International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

# Scheduling For Resource Optimisation in Cloud Computing: Genetic Algorithm Approach

S. R. Mishra[1], M. Mahita[2], K.. D. Naik[3]
[1-2]UG Scholar, [3]HOD, CSE Department, Parala Maharaja Engineering College, BPUT, Berhampur, Odisha, India
[1]saumya97.mishra@gmail.com, [2]mahitamavuduru@gmail.com,[3*]kdnaik0477@gmail.com

***Abstract:*** ***Genetic algorithms (GAs) are stochastic search algorithms inspired by the basic principles of biological evolution and natural selection. GAs simulates the evolution of living organisms, where the fittest individuals dominate over the weaker ones, by mimicking the biological mechanisms of evolution, such as selection, crossover and mutation. GAs have been successfully applied to solve optimization problems, both for continuous (whether differentiable or not) and discrete functions. In recent years it has been observed that cloud computing has laid strong market. Recently, cloud computing emerged as the leading technology for delivering reliable, secure, fault-tolerant, sustainable, and scalable computational services, which are presented as Software, Infrastructure, or Platform as services (SaaS, IaaS, PaaS). Moreover, these services may be offered in private data centers (private clouds), may be commercially offered for clients (public clouds), or yet it is possible that both public and private clouds are combined in hybrid clouds. The provisioning of cloud resources, as to execute or carry out various tasks in critical tasks, as the number of users increases. As the number of task also increases the provisioning of the virtual machines should be such that the throughput will be maximum .In order to achieve the above situation genetic algorithm is taken into consideration and applied in this paper. The paper is divided into five parts. Part I introduces various concepts that have been used in this paper. Part II consists of all the assumption which is taken into consideration while applying genetic algorithm .Part III consists all the steps of genetic algorithm applied and how it is implemented in the process of executing the cloudlet. Part IV consists the final conclusion of the research where the default allocation of virtual machines is compared with genetic algorithm applied. Part V consists of various reference works used.***

***Keyword:*** ***Genetic Algoritm, Evolutionary Algorithm, Cross Over Process, Cloudsim, Java Environment.***

## I. INTRODUCTION

*Genetic algorithm,* in artificial intelligence, a type of evolutionary computer algorithm in which symbols (often called "genes" or "chromosomes") representing possible solutions are "bred." This "breeding" of symbols typically includes the use of a mechanism analogous to the crossing-over process in genetic recombination and an adjustable mutation rate. A fitness function is used on each generation of algorithms to gradually improve the solutions in analogy to the process of natural selection. The process of evolving the genetic algorithms and automating the selection is known as genetic programming. In addition to general software, genetic algorithms are sometimes used in research with artificial life, cellular automatons, and neural networks. Utilization of cloud has been proliferating through recent years due to easy access and security .Cloud computing can be simply defined as "The practice of using a network of remote server hosted on the internet to store, manage and process data rather than local server or a personal computer. The following experiment can be implemented in environments like either Java environment or CloudSim environment. CloudSim Automation is a Java command line tool based on CloudSim and Cloud Reports classes that is able to read specifications of CloudSim simulation scenarios from a YAML file, a very human readable data format. Simulation scenarios can be written inside a YAML file and Cloud Automation Tool reads these simulation scenarios, creates and runs them on CloudSim. By this way, the attention can be focused on the problem to be solved, such as creation of new algorithms to load balancing, new virtual machine scheduling policies, VM placement, resource provisioning, workload prediction, server consolidation, energy efficiency, cost reduction and so on. As already stated by the definition by cloud computing earlier, its major operations can be summarized as storage, management and processing of data. Out of the above three major operations this paper concentrates on processing of data. In the following research, we have considered that each data to be processed as a task and size of each task is considered as million of instruction that a virtual machine has to execute.

## II. ASSUMPTIONS

This part provides a brief description about various assumptions that were made while applying proposed genetic algorithm. Some of the assumptions considered are:

The immigration rate of the task is x2.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

The emigration rate of the tasks in nx where n can be described as the number of virtual machines.

The task lengths are assumed to be within 1000 million instructions per second to 2000 million instruction.

Bandwidth of the virtual machine is considered to --

Maximum time taken to execute a task is 2second.

Execution speed of virtual machine is 1000 million instruction per second.

At any time when the waiting list is full the genetic algorithm starts implementing.

At max 14 tasks are taken to apply genetic algorithm

The number of small tasks are decreasing continuously and the number of large tasks are increasing.

The immigration time between the tasks are so small that they can be considered to immigrate at same time.

## III. IMPLEMENTATION OF PROPOSED ALGORITHM

Genetic algorithm (GA) is a type of evolutionary algorithm (EA).Evolutionary algorithm is a generic population based on meta heuristic optimization algorithm that is inspired by natural evolution.[1]

*Initialization-*

In this step a random population is generated;

Tasks of various sizes enter arrives into the content of cloud. There is no fixed pattern in which they arrive so in the initialization phase the tasks that resides or are waiting in the waiting list are taken and the process is further proceeded.

*Calculation:*

In the calculation of the fitness value of the available population is found out and the people whose fitness value is high is most likely to be selected.

In the context of cloud computing the available tasks that are selected in the initialization process is taken. Since we have assumed that at a time we have only 14 cloudlets therefore we divide it into 7 different batches.

The batches are formed in three different combinations-

In the first combination we have taken the consecutive tasks as one batch.

In the second combination we have considered an alternate tasks as one batch.

In the third combination we have considered a different combination that is first and fourth task is taken in one batch second and fifth task in second batch, third and sixth in third batch and so on.

The time required to complete each batch is calculated in each combination and the combination that requires least time is considered to be fit.

Since we considered only 3 virtual machines therefore the time required for throughput equals to six is considered.

*Selection:*

The selection of the three batch is done according to the fitness value the batch or the combination taking less time to achieve a throughput of 6 is given to the virtual machines.

*Crossover & Mutation:*

In the above phases only six of the available tasks are considered and the rest of the tasks are left out in the waiting list.

Moreover during the execution of the above selected batch more number of the tasks are added to the waiting list. Since we have considered the immigration rate as x2 hence the tasks are increasing exponentially.

With the combination of the new set of population i.e., the left out task and newly arrived task, fitness value is again calculated and selection and crossover processes are repeated until all the tasks finish executing.

*Background:*

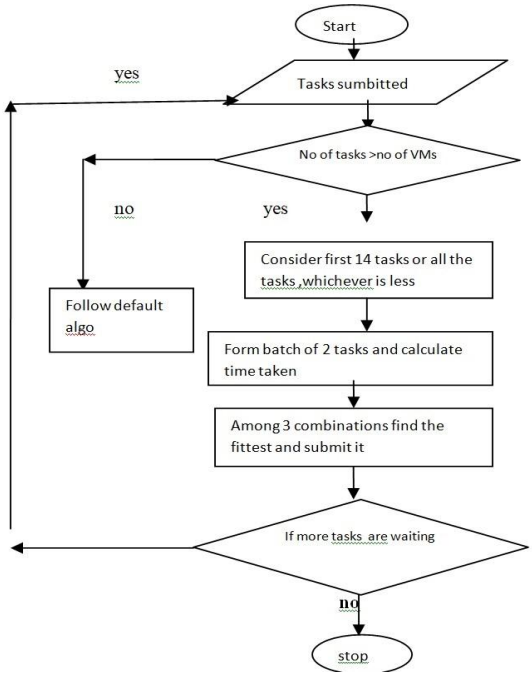



Fig.1.

International Journal of Electrical Electronics & Computer Science Engineering
Volume 5, Issue 1 (February, 2018) | E-ISSN : 2348-2273 | P-ISSN : 2454-1222
Available Online at www.ijeecse.com

*Genetic algorithm:*

1. Take first 14 tasks or the waiting tasks considering the greater one.
2. Form the three different combinations of taking two tasks at a time.
3. Calculate the finishing time for each task.
4. For any particular combination of three batches takes less time consider it as the fitness value and calculate the finishing time.
5. Consider the most suitable batches and give it to other VMs.
6. Repeat the above steps till waiting list is NULL.

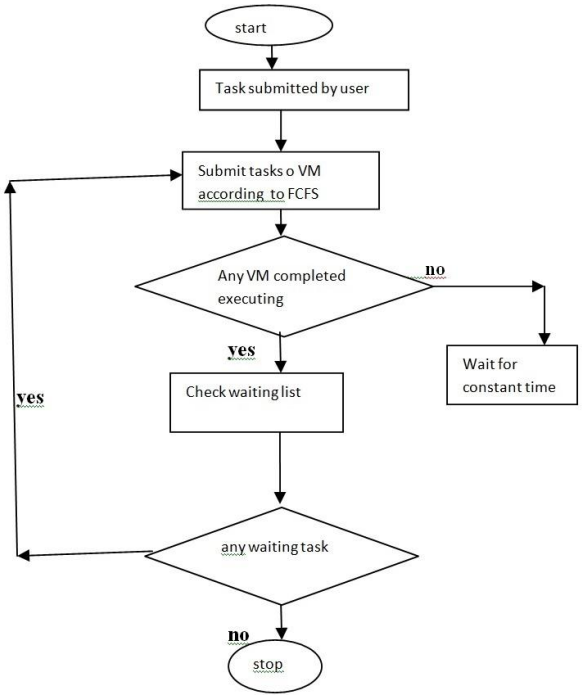



Fig.2.

*Default algorithm:*

Task arrived from the user while(waiting_list!=NULL){

for(i=0;i<=n;i++){

if(vm_status[i]==-1)//free

{

submit task to vm[i];

}}

wait for constant time

}

}

## IV. EXPERIMENT AND RESULTS

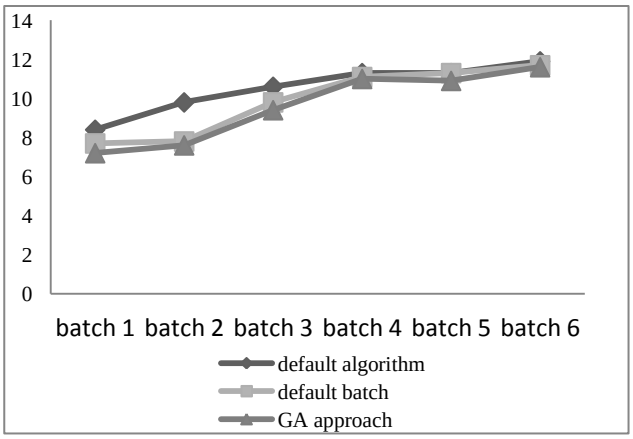
In the context of genetic algorithm, rather than scheduling calculation of fitness value is given more importance.

The comparison of the default allocation of the tasks with the genetic algorithm approach is as follows-

| No of Tasks | Execution Time By Default Allocation (In Sec) | Execution Time By Genetic Approach (In Sec) | Default Algorithm |
|---|---|---|---|
| Batch 1(6) | 7.7 | 7.2 | 8.4 |
| Batch 2(6) | 7.8 | 7.6 | 9.8 |
| Batch3(6) | 9.8 | 9.4 | 10.6 |
| Batch4(6) | 11.1 | 11.0 | 11.3 |
| Batch5(6) | 11.3 | 10.9 | 11.3 |
| Batch6(6) | 11.7 | 11.6 | 11.9 |
| Total 36 | 59.4sec | 57.7sec | 63.3 |

Total optimization of 5.6 seconds in comparison to default but considering the calculation of fitness value (applying GA) 1.6 sec is optimized.

Fig.3




So from above graph it is observed that the time taken in proposed genetic algorithm approach is less than default algorithm allocation.

## V. CONCLUSION & FUTURE WORK

The genetic algorithm approach cannot be considered as the best one but it can be considered as a way of scheduling can optimize the usage of virtual machines in the cloud.

Here we considered only 14 cloudlets during the formation of batches but if we consider the large number of cloudlets then the optimization may increase. Further only three different combination are considered and more combination of task may optimize the utilization further.

## VI. REFERENCE

[1] www.ai-junkie.com/ga/intro/gat1.html

[2] www.cloudbus.org/cloudsim/

[3] www.rennard.org/alife/english/gavgb.pdf

[4] www.scribd.com/document/284431197/Genetic-Algorithms-pdf

[5] Research on resource optimization by D.Pan, J.Cao.

[6] Utility optimization strategy of resource scheduling. By Y.Wang, J.wang, C.Wang.