IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper | Volume-4, Issue-8 | E-ISSN: 2347-2693

Improvement of SLA Parameters in Virtual Machine Migration by using Genetic Algorithm

Anu Thind[1*], Mandeep Devgan[2]

[1]Masters of Technology, Information Technology, Chandigarh Engineering College, Landran
[2] Information Technology, Chandigarh Engineering College, Landran

Available online at: www.ijcseonline.org

Received: 16/Jul/2016 Revised: 10/Jul/2016 Accepted: 20/Aug/2016 Published: 31/Aug/2016

***Abstract*—** Virtualization plays a main role in the cloud computing technology, usually in the cloud computing, users share the data there in the clouds like application etc., but with virtualization users shares the communications. A general technique to enhance the energy proficiency of a datacentre is VM placement by coordinating the quantity of dynamic servers to the present needs of the VMs and setting the remaining servers in low-control standby modes using SLA violations .In cloud computing environments, cloud service users consume cloud resources as a service and pay for service use. Before a cloud provider provisions a service to a consumer, the cloud provider and consumer (or broker) need to establish a service level agreement. So this work explored the utilization of VM migration with GA algorithm in MATLAB environment. We found the performance parameters like Number of VMs, Response time, Throughput and Execution Time.

Keywords*—*Virtual Machine Security, Genetic Algorithm, Deployment Models, Service Level Agreement.

I. INTRODUCTION

The request for Cloud computing usage is growing day by day due to the compensations and abilities it offers. The fast rising rate of the usage of large-scale figuring machines on cloud platform has ensued in increased consumption of energy and emission of carbon. Such negative effects should be restricted for a more environmental-friendly computing platform, e.g. a green cloud computing platform. Development is one of the aspects that can be looked into in regards to the development of the whole cloud construction[1], specifically in the possibility of resource management. One of the significant features that need to be optimized in regards to scheduling is the load balancing procedure that emphasizes on optimum resource utilization, maximum throughput, extreme response time and prevention of overload.

A. Cloud Computing

Cloud is a computing framework which usually denotes storing and accessing data and programs over the internet, instead of your computer's hard drive. Cloud Computing is handled by means of the great prospective paradigm utilized for placement of applications taking place over Internet. This perception also elucidates about the applications which are widen on the way to be manageable over and done with the Internet. Cloud applications utilize huge information centers as well as operational servers which is utilized to host net applications plus services [1]. Cloud computing is a service distribute over the internet for computing, data access and cloud storage by create scalability, elasticity and less cost. Second invention platform for division which suggests [1] a variety of services and applications to the user not including actually attain them. A cloud computing is one of the rising information technologies used in computation now days. It is green technologies which agree to accessing, computing and storing the assets by offer a variety services. A cloud computing is normally includes models like Infrastructure-as-a-service [2], Platform-as-a-service and Platform-as-a-service. To reduce the computation time and to conquer the storage space lack issues the most of the organization now a day's irregular to cloud computing from the established process of calculation. It mainly focuses on allocating data and computations over a scalable information centers of network.

B. Crucial issues in cloud computing

1. Load Balancing
2. Virtual Machine Migration
3. Fault Tolerance

***Load Balancing*:** It is the process of distributing tasks or load evenly among various nodes of the system so that time efficiency can be increased. Also proper utilization of resources takes place.

***Virtual Machine Migration*:** Virtual Machine (VM) migration is a powerful management technique that gives data centre operators the ability to adapt the placement of VMs in order to better satisfy performance objectives, advance resource utilization & communiqué locality, mitigate performance hotspots, achieve fault tolerance, reduce energy consumption[3]. A virtual machine provides interface identical to underlying bare hardware i.e. all devices, interrupts, memory, page tables etc.

Advantages of Virtualization [4]:

- efficient use of resources
- superior degree of abstraction
- Replication
- scalable and flexible infrastructure

***Fault Tolerance*:** Fault tolerance allows the virtual machines to continue their task even if any part of scheme fails. This method migrates the virtual machine from one physical server to another physical server established upon the estimate of the failure occurred. The objective fault tolerant migration technique is to improve the obtainability of physical server & to avoid the presentation degradation of applications[5][6].

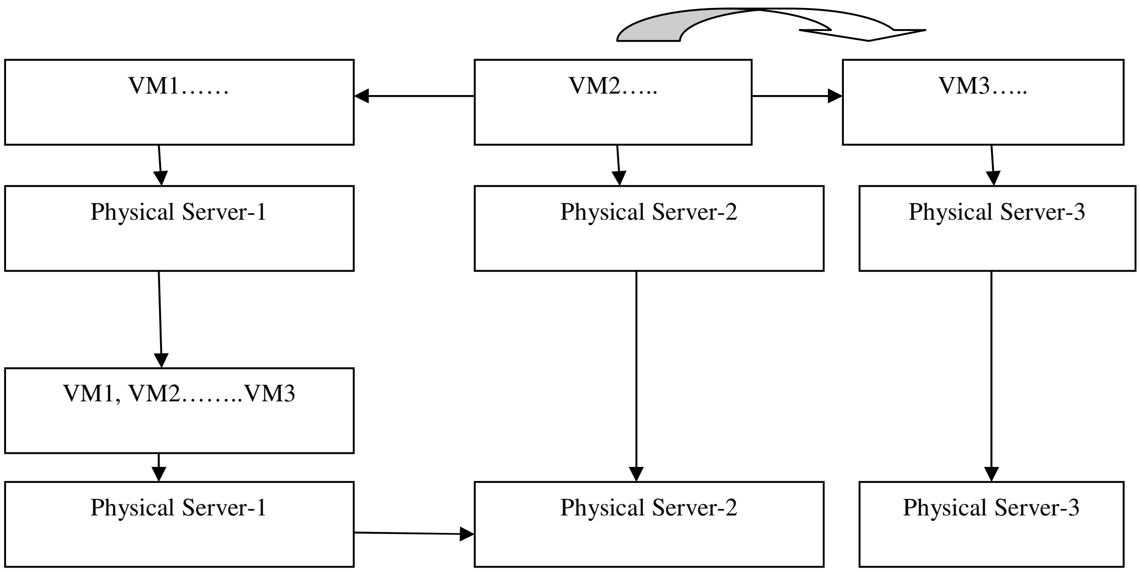

Fig 1: Load balancing using VM

II. METHODOLOGY

A. Improving SLA Parameters

Firstly implement existing minimization of migration (MM) by varying parameters like SLA (response time, execution time, and throughput), no. of customers, virtual machines and no. of request for checking the performance of this technique. Secondly implement the proposed scheme based on three different SLA parameters: response time, execution time, and throughput using GA algorithm. And check the performance of these three options. At the end compare the results to check which option is efficient in terms of energy consumption. It can be written in the following way also:

Step: 1

User Code: It defines the simulation specification, virtual machine specification and system specification for Migration.

User Code: The *User Code* is a computer ID that is used to approve your Scheme Product. Although the *user code* comprises no personal info about you or your computer, every *User Code* is unique to a detailed computer.

Step: 2

Virtual Machine: Arrange VM specification like no. of VMs and simulation time. Then check what load has been put in every repetition, check how much power is getting consumed in every repetition and check that how many tasks are getting completed in every repetition.Then generate task specification on the basis of alive VM id, alive VM power and alive VM load. Then keep track of tasks that are getting completed. Then keep also path of VM that will get task. Then total time for the execution of task which is getting consumed. After this check that which VM has completed how many tasks and checking that total number of provided tasks to one VM does not exceed with the amount of data provided to it.

Step: 3 Same work is done in System Specification and hence the migration is calculated.

Step: 4 After minimization we apply the Evolutionary Algorithm like Genetic Algorithm.

Step: 5 Optimization Technique generate the reduce Index and add the fitness function to provide the fit value.

Step: 6 Fit value provides the evidence for Optimize Result in Virtual Machine.

Step: 7 Then Calculate following parameters;

- Power_Overall_Consumed
- Tasks_Overall_Iteration
- Percentage Of The Completion Of Tasks

Step 8: Evaluate the SLA performance parameters (Response time, Throughput and Execution Time).

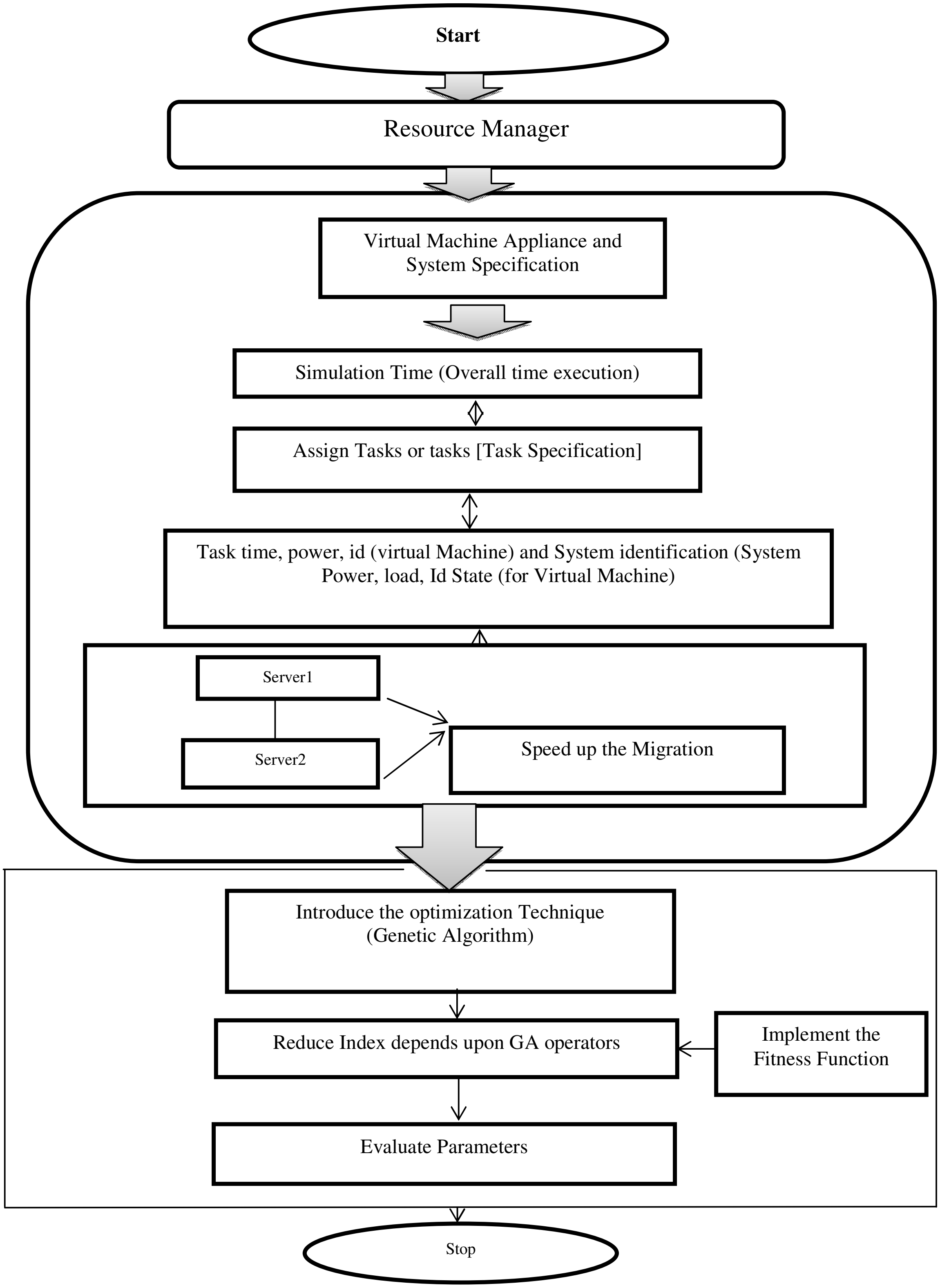

Fig 2: Methodology

B. Genetic Algorithm

Genetic algorithm is computer program that simulates the [7] process of natural evolution in order to solve difficulties and to model evolutionary systems.Vital solutions to optimization problems using operators like selection, mutation and crossover[8],[9],[10] are carried out by using genetic algorithm. It requires a fitness function to generate best solutions. It accomplishes its task as under

```
// Begin with an initial time
            t := 0;

// initialize a usually  general population of individuals
            Init population P (t);

// check fitness of all primary individuals of populace
            evaluate P (t);

// check for the completion condition(time, fitness, etc.)
            while not terminating do

// increment the time counter
        t := t + 1;

// choose a sub-population for generating offspring
           P' := selectparents P (t);

// rejoin the "genes" of selected parents
            recombine P' (t);

//  disturb and unsettle the mated population
             mutate P' (t);

//  compute its next fitness
             evaluate P' (t);

// choose the survivors from original fitness
             P := survive P,P' (t);
            od
    end GA.
```

III. RESULT ANALYSIS

In the proposed work,a numeric dataset with different categories of loads on Virtual machines have been chosen. The main focus is to improve the SLA parameters. The procedure goes as follows; Firstly, Vm count, active VM and percentage of completed tasks have been recorded for 3 iterations. Then on their basis, Power Consumption, total number of completed tasks, Response Time, Execution Time and Throughput have been observed. Finally, the results for Response Time and Execution Time have been compared with previous work. The optimization is carried using genetic algorithm and optimal solution appears in the form of optimized graph as shown in below figures.

A. 1st Iteration

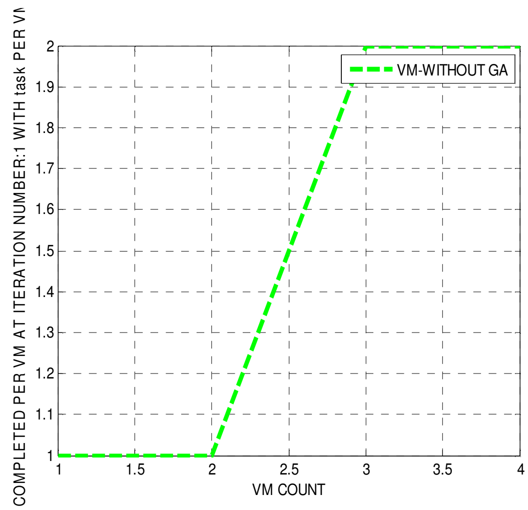

Fig 3.1: Completed tasks

The above figure defines the number of jobs completed per virtual machine at iteration 1. The number of Virtual Count and completed tasks are calculated according to the virtual machine. The Completed number of tasks are 2.

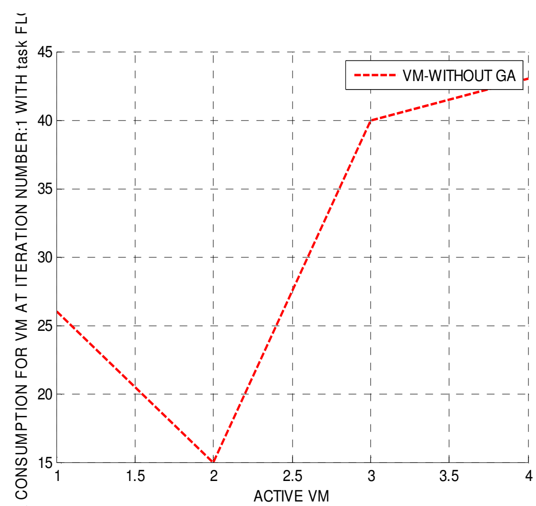

Fig 3.2: Active VM count for iteration = 1 Without GA

Active VM are the number of VM counts that are active at present for resource utilization. The idea of Virtual Machines (VMs) is connected to diminishing energy utilization as it essentially decreases the rate of idle power in the general base. Assuming that the data centre contains a number of physical servers with the resource capacity RC where RC= {RC1, RC2, RC3…..RC_N} is the amount of capacity for each resource (CPU, memory, disk). There are a predefined set of VM types VT= {VT1, VT2, VT3, VT4…..VT_N}. For one period, the datacentre receives Map Reduce tasks from multi users. So above figure displays the active VM for iteration 1.

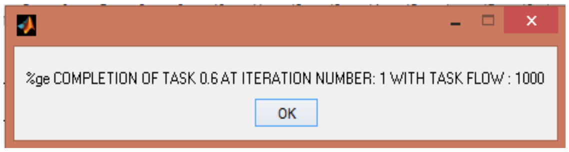

Fig 3.3: Task Completion Percentage for iteration = 1

Above window gives the percentage of completion of tasks for iteration 1.

B. 2nd Iteration

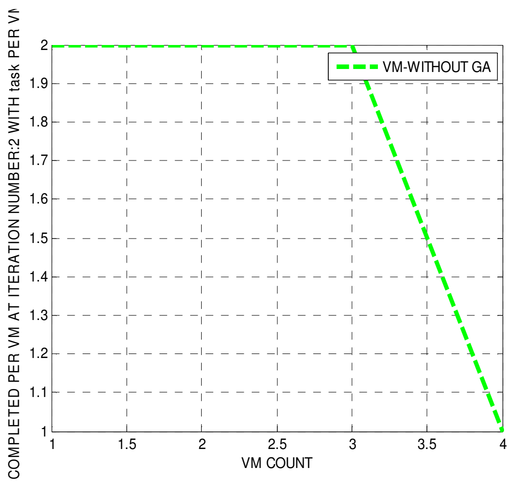

Fig 3.4: VM count for iteration = 2

The above figure defines the number of jobs completed per virtual machine at iteration 2. The number of Virtual Count and completed tasks are calculated according to the virtual machine. So above figure shows the VM count for iteration 2 with number of total tasks completed = 15.

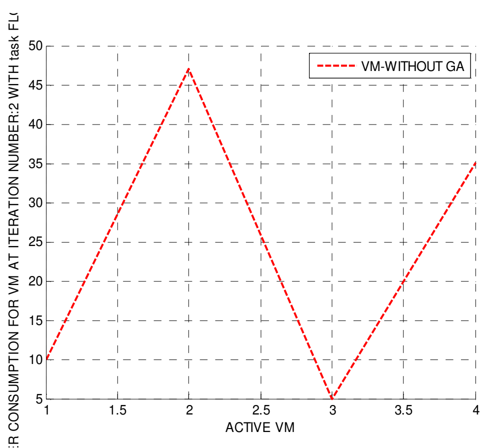

Fig 3.5: Active VM count for iteration = 2

Active VM are the number of VM counts that are active at present for resource utilization. The idea of Virtual Machines (VMs) is connected to diminishing energy utilization as it essentially decreases the rate of idle power in the general base. Assuming that the data centre contains a number of physical servers with the resource capacity RC where RC = {RC1, RC2, RC3…..RC_N} is the amount of capacity for each resource (CPU, memory, disk). There are a predefined set of VM types VT= {VT1, VT2, VT3, VT4…..VT_N}. For one period, the datacentre receives Map Reduce tasks from multi users. So above figure displays the active VM for iteration 2.

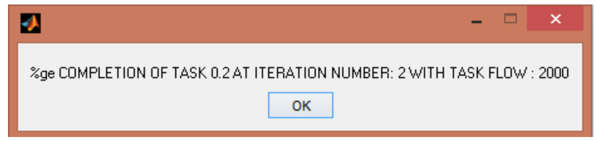

Fig 3.6: Task Completion Percentage for iteration = 2

Above window gives the percentage of completion of tasks for iteration 2.

C. 3rd Iteration

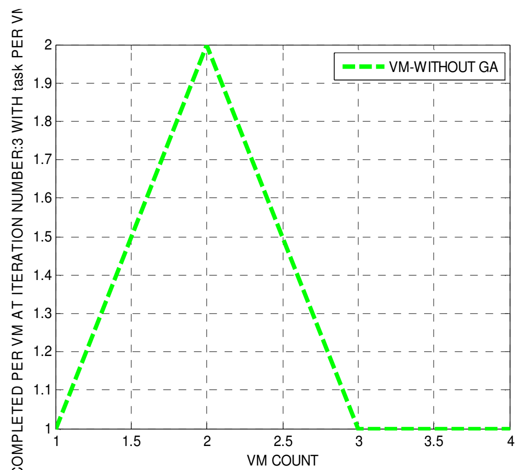

Fig 3.7: *.VM* count for iteration = 3

The above figure defines the number of jobs completed per virtual machine at iteration 1. The number of Virtual Count and completed tasks are calculated according to the virtual machine. So above figure shows the VM count for iteration 3 with number of total task completed = 12.

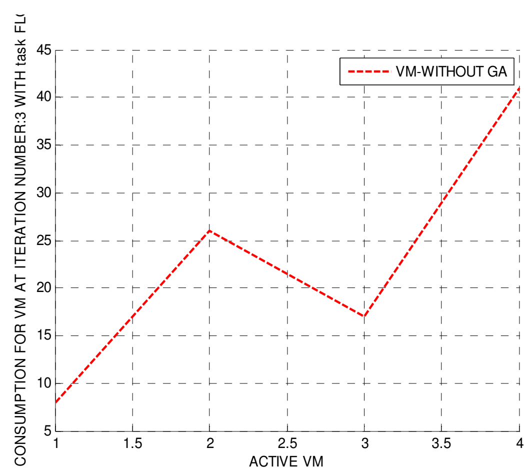

Fig 3.8: Active VM count for iteration = 3

Active VM are the number of VM counts that are active at present for resource utilization. The idea of Virtual Machines (VMs) is connected to diminishing energy utilization as it essentially decreases the rate of idle power in the general base. Assuming that the data centre contains a number of physical servers with the resource capacity RC where RC = {RC1, RC2, RC3…..RC_N} is the amount of capacity for each resource (CPU, memory, disk). There are a predefined set of VM types VT= {VT1, VT2, VT3, VT4…..VT_N}. For one period, the datacentre receives Map Reduce task from multi users. So above figure displays the active VM for iteration 3.

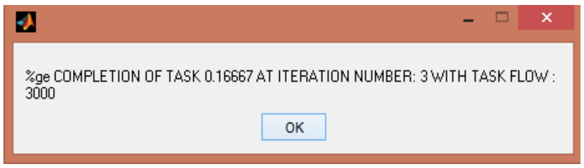

Fig 3.9: Task Completion Percentage for iteration = 3

Above window gives the percentage of completion of tasks for iteration 3.

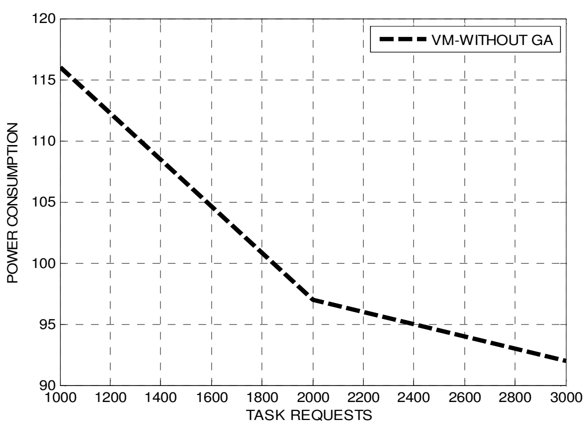

Fig 3.10: Power consumption without GA

The above figure defines the Power Consumption versus Task Request.

Today's computer systems consists of electronic circuits with millions of transistors. Every transistor consumes energy that is based on power. Power consumption by VMs in datacentre is mostly determined by the CPU, memory, disk storage and network interfaces. It has been found out that the power draw by servers even when idle, servers used over 60% of the power consumed by the server running at the full CPU speed. Therefore, in this work we use the power model as:

$$P = CV$$

Where P= Power consumed, C = load capacitance, V is voltage supply. So above figure shows that the power consumption is 1350 J.

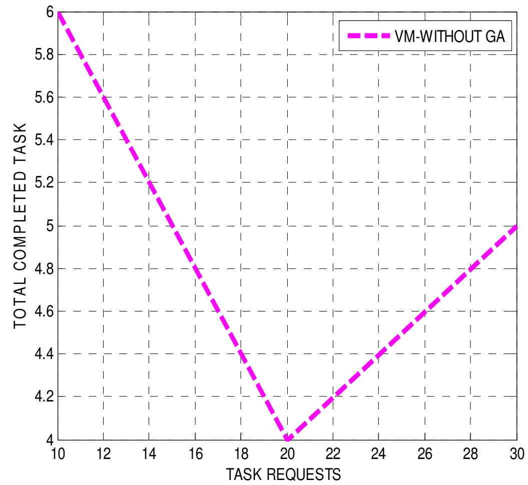

Fig 3.11: Total Completed Task versus Task Request

Each user can access only single application at one time. None of the other users can access others servers except in cloud computing. Number of requests per period for each user are assumed to be same. So it can be described as below:

$$\text{Total completed tasks} = \frac{\text{no. of requests(Rn)}}{\text{Total no. of requests)}}$$

Above figure shows the total completed task = 53.

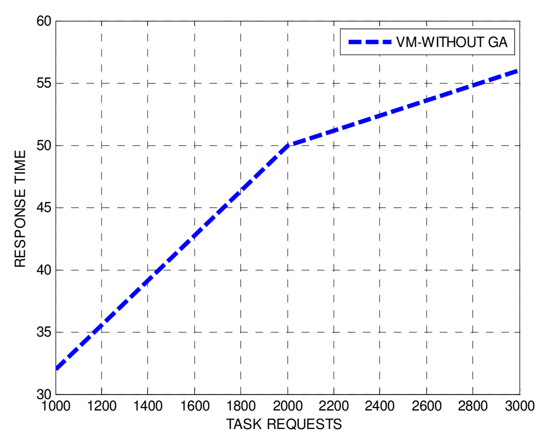

Fig 3.12: Response Time

The above figures define the response time according to the task requests. The response time decreases according to the task requests.

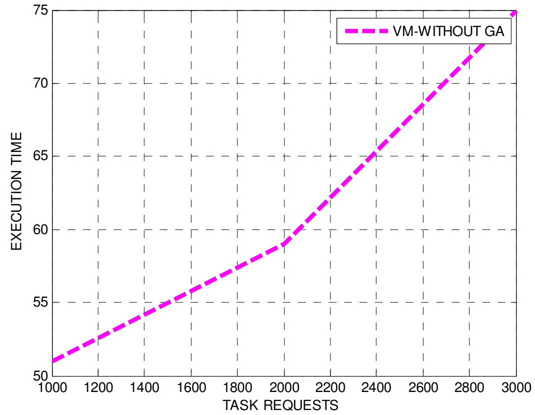

Fig 3.13: Execution Time without GA

The above figure defines the execution time according to the task requests. Execution Time increases as the task request increases.

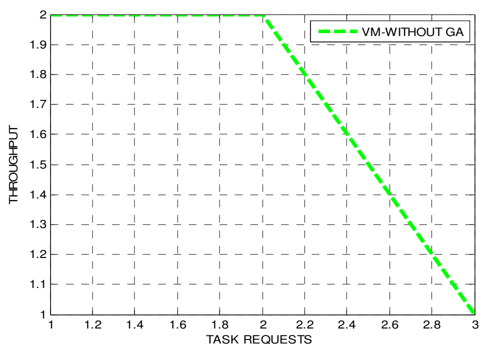

Fig 3.14: Throughput without GA

The above figure defines the throughput without GA according to the task requests. As the number of task requests increases, the Throughput decreases.

D. Implementation of Genetic Algorithm

1. 1st Iteration

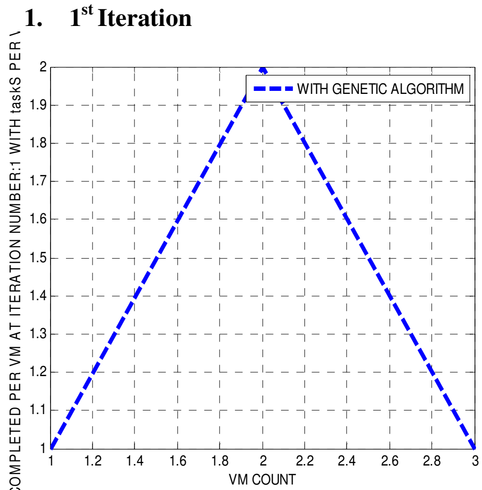

Fig 3.15: VM Count for iteration = 1 with GA

The above figure defines the number of jobs completed per virtual machine at iteration 1. The number of Virtual Count and completed tasks are calculated according to the virtual machine. So above figure shows the VM count for iteration 1 with number of total Task completed = 16.

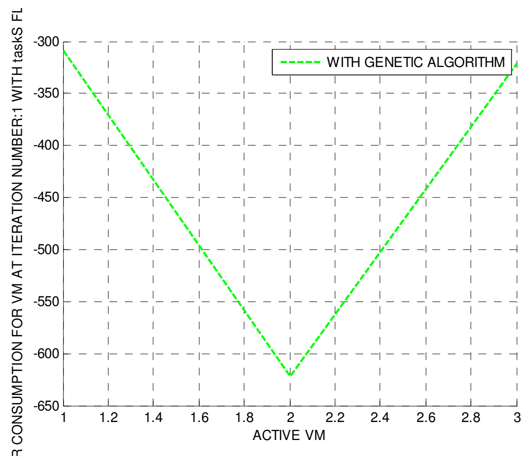

Fig 3.16: Active VM count for iteration = 1 with GA

Active VM are the number of VM counts that are active at present for resource utilization. The idea of Virtual Machines (VMs) is connected to diminishing energy utilization as it essentially decreases the rate of idle power in the general base. Assuming that the data centre contains a number of physical servers with the resource capacity RC where RC = {RC1, RC2, RC3…..RC_N} is the amount of capacity for each resource (CPU, memory, disk). There are a predefined set of VM types VT= {VT1, VT2, VT3, VT4…..VT_N}. For one period, the data centre receives Map Reduce tasks from multi users. So above figure displays the active VM for iteration 1.Increase the Active VM in iteration 1 with GA.

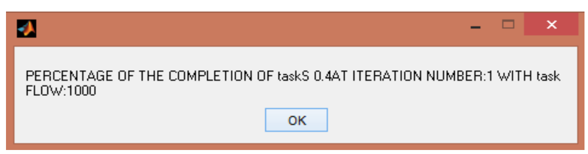

Fig 3.17: Task Completion Percentage for iteration = 1 with GA

Above window gives the percentage of completion of tasks for iteration 1. Completed tasks are 40%.

2. 2nd Iteration

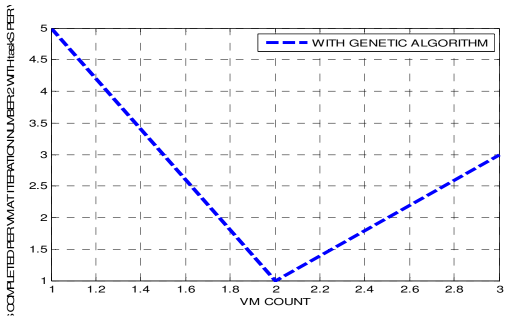

Fig 3.18: VM count for iteration = 2 with GA

The above figure defines the number of jobs completed per virtual machine at iteration 1. The number of Virtual Count and completed tasks are calculated according to the virtual machine. So above figure shows the VM count for iteration 2 with number of total tasks completed = 30.

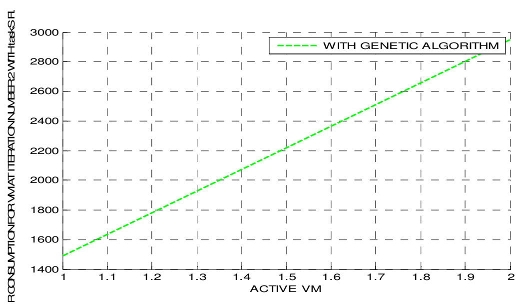

Fig 3.19: Active VM count for iteration = 2

Active VM are the number of VM counts that are active at present for resource utilization. The idea of Virtual Machines (VMs) is connected to diminishing energy utilization as it essentially decreases the rate of idle power in the general base. Assuming that the data centre contains a number of physical servers with the resource capacity RC where RC = {RC1, RC2, RC3…..RC_N} is the amount of capacity for each resource (CPU, memory, disk). There are a predefined set of VM types VT= {VT1, VT2, VT3, VT4…..VT_N}. For one period, the datacentre receives Map Reduce tasks from multi users. So above figure displays the active VM for iteration 2.

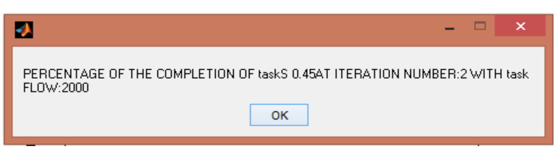

Fig 3.20: Task Completion Percentage for iteration = 2

Above window gives the percentage of completion of tasks for iteration 2.

3. 3rd Iteration

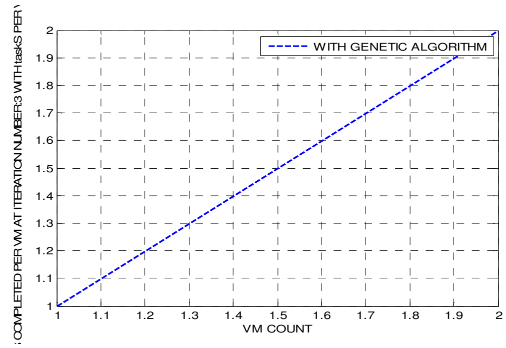

Fig 3.21: VM count for iteration = 3 with GA

The above figure defines the number of jobs completed per virtual machine at iteration 1. The number of Virtual Count and completed tasks are calculated according to the virtual machine. So above figure shows the VM count for iteration 3 with number of total tasks completed = 17 with Genetic Algorithm .

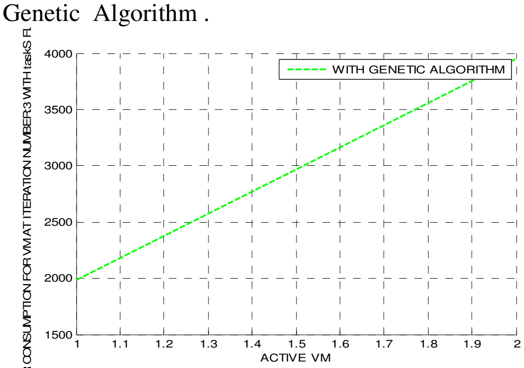

Fig 3.22: Active VM count for iteration = 3

Active VM are the number of VM counts that are active at present for resource utilization. The idea of Virtual Machines (VMs) is connected to diminishing energy utilization as it essentially decreases the rate of idle power in the general base. Assuming that the data centre contains a number of physical servers with the resource capacity RC where RC= {RC1, RC2, RC3…..RC_N} is the amount of capacity for each resource (CPU, memory, disk). There are a predefined set of VM types VT= {VT1, VT2, VT3, VT4…..VT_N}. For one period, the datacentre receives Map Reduce tasks from multi users. So above figure displays the active VM for iteration 3.

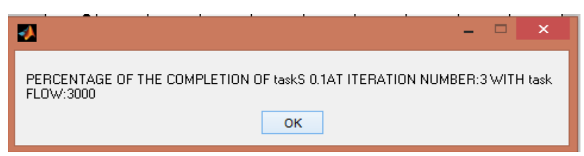

Fig 3.23: Task Completion Percentage for iteration = 3

Above window gives the percentage of completion of tasks for iteration number 3.

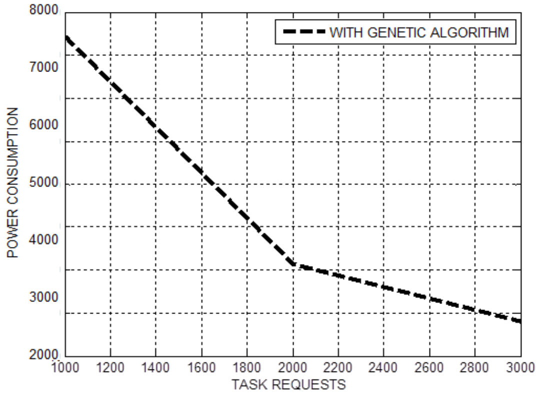

Fig 3.24: Power Consumption with GA

Above window gives the Power Consumption versus Task Request.

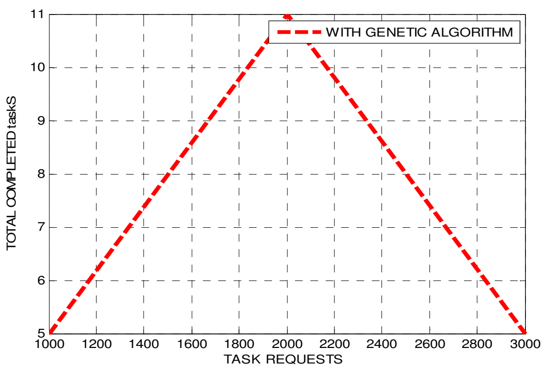

Fig 3.25: Total Completed Tasks versus Task Requests

Each user can access only single application at one time. None of the other users can access others servers except in cloud computing. Number of requests per period for each user are assumed to be the same.

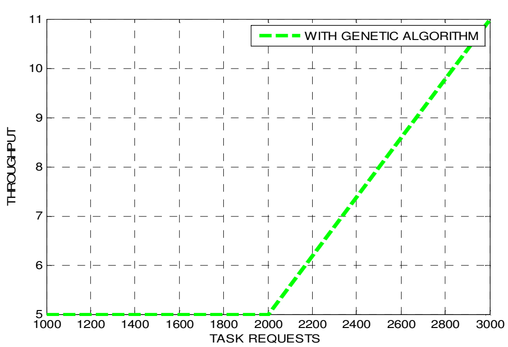

Fig 3.26: Throughput with GA

The above figure defines the throughput with genetic algorithm. As the Task Request increases, the Throughput increases.

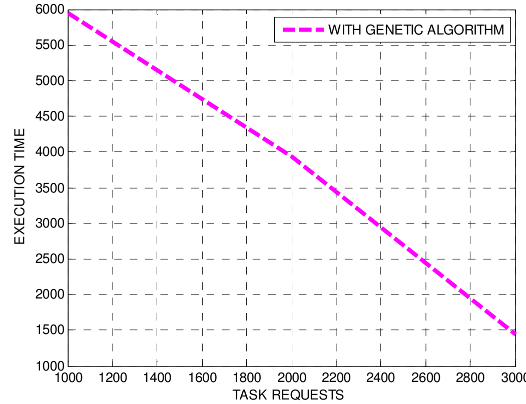

Fig 3.27: Execution Time with GA

The above figure defines the execution time with Genetic Algorithm. As the Task Requests increases, the execution time decreases.

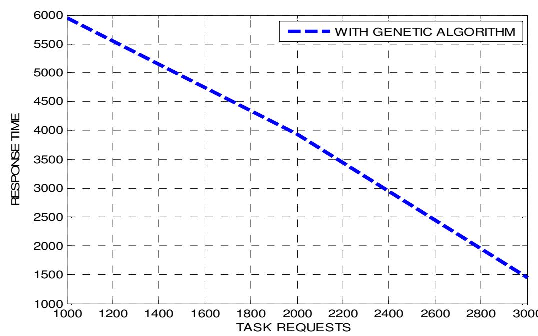

Fig 3.28: Response Time with GA

The above figure defines that the response time decreases as the task request increases.

IV. COMPARISON AND DISCUSSION

A. Comparison Between Response Time of Existing and Proposed Work

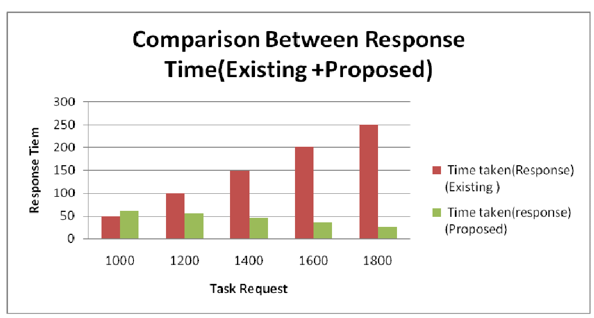

Fig 4.1: Comparison Between Response Time(existing and Proposed Work)

The above figure defined that the comparison between response time according to the task request. In previous work according to task request increase the response time and using Genetic approach decrease the response time.

B. *Comparison Between Execution Time of Existing and Proposed Work*

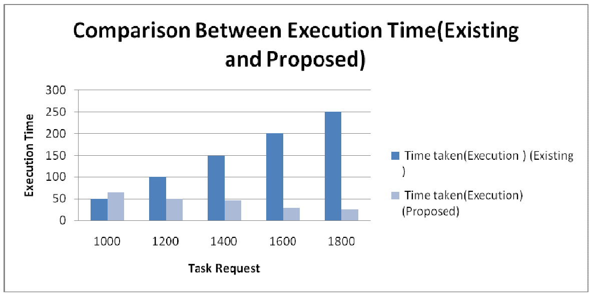

Fig 4.2: Comparison Between Execution Time (Previous and proposed)

The above figure defined that the comparison between execution time according to the task request. In previous work according to task request increase the execution time and using Genetic approach decrease the execution time.

Task Request	Response Time (Existing)	Response Time (Proposed)
1000	50	60
1200	100	55
1400	150	45
1600	200	35
1800	250	25

a. Comparison between Response Time (existing and proposed)

Task Request	Time taken(Execution) (Existing)	Time taken(Execution) (Proposed)
1000	50	65
1200	100	50
1400	150	45
1600	200	30
1800	250	25

b. Comparison between execution Time (existing and proposed)

V. CONCLUSION

This work presented a comparative review of performance measuring parameters of energy minimization in cloud computing using VM migration. In cloud computing environments, cloud service users consume cloud resources as a service and pay for service use.

In this paper, firstly we have reviewed the earlier work done then proposed a new algorithm based on GA method. As, today's computer systems consists of electronic circuits with millions of transistors. Every transistor consumes energy that is based on power. Power consumption by VMs in data centre is mostly determined by the CPU, memory, disk storage and network interfaces. It has been found out that the power draw by servers even when idle, servers used over 60% of the power consumed by the server running at the full CPU speed. So obtained value for the power consumption is 1350 J and the total completed Tasks = 72.

In proposed work , Every transistor consumes energy that is based on power consumption. The server used over 50% power consumed by the server running at the complete CPU speed. So obtained value for the power consumption is 1250J and the Total task completed = 80. Calculate the Service level agreement parameters like response time, execution time and throughput with and without genetic algorithm.

VI. FUTURE SCOPE

In future, ABC (Ant Bee Colony) + GA (Genetic Algorithm) hybrid approach can be implemented to reduce the power consumption, more task complete, improve the performance the SLA parameters. Less decrease the Execution time and response time. Improve the performance of the servers via throughput.

REFERENCES

[1] A. A. Patel, J. N. Rathod, "*Reducing Power Consumption & Delay Aware Resource Allocation in Cloud Data centers*", Journal of Information, Knowledge and Research in Computer Engineering, V.02, 2010, pp.337-339.

[2] B.Wei, C.Lin, X.Kong, *"Energy Optimized Modeling for Live Migration in Virtual Data Center*", International Conference on Computer Science and Network Technology, 2011, pp. 2311-2315.

[3] A.Beloglazov, R.Buyya, *"Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing"*, 2012, pp.755-768.

[4] A.Huth, J.Cebula, *"The Basics of Cloud Computing"*, US-*CERT*, 2011.

[5] Wu, Chih-Hung, Wei-Lun Chen, and Chang Hong Lin. *"Depth-based hand gesture recognition."* Multimedia Tools and Applications (2015): 1-22.

[6] Justina, J. P., and SangeethaSenthilkumar. *"A Survey Analysis on: Vision-Based Hand Gesture Recognition."* International Journal 4.1 (2014)

[7] C.Lin, P.Liu, J. Wu, *"Energy-Aware Virtual Machine Dynamic Provision and Scheduling for Cloud Computing"*, IEEE 4th International Conference onCloud Computing, 2011, pp.736-

737.
[8] Goldberg, David E., and John H. Holland. "Genetic algorithms and machine learning." Machine learning 3.2 (1988): 95-99.
[9] D.Jayasinghe, C.Pu, T. Eilam, M.Steinder, I.Whalley, E.Snible, *"Improving Performance and Availability of Services Hosted on IaaS Clouds with Structural Constraint-aware Virtual Machine Placement",* IEEE InternationalConference on Services Computing, 2011, pp.72-79.
[10] Frank Schulz , *"Towards Measuring the Degree of Fulfillment of Service Level Agreements",*Third International Conference on Information andComputing,2010,pp.273-276.
[11] Grace Lewis, *"Basics about Cloud Computing*", Software Engineering Institute, September 2010.
[12] Ian Foster, *"Developing Models",* http://www.mcs.anl.gov/~itf /dbpp/text/ node29. html, last accessed on 31 july,2013.
[13] K.Ye, D.Huang, X.Jiang, H.Chen, S.Wu, *"Virtual Machine Based Energy-Efficient Data Center Architecture for Cloud Computing: A Performance Perspective"*, IEEE/ACM International Conference on Green Computing and Communications, 2010, pp.171-178.
[14] M.Maurer, V.C.Emeakaroha, I.Brandic, J.Altmann, *"Cost–benefit analysis of an SLA mapping approach for defining standardized Cloud computing goods",*2012,pp.39-47.
[15] N. Sadashiv, S. M. Dilip Kumar, *"Cluster, Grid and Cloud Computing: A Detailed Comparison*", The 6th International Conference, August 2011, v 3.33, pp.477-482.

AUTHORS PROFILE

Anu Thind is presently pursuing Masters of Technology from Chandigarh Engineering College, Landran in the field of Information Technology. She received her Bachelor's degree in 2014 from SVIET, Banur.

Mandeep Devgan is presently working as an Assistant Professor in IT department at Chandigarh Engineering College, Landran. He has presented/published a number of papers in national and international conferences/journals.

