Efficient Task Scheduling for Cloud Computing

Miss. Archana Bhat, Dr. Demian Antony D' Mello
2nd year,M.Tech,Professor
Dept. of CSE,Dept. of CSE,
St. JosephEngineering CollegeSt. JosephEngineering College
Mangalore, Karnataka, India.Mangalore, Karnataka, India.

***Abstract*---Cloud Computing is a computing platform for the next generation of the Internet.However, data centres hosting Cloud applications consume huge amounts of electrical energy, contributing to high operational costs and carbon footprints to the environment. Therefore, we need Green Cloud computing solutions[1] that can not only minimize operational costs but also reduce the environmental impact. Scheduling is one of the aspects which facilitate Green computing.**

Scheduling refers to the appropriate assignment of tasks to the resources available like CPU, memory and storage such that there is a maximum utilization of resources. Out of the various issues regarding a cloud, scheduling of user's jobs plays a very important role in determining the quality of service provided by the cloud to its customer. An effective scheduling policy is a necessity for quality service as well as providers.

We have validated our approach by conducting a performance evaluation study with experimentation using the CloudSim toolkit. The results demonstrate that the proposed schedulingfor a cloud computing model has immense potential as it demonstrates high potential for the improvement of energy efficiency under workload.

Keywords: Cloud computing, Scheduling.

I. INTRODUCTION

"Cloud" computing – a comparatively recent term, defines the ways ahead in technology world. Being designed on decades of analysis it utilizes all recent achievements in virtualization, distributed computing, utility computing and networking. Cloud computing could be a comparatively new manner of relating the utilization of shared computing resources, and it's another to having native servers handle applications. Cloud computing groups together massive numbers of pc servers associate degree different resources and generally offer their combined capability on an on-demand, pay-per-cycle basis. The top users of a cloud computing network sometimes don't have any plan wherever the server's square measure physically placed them merely spin up their application and begin operating.

Cloud computing infrastructures will enable enterprises to realize additional economical use of their IT hardware and computer code investments. Cloud computing is associate degree example of associate degree ultimately virtualized system, and a natural evolution for knowledge centers that use machine-controlled systems management, work equalization, and virtualization technologies. Cloud computing is that the maturation and coming back along of many previous computing ideas like Grid Computing, ASP, Server Hosting, Utility Computing and Virtualization.

One implication of Cloud platforms is the ability to dynamically adapt the amount of resources provisioned to an application in order to attend variations in demand that are either predictable and occur due to access patterns observed during the day and during the night or unexpected and occurring due to a subtle increase in the popularity of the application service. This capability of clouds is especially useful for elastic applications, such as web hosting, content delivery and social networks that are susceptible to such behaviour.

II.PROBLEM STATEMENT

In the world of Cloud Computing, scheduling of the user's workload is an interesting issue which is open for research.The success of this rising technology relies on the effectiveness of techniques used to execute the user's jobs within the best approach achievable. So to attain this, the subsequent state of affairs has been obsessed because the downside statement.

Problem Definition:*Let C={d1, d2,…….,dn}, where C is a cloud. d1, d2,…..,dn are the datacenters. Let di ={s1,s2,………,sm}, where di is a datacenter and s1, s2,….sm are the servers in each datacenters..Let sj={vj,1, vj,2,……,vj,q} where vj,1, vj,2,….vj,q are the virtual machines in server j. LetW={j1, j2,j3,…..,jp} be the workload that is to be executed in the cloud. Let j1, j2, ,j3,….jp areindividual jobs such that p>l.. Find schedule for a set of heterogeneous workload 'W' in cloud 'C' such that each resource in C is optimally utilized and the overall execution time of the workload 'W' is minimal.*

The problem definition states that the aim is to propose a technique to efficiently schedule user's workload such that the utilization difference and overall execution time for the jobs is as less as possible.

III.PROJECT SCOPE

Scheduling using priority takes less time, overloading of some resources takes place. Sorted jobs are submitted to sorted virtual machines. Graph data structure is not used. Tasks are not grouped. The algorithm provides improved result when compared to traditionally used scheduling algorithms.

Scheduling using group based takes lesser time and load balancing technique is used to solve the problem of Overloading. VMs are arranged as nodes of a graph. Jobs are submitted to the leaf nodes which delegates jobs to its parents. The data structure used is graphs Sorted jobs are assigned to VMs in groups. If more number of job lengths is available then find the utility factor i.e occupied / capacity. Apply utility factor to all the nodes. Wherever there is minimum utility factor apply the job length. Remaining job lengths will be applied in the same sequence.

While adding new virtual machine before task, add to a last leaf and use bottom up approach. Swap till the root satisfies the virtual machine tree requirements and calculate its bandwidth, RAM and keep it ready. While adding new job after allocation before execution clearly prove the resources going to use before execution

IV.METHODOLOGY

A.*Proposed Architecture*

Consider a scheduling scenario in a cloud, where user jobs are to be executed such that they consume minimum execution time.

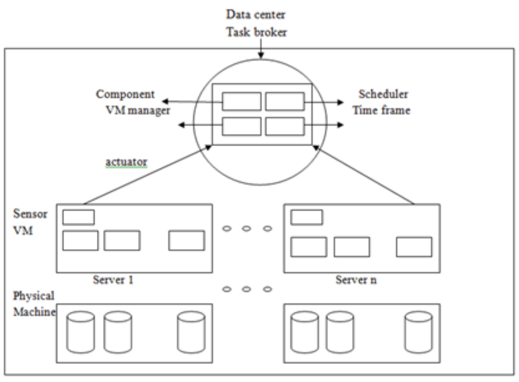

Figure 4.1: General Architecture for a Scheduling Strategy

a) Service Scheduler: Assigns requests to VMs and determines resource entitlements for the allotted VMs. If the automotive vehicle scaling practicality has been requested by a client, it conjointly decides once VMs area unit to be another or removed to fulfil the demand.

b) VM Manager: Keeps track of the provision of VMs and their resource usage. It is in charge of provisioning new VMs as well as reallocating VMs across physical machines to adapt the position.

c) VMs: Multiple VMs will be dynamically started and stopped on one physical machine per incoming requests, thus providing the flexibility of configuring varied partitions of resources on identical physical machine to totally different needsof service requests. Multiple VMs will at the same time run applications supported totally different software system environments on one physical machine. Bydynamically migrating VMs across physical machines, workloads willbe consolidated and unused resources will be switched to a low-power mode, turned off or designed to work at low-performance levels (e.g. victimization DVFS) so as to avoid wasting energy.

*d) Physical Machines:*The underlying physical computing servers give the hardware infrastructure for making virtualized resources to fulfill service demands.

B. Scheduling Algorithm

```
//Input: workload (jobs)
//Output: Time spent in executing each job
by each VM
//Algorithm: createBroker()

1.   Sort job lengths
2.   Sort VMs
3.   changeOrder()
Algorithm: changeOrder()

1.   Get the job length id and length
2.   Create tree
3.   Group job lengths based on no of leaves
4.   Store the order of tree traversal nodes in array
5.   For each traversal
5.1    While job lengths are present in a group
5.1.1     For each node in the particular traversal
5.1.1.1   If flag=1 and if leaf node
                Bind job length to root VM
              Flag=0
                Calculate finish time of node

5.1.1.2   Else
                Bind job length to the
particular node VM
              Calculate finish time of node

5.1.1.3   Consider next job length in the array
5.1.1.4   If end of group
              Break

5.1.2      If more than 2 nodes in the traversal
5.1.2.1      If leaf node
5.1.2.1.1        If finish time of leaf > finish time of  root or
                 finish time of leaf > finish
                 time of  root's first child
5.1.2.1.1.1      flag=1
```

V. IMPLEMENTATION

A. Overview OfCloudsim

In our project, we have made use of CloudSim simulator which is the basic framework for the simulation of cloudlets in cloud computing environment. The features of CloudSim are that it supports modelling and simulation of cloud computing systems and applications, helps in the provisioning of resources to the virtual machine, supports dynamic insertion of simulation elements and stop and resume of simulation.CloudSim is a class which provides entities like Virtual Machines, Hosts, Datacenters and Cloudlets. Figure 5.1 depicts CloudSim core simulation engine.

The purpose of CloudSim is to offer its users an extensible framework for modelling and experimenting with application services and Cloud computing infrastructures.

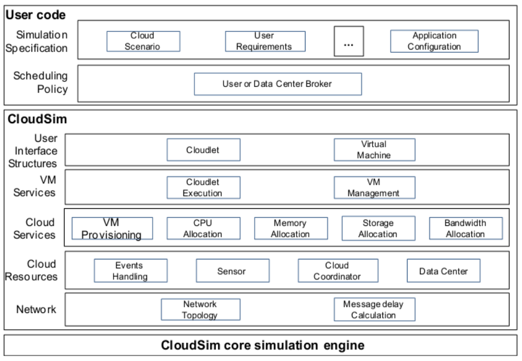

Figure 5.1: CloudSim core simulation engine

It will able to focus on and investigate specific system design issues without worrying about the Cloud-based infrastructures and services. CloudSim is cross-platform and it works on Mac OS X, Windows and Linux.CloudSim layer manages the instantiation and execution of core entities (VMs, hosts, data centres, application) during the simulation period. This layer is capable of concurrently instantiating and transparently managing a large scale Cloud infrastructure consisting of thousands of system components.

VI. EXPERIMENTS AND EVALUATION

A. Illustration

The proposed application undergoes following steps:

a) Sorting joblength and VMs: Sorting joblength and VMs are sorted using heapsort.A heap sort is efficient for data that is already stored in a binary tree.

b) Grouping of joblength allocated to VMs: Sorted VMs will be grouped based on the leaf nodes job length.

c) Adding new virtual machines before task: Once grouping is being done VMs will be added before the PEs of each task.

d) Finding the average response time

e) Finding turnaround time.

B. Experiments

The inputs given to simulation, cloudlets, host, VMs and random parameters will be kept as the typical values set in CloudSim.

Table 6.1: Simulation

Parameters	Values
Repeated simulation runs	30
Initial seed	42
Maximum interval	50
Minimum interval	100
Counter	100
Final cut %	5

Table 6.2: Cloudlets

Parameters	Values
Min number of cloudlets in each request	0
Max number of cloudlets in each request	5
Length Min	40000
Length Max	40000
File size Min	300
File size Max	300
Output size Min	300
Output size Max	300
PES number Min	1
PES number Max	1

Table 6.3: Hosts

Parameters	Values
Host number	5
RAM	16384
Storage	1000000
Bandwidth	100000
PE number	4
MIPS per PE	1000
Architecture	X86
Operative System	Linux
Virtual Machine Manager	Zen
Cost per bandwidth	0.1
Time Zone	10
Cost Per Host	3
Cost per Memory	0.05
Cost per Storage	0.1

Table 6.4: VM

Parameters	Values
Storage Required Min	10000
Storage Required Max	10000
RAM Required Min	512
RAM Required Max	512
Bandwidth Required Min	1000
Bandwidth Required Max	1000
MIPS Min	250
MIPS Max	250
PES Min	1
PES Max	1
Virtual Machine Number Min	5
Virtual Machine Number Max	5

Table 6.5: Random

Parameters	Values
Linear	
Gaussian - Mean	5
Standard Deviation	1
Exponential- Rate	5

C. Results

By using the above Table values it gives a graph as shown below.

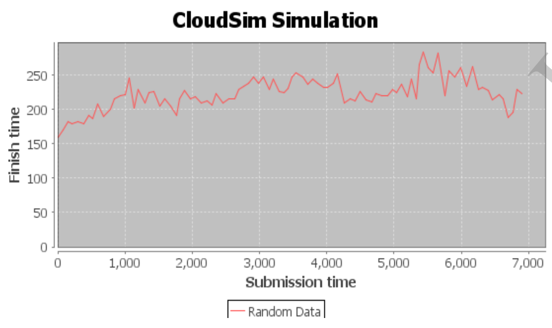

Figure 6.1: Progressive graph of submission time versus finish time.

D.Evaluation

As the submission time increases the finish time will remain in the same range, thus causing the host to utilize the VMs execution potential at the best.

Turnaround time for above graph is 0.199ms.

Average response time=39.035ms

Consider if there are three systems. Suppose first system finishes the job and second system takes more time compared to first system and third accordingly then first system takes job from second system automatically and try to finish the job of all three systems almost same time. Therefore execution of termination is almost at the same time. So load balancing technique is used, by distributing the job to multiple systems.

By analysing experiments, the calculated average response time and turnaround time are shown in the graph taking host as x-axis.

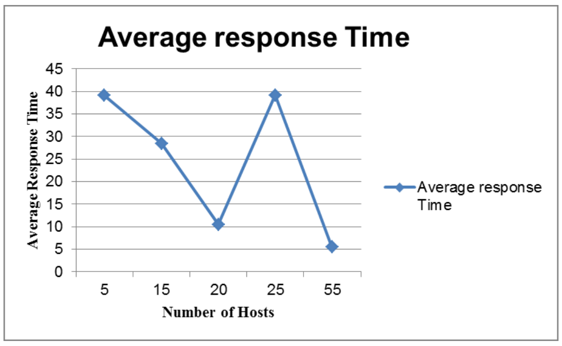

Figure 7.2: Average Response time

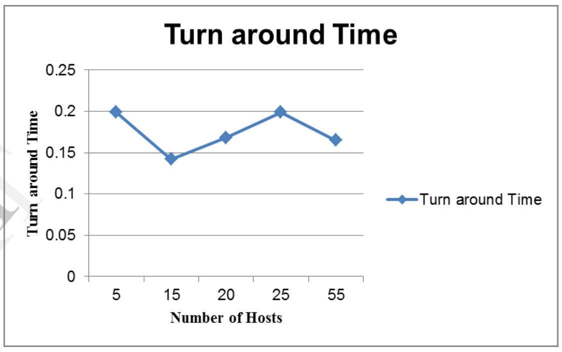

Figure 7.3: Turn Around time

VII. CONCLUSION

Efficient task scheduling mechanism can meet users' requirements and improve the resource utilization, thereby enhancing the overall performance of the cloud computing environment.Scheduling is one of the key issues in the management of application execution in cloud environment.We are using scheduling method for energy efficient cloud environment and have been utilizing the VM, VMs potential against job length scheduling.

In our project, we have made use of CloudSim simulator which is the basic framework for the simulation of cloudlets in cloud computing environment. The features of CloudSim are that it supports modelling and simulation of cloud computing systems and applications, helps in the provisioning of resources to the virtual machine, supports dynamic insertion of simulation elements and stop and resume of simulation.CloudSim is a class which provides entities like Virtual Machines, Hosts, Datacenters and Cloudlets.

By giving inputs we have obtained a progressive graph of submission time versus finish time.As the submission time increases the finish time will remain in the same range, thus causing the host to utilize the VMs execution potential at the best. The work proposes an optimized graph based algorithm for scheduling of user tasks on a cloud infrastructure. The proposed algorithm not only minimizes the cost and finish time of workload execution, it also ensures optimal utilization of resources.

The performance evaluation provides on how the ability to assignspecific virtualmachine types to specific tasks of a processing job,as wellas the possibility to automatically allocate/deallocate virtual machines in the course of a job execution. The work also takes load balancing into consideration i.e. guaranteeing that resources do not get overloaded.It also reduces the consumption of power less than previously existing methods and provides maximum utilization.

There may be other factors which can be obtained only after the practical implementation. We can improve our project by using hybrid cloud where one data centres have multiple clouds of different configurations which should be able to move.

REFRENCES

[1] Anton Beloglazov ,JemalAbawajy, RajkumarBuyya "Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing", Future Generation Computer Systems 28 (2012) 755–768.

[2] AshutoshIngole ,SumitChavan , UtkarshPawde "An Optimized Algorithm for Task Scheduling based on Activity based Costing in Cloud Computing", 2nd National Conference on Information and Communication Technology (NCICT) 2011 Proceedings published in International Journal of Computer Applications® (IJCA) .

[3] Daniel Warneke, Odej Kao "Exploiting Dynamic Resource Allocation for Efficient Parallel
Data Processing in the Cloud" Ieee Transactions On Parallel And Distributed Systems, January 2011

[4] SujitTilak , Prof. DiptiPatil , "A Survey of Various Scheduling Algorithms in Cloud Environment", International Journal of Engineering Inventions ISSN: 2278-7461, www.ijeijournal.com Volume 1, Issue 2 (September 2012) PP: 36-39.

[5] SandeepTayal "Tasks Scheduling optimization for the Cloud Computing Systems", InternationalJournal Of Advanced Engineering Sciences And Technologies Volume No.5, Issue No. 2, 111 - 115