespan. In the proposed approach operations of the memethic algorithm such as crossover and mutation are fine tuned to produce the near optimal solution. Here, the allocation of the jobs depends on the efficiency of the resources i.e. higher efficient machine will allocated with more number of jobs.

## II. LITERATUREREVIEW:

In cloud, computing the resource utilization and efficiency of the machine becomes a key factor, thus the memethic algorithm is used here to schedule the process and make use of the resources in a proper manner. Resource scheduling is a major process for cloud processing such as Infrastructure as a Service cloud (Iaas), for the efficient use of algorithms, the optimised way of scheduling algorithm is used for the optimization or sub-optimization problems in cloud. Here they majorly concentrated in the allocation and resource utilization of the resources.in this paper they use parallel genetic algorithm instead of traditional genetic algorithm for the faster process [1]. Here Multi Queue Scheduling (MQS) algorithm used to reduces the cost of reservation and the cost on-demand plans using the global scheduler. In this system the sharing of resources in the cloud, computing is done using the concept of clustering the jobs based on burst time [2]. This paper explains about the scheduling of tasks in the cloud. The primary aim of cloud computing is to provide efficient access to remote and the geographically distributed resources. In the traditional system the Generalized Priority algorithm was used for efficient execution of task and compared with FCFS and Round Robin Scheduling [8]. Improved Ant colony algorithm was used produce the near optimal solution in grid environment [10]. A biometric template has been proposed based on the scheduling on biometric attributes and characteristics [9]. Finger print and speech authentication had been efficiently scheduled using SVM classifier [11]. Here the memethic algorithm is used in Reference point based optimization, offers tools for the effective treatment of preference based multi-objective optimization problems [5]. A multi-objective memethic algorithm is used for obtaining the optimal solution in PPI (proteinprotein interaction) network alignment. Here the memethic algorithm is used for the job scheduling in the cloud processing **[**6**]**. Resource and Deadline-Aware Job Scheduling is done in Dynamic Hadoop Clusters. This proposal is helpful forperformance analysis of wireless networks [7]. This Job scheduling algorithm is used for improving productive and environmental performances in a job shop system. In this proposal the productivity and the environmental performance in enhanced by scheduling of jobs [8].

## III. METAHEURISTIC ALGORITHM:

Efficiency of the cloud computing scheduling algorithm depends on the managing the process efficiently and increasing the performance of the server and alsothe resources for that process. As we have discussed in the previous algorithm for the scheduling process we are facing many problems with that. In order to increase the efficiency, we should minimise those problems in all possible ways. In this paper, we propose a scheduling algorithm to schedule the jobs in an efficient manner for improving the resource utilization. The main goal of our proposal is to improve the utilization of servers and efficient allocation of the jobs. In this paper, we proposed an enhanced metaheuristic algorithm called improved memethic algorithm. The metaheuristic approach is an combination of the evolutionary algorithms and produces an accurate search within the problem space. The memethic algorithm is the combination of genetic algorithm and a local search algorithm. The local search algorithm named hill climbing algorithm is used along with the genetic algorithm to produce the near optimal solution. The objective of this paper is achieved by the following evaluating the following steps.

- To process the job having higher priority.
- Improve the resource utilization.
- Minimizes the completion time (makespan) of Map Reduce jobs
- Minimizing the waiting time
- Minimizing the switching time

## IV. CROSSOVER AND MUTATION

### *A. CROSSOVER:*

Crossover and Mutation enables to produce many possible solutions for the collection of random variables. Here the possible values represent the makespan of each and every scheduling iteration of the resources.

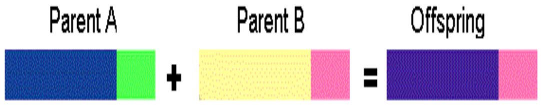

Fig. 1. Croossover and Mutation

11001011+11011111 = 11001111

Two Point crossover - Two point crossover are selected, binary string from beginning of chromosome to the first crossover point is copied from one parent, the part from the first to the second crossover point is copied from the second parent and the rest is copied from the first parent. Here the processing time of the resources are calculated and it swaps the jobs across the resources to identify the processing time of the resources and jobs are swapped across the resources to attain the better results.

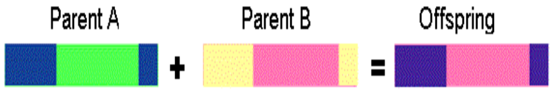

Fig. 2. Two Point Crossover

**11**0010**11** + 11**0111**11 = **11011111**

Uniform crossover - Bits are randomly copied from the first or from the second parent. Here the jobs are randomly swapped across the resources and processing time of the jobs are calculated.

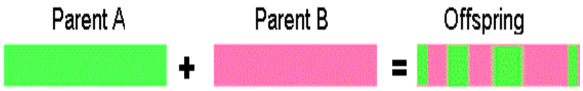

Fig. 3. Uniform Crossover

1**10**010**11** + **1**10**111**01 = 11011111

Arithmetic crossover - Some arithmetic operation is performed to make a new offspring. The jobs with a set are combined by using the arithmetic operation and the combined jobs are swapped across the resources to identify the processing time.

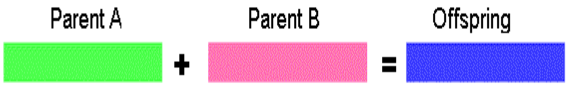

Fig. 4. Arithmetic Crossover

11001011 + 11011111 = 11001001 (AND)

### *B. Mutation:*

Bit Reversing - Selected bits are reversed. Some of the jobs within a set are selected and they are reversed. In this sense reversing means changing the order of the jobs within a particular set or out of the set. Finally, the reversed jobs sets are swapped across the resources to identify the processing time of the job in the cloud environment using this mutation operation.

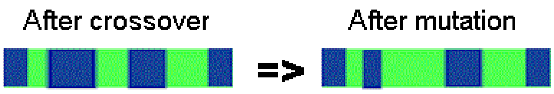

Fig. 4. Bit Reversing (Mutation)

11001001 => 10001001

## V. RESULTS AND DISCUSSION:

STEP 1:

In job scheduling if there is no dependency for the jobs as well as the resources then we consider the switching time because it is more reliable and flexible. If there are any dependencies among the jobs or resources, then directly jump to step 2. For example, in the below figure there are dependency among the resources and jobs also. So there can be chance of deadlock and critical sections. So to avoid this situation we move to step2.

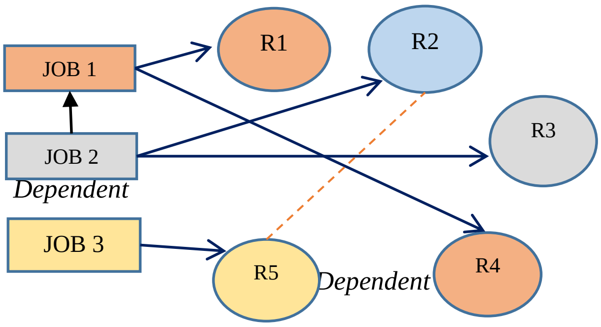

Fig. 5. Task and Resource Dependency

STEP 2:

In the above mentioned the server does not assign any priority to the tasks. To check which job needs to be processed first we take priority as a parameter. For example, just take maximum priority as 1 and minimum priority as 5 then assign the jobs in the priority manner and this helps in the improvement of server performance and also the resource utilisation. Here we take the Backfilling technique to improve the process of resource utilization.

- *TASK 1(2,5,1)*
- *TASK4(2,5,2)*
- *TASK3(2,5,3)*
- *TASK2(6,10,5)*

TASK I (X, Y, Z)

I=Job number
X=Resource required
Y=Time required for the completion of job
Z=Priority for the process.

If there is any dependency among the resources or tasks used, then the higher priority always assigned to the independent job and the dependent job gives the second priority. Thus, the deadlock situation will be reduced to minimum.

STEP 3:

After assigning, the priority to the jobs one question arises that which job is assigned to which machine. Therefore, to know which job is assigned to which machine here we took the average time of execution time of all the jobs. In addition, the identification of the processing time of the machine is evaluated. Then the jobs or tasks are allocated to the respective machine or server. In the below example, we are having three jobs i.e.: Job1, Job2, Job3. Here the average time of the processor is 20 sec and thus all the jobs must complete their jobs in 20 sec. In this regard, the processor which takes higher completion time should be swapped among the processor to balance the load on the cloud environment, which has high processing time, repeat the procedure until all the jobs are assigned to the system or server. Finally, the resources start executing their process in parallel.

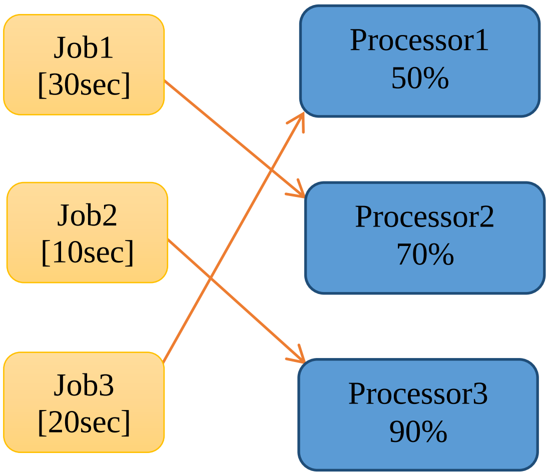

Fig. 6. Timing Specification of the Task and % of Resource utilization

```
Procedure: Improved Memethic Algorithm
Initialize: Generate an initial population;
While Stopping conditions are not satisfied do
        Evaluate all individuals in the population.
Evolve a new population using search operators.
Select the subset of individuals, that should undergo the
individual improvement procedure.
**states are randomly sampled with a specific probability
according to their fitness value**  //Selection
*selected individuals are randomly shuffled or combined at
a specific point**
//Crossover
**child individuals may flip some of their bits (each one
with independent probability)**
//Mutation
for each individual do
```

```
Perform individual learning using meme(s) with frequency
or probability.
endfor
endwhile
```

In the above algorithm, the initial step is to represent the jobs and resources which are all used for processing in the cloud environment. The initial makespan of the resources are considered and it is taken as benchmark for our scheduling process. The objective is to reduce the makespan and to minimise the makespan when comparing with the initial makespan. The fitness function of the environment is constructed and the makespan of the resources are evaluated by using the fitness function. After the evaluation the best solution of the fitness function is displayed. Several iterations are performed to minimize the makespan of the resources by using the major four operations such as crossover, mutation, elitism, reproduction. In these four operations crossover and mutation plays a vital role. Repeat the iteration by using these four operation until no further improvement can be made on the solution of the fitness function.

Our proposed approach results have been compared with other traditional approaches such as Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Genetic Algorithm (GA). The proposed enhanced memethic algorithm performs well and produces better results when compared to the traditional approaches as mentioned above. The comparison has been performed in three dimensions as number of occurrence of jobs, makespan of the resources and resource utilization. Our proposed algorithm works well and produces the near optimal solution for the entire dimension as mentioned above. The problem with ant colony algorithm was it concentrates only on the resource utilization in a distributed environment while scheduling the jobs. In particle swarm optimization the competency of the server changes dynamically which leads to the poor performance evaluation the scheduled jobs in the cloud environment. The improved metaheuristic algorithm in the proposed approach concentrates on the makespan of the resources, resource utilization and occurrence of the jobs in the cloud environment. The makespan is minimized even number of jobs are increased and resources are utilized properly in the cloud environment by using the proposed algorithm.

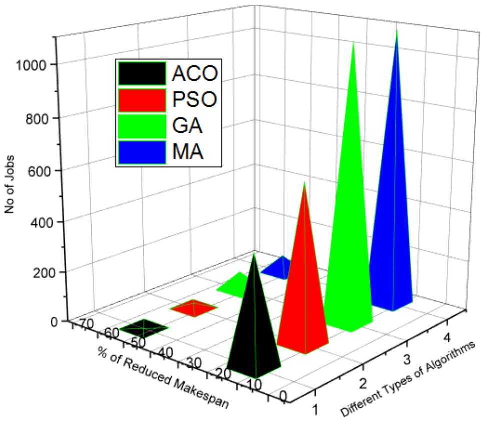

Fig. 7. Comparison Graph

The percentage of reduced makespan is high when comparing to ant colony algorithm, particle swarm optimization and genetic algorithm, More than 70 percent the makespan of the cloud environment is reduced in the propose approach.

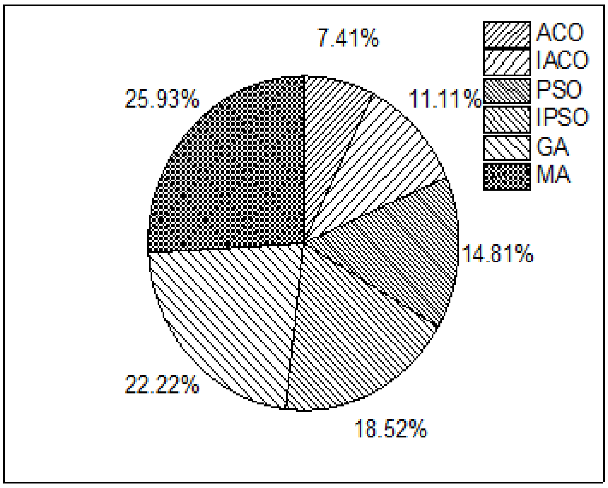

Fig. 8. Comparison Pie chart

The above pie chart shows the comparison among the Ant colony optimization, improved ant colony optimization, particle swarm optimization, improved particle swarm optimization, Gemetic algorithm and memethic algorithm.

## VI. CONCLUSION AND FUTUREWORK:

In this paper, we proposed an improved metaheuristic algorithm along with the hill climbing algorithm for the process of optimised job scheduling in the cloud environment. By using the proposed algorithm resources are allocated in optimised way, thus the wastage of resources in allocation is reduced. In memethic algorithm, the swapping process takes place between the jobs to obtain the optimised solution. Crossover and mutation process is also takes place in job scheduling to obtain the reduced makespan. The hill climbing algorithm is merged with the metaheuristic algorithm to enhance the search mechanism of the problem space. In our current scenario, scheduling of jobs and usage of resources to held in optimised way is difficult, but our work will help to obtain the near optimal solution by reducing the percentage of the makespan more than 70% of optimised solution and resources are allocated effectively and avoids the wastage of the resources. In this concept, we used job-scheduling process for reducing the makespan and to obtain the optimised solution. In future, we work to develop our work and the ultimate goal of this work is to obtain the reduced makespan of minimum of 90% and to attain the best resource utilization.

## VII. REFERENCE:

[1] Zhongni Zheng, Rui Wang, Hai Zhong, Xueijie Zhang,"Improved Job scheduling and resource allocation", Computer Research and Development (ICCRD), 2011 3rd International conference on IEEE march 2011.

[2] M. Kowsigan, P. Balasubramanie, "An Improved Job Scheduling in Cloud Environment using Auto-Associative memory Network", Asian Journal of Research in Social Science & Humanities, Vol. 6, No 12, pp.390-410, Dec 2016.

[3] Saloni Jain, "Enhanced job scheduling and resource utilization" International Journal of Computer Trends and Technology (IJCTT), 2014 march.

[4] M. Kowsigan ,Dr. P. Balasubramanie "Scheduling of Jobs in Cloud Environment Using Soft Computing Techniques Computing Techniques" International Journal of Applied Engineering Research, ISSN 0973-4562 Vol. 10 No.38, 2015

[5] Hamiez, "Optimised work on job scheduling and resource allocation" on Computer and Operation Research Vol. 43, pp.318–327, March 2014,

[6] A JameerBasha "Performance Analysis of wireless OCDMA System using PC OOC, EPC Codes", Asian Journal of Information Technology, Vol. 15, Issue 12, pp.2087-2093, 2016.

[7] Jesus Alejandro Hernandez Mejia, Oliver Schutze,Adrina Lara, Kalyanmoy Deb, "A memetic algorithm for reference point based multi-objective optimization"Journal on Engineering Optimaization", Aug 2016.

[8] Dazhao Cheng, Jia Rao, Changjun Jiang, Xiaobo Zhou, "Resource and Deadline-Aware Job Scheduling in Dynamic Hadoop Clusters", IEEE International Symposium Parallel and Distributed Processing, May 2015.

[9] Poongodi. P, Betty, P, "A study on Biometric Template protection Techniques", International Journal of Engineering Trends and Technology, 2014.

[10] J. Rajeshkumar, M. Kowsigan, "Efficient Scheduling in Computational Grid with an improved Ant Colony Algorithm", International Journal of Computer Science and Technology, Vol. 2, pp.317-321.

[11] A Jameer Basha, V Palanisamy, T Purusothaman, "Multimodal Personal Authentication with Fingerprint, Speech and Teeth Traits using SVM Classifier", European Journal of Scientific Research, Vol 76, No 3, pp.463-473.