International Journal of Science Engineering and Advance Technology, IJSEAT, Vol 2, Issue 9

ISSN 2321-6905
September-2014

IJSEAT
International Journal of Science Engineering and Advance Technology

Design An Optimal Scheduling Algoritm That Minimize The Cost And Task Comletion Time

,
Shashi prabha chahal,Mrs.anu Sharma
shashi.cs2101@gmail.com,er.anusharma18@gmail.com

ABSTRACT:

Cloud computing has emerged as a popular computing model to support on demand services. It is a style of computing where massively scalable resources are delivered as a service to external customers using Internet technologies. The aim is to make effective utilization of distributed resources and put them together in order to achieve higher throughput. In this paper, a scheduling algorithm is proposed which addresses the major challenges of task scheduling in cloud. It overcomes the problems of high task execution cost, improper resource utilization and improves the task completion time. the proposed algorithm also obtains optimum resource utilization. The proposed model is implemented and tested on simulation toolkit. We analyze the performance of proposed algorithm. The analysis results are also compared with existing scheduling approach used by the simulator. The results validate the correctness of the framework.

Keywords

Cloud Computing. Software as a Service (SaaS), Platform as a Service (Paas) :

1. INTRODUCTION

.

Cloud computing is receiving a great deal of attention, both in publications and among users, from individuals at home to the U.S. government. Yet it is not always clearly defined. Cloud computing is a subscription-based service where you can obtain networked storage space and computer resources. One way to think of cloud computing is to consider your experience with email. Your email client, if it is Yahoo!, Gmail, Hotmail, and so on, takes care of housing all of the hardware and software necessary to support your personal email account. When you want to access your email you open your web browser, go to the email client, and log in. The most important part of the equation is having internet access. Your email is not housed on your physical computer; you access it through an internet connection, and you can access it anywhere. Except instead of accessing just your email, you can choose what information you have access to within the cloud[1].

2. CLOUD SERVICES AND MODELS:

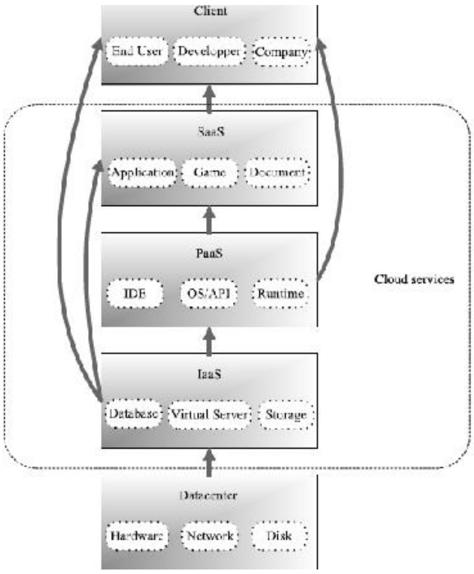
Service means the different types of applications provided by different servers across the cloud. It is generally given as ”as a service”. Services in a cloud are of three types as given [8].

Software as a Service (SaaS) :

In this model, a complete applicationis offered to the customer, as a service. On the customers’ side, there is no need for upfront investment in servers or software licenses, While for the provider, the costs are lowered, since only a single application needs to be hosted & maintained[2].

Platform as a Service (Paas) :

Here, a layer of software or development environment is encapsulated & offered as a service, upon which other higher levels of service can be built. The customer has the freedom to build his own applications, which run on the provider’s infrastructure.



Cloud Services [8]

DEPLOYMENT MODELS:

The NIST definition defines four deployment model [4]:

Public Cloud:
Public cloud services are characterized as being available to clients from a third party service provider via the Internet. The term "public" does not always mean free, even though it can be free or fairly inexpensive to use. Public cloud vendors typically provide an access control mechanism for their users. They provide an elastic, cost effective means to deploy solutions.

Private Cloud: A private cloud offers many of the benefits of a public cloud computing environment, such as being elastic and service based. But in a private cloud-based service, data and processes are managed within the organization without the restrictions of network bandwidth, security exposures and legal requirements. In addition, they offer the provider and the user greater control of the cloud infrastructure, improving security because user access and the networks used are restricted.

Community Cloud:
A community cloud is controlled and used by a group of organizations that have shared interests, such as specific security requirements or a common mission. The members of the community share access to the data and applications in the cloud.

Hybrid Cloud:
A hybrid cloud is a combination of a public and private cloud that interoperates. In this model users typically outsource nonbusiness- critical information and processing to the public cloud, while keeping business-critical services and data in their control.

3.SCHEDULING:

There has been various types of scheduling algorithm exist in distributed computing system. Most of them can be applied in the cloud environment with suitable verifications.

First Come First Serve Algorithm: Job in the queue which come first is served. This algorithm is simple and fast.

Round Robin algorithm: In the round robin scheduling, processes are dispatched in a FIFO manner but are given a limited amount of CPU time called a time-slice or a quantum. If a process does not complete before its CPU-time expires, the CPU is preempted and given to the next process waiting in a queue. The preempted process is then placed at the back of the ready list[3].

Min–Min algorithm: This algorithm chooses small tasks to be executed firstly, which in turn large task delays for long time.

Max – Min algorithm: This algorithm chooses large tasks to be executed firstly,which in turn small task delays for long time.

Most fit task scheduling algorithm: In this algorithm task which fit best in queue are executed first. This algorithm has high failure ratio.

Priority scheduling algorithm: The basic idea is straightforward: each process is assigned a priority, and priority is allowed to run. Equal-Priority processes are scheduled in FCFS order. The shortest-Job-First (SJF) algorithm is a special case of general priority scheduling algorithm[6]. An SJF algorithm is simply a priority algorithm where the priority is the inverse of the (predicted) next CPU burst. That is, the longer the CPU burst, the lower the priority and vice versa. Priority can be defined either internally or externally. Internally defined priorities use some measurable quantities or qualities to compute priority of a process.

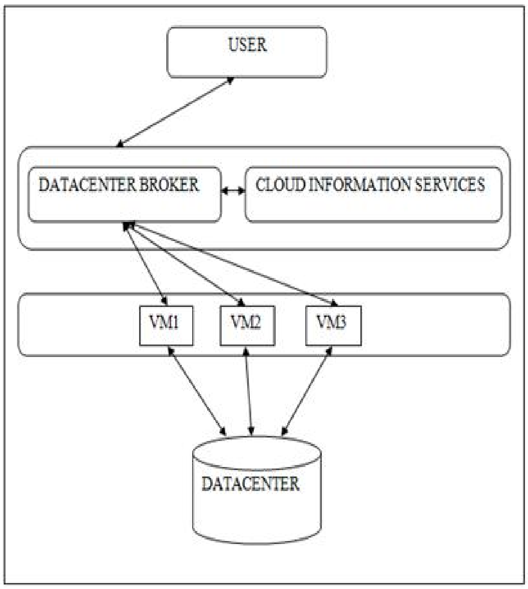
4. SCHEDULING IN CLOUD:

Cloud is a system of high diversity. Scheduling process in cloud can be generalized into three stages namely–

Resource discovering and filtering – Datacenter Broker discovers the resources present in the network system and collects status information related to them[7].

Resource selection according to certain objectives – The target resource is selected based on certain parameters of task and resource.This is the deciding stage.

Task submission – Finally the task is submitted to the resource selected.
The simplified scheduling steps mentioned above are shown in Figure.



Scheduling in Cloud

Resources in a cloud environment can be selected in various ways[8]. The selection of machine can be either random, sequential, according to its processing power or based on any other means. In random selection tasks are submitted to broker which chooses a resource randomly. It is very simple to implement and has less overhead on the scheduler [10]. In the sequential assignment as the tasks are submitted to the broker, the resources are allocated in order in which the request is made. For example if there is a datacenter with two virtual machines (i.e. V0, V1) then task T0 is submitted to V0, T1 to V1, and T2 to V0 and so on. In this algorithm the available resources have equal distribution of tasks[5].
Traditional way for task scheduling in cloud computing tended

to use the direct task of users as the overhead application base. This leads to overpriced tasks [11]. The CloudSim toolkit supports First Come First Serve (FCFS) and round robin (RR) scheduling strategies for internal scheduling of jobs. FCFS and RR suffer from long average waiting time for longer jobs which necessitates for the deployment of a better scheduling strategy at the cluster level.

5. EXISTING SECHDULING ALGORIYHM IN EUCALYPYUS:

Scheduling in Eucalyptus determines the method by which Virtual Machines are allocated to the nodes. This is done to balance the load on all the nodes effectively and to achieve a target quality of service. The need for a good scheduling algorithm arises from the requirement for it to perform multitasking and multiplexing.

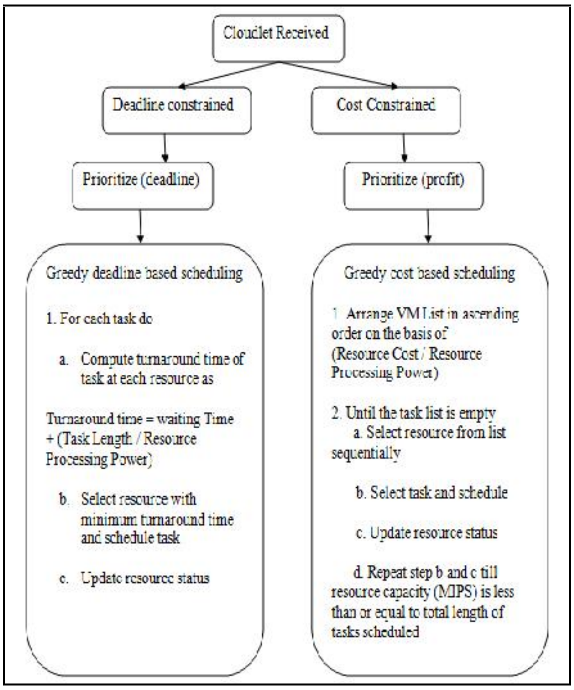
The scheduling algorithm in Eucalyptus is concerned mainly with:

Throughput - number of VMs that are successfully allocated per time unit.

Response time - amount of time it takes from when a request was submitted until the first response is produced.

Fairness / Waiting Time – All the requests for an allocation of a node should be treated in the same manner without any bias.

6. IMPLEMENTATION OF ALGORITHM



CloudSim classes modified for experimentation purpose –

DatacenterBroker - This class models a broker, which is responsible for mediating between users and service providers depending on users' QoS requirements and deploys service tasks across Clouds. The broker acting on behalf of users identifies suitable Cloud service providers through the Cloud Information Service (CIS) and negotiates with them for an allocation of resources that meet QoS needs of users.

In accordance with the proposed algorithm the DatacenterBroker class is modified according to requirements. The modifications done in the class include –

1. The tasks submitted to the broker are checked for their constraint and grouped accordingly in one of the two groups – deadline constrained cloudlet group or cost constrained cloudlet group.

2. Bubble sort is applied to the grouped tasks on deadline or priority parameter and the tasks are rearranged. The new task list is submitted to CIS.

3.The modified BindCloudletToVm function is called depending on the group type of cloudletsBind task according to the greedy approach depending on the cloudlet type and scheduling task accordingly.

4.In BindCloudletToVm(Cost based) function

- i) The VM list retrieved is sorted in descending order of VM cost to processing power ratio.
- ii) Tasks are mapped to same resource and VM status updated until the VM capacity exceeds the limit.
- iii) The count of scheduled cloudlets is updated after each task mapping.
- iv) Next VM is chosen and step 2 is repeated till all the tasks are scheduled.

5.In BindCloudletToVM (deadline based) function

- i) Turnaround time of task at each VM is computed using sum of waiting time and ratio of Task length to VM processing power.
- ii) Task is mapped to VM with minimum turnaround time.
- iii) Waiting time of VM is updated.
- iv) Above steps are repeated till all the tasks are scheduled.

Cloudlet - This class models the Cloud-based application services (content delivery, social networking, business workflow), which are commonly deployed in the data centers. Every application component has a pre-assigned instruction length. The Cloudlet class is modified with deadline and priority as additional parameters.

The cloudlet deadline and priority are added parameters to enable prioritization of cloudlets on their basis. These parameters are beneficial as the user can notify the importance of tasks individually.

Vm - This class models an instance of a VM, whose management during its life cycle is the responsibility of the Host component. A host can simultaneously instantiate multiple VMs and allocate cores based on predefined processor sharing policies. The processor sharing policies are space-shared, time-shared.

The VM class is modified with VM cost as additional parameter. This helps in deciding the target VM when the task execution has the requirement of minimum cost.

VMMAllocationPolicy. This is an abstract class implemented by a Host component that models the policies (space-shared, time-shared) required for allocating processing power to VMs. The functionalities of this class can easily be overridden to accommodate application specific processor sharing policies. Time shared VM allocation policy is used in the experimental setup.

7. RESULT:

The CloudSim toolkit is used to simulate heterogeneous resource environment and the communication environment. CloudSim(2.1.1) simulator is used to verify the results. The experiments are performed with Sequential assignment which is default in CloudSim and the proposed algorithm. The jobs arrival is Uniformly Randomly Distributed to get generalized scenario. The scheduler submits these jobs on available resources according to these algorithms. All parameters are varied in a similar fashion for judging the performance of the two algorithms.

The configuration of datacenter created is as shown below -

Number of processing elements – 1 Number of hosts – 2

Configuration of Hosts

RAM(MB)	10240
Processing Power(MIPS)	110000
VM Scheduling	Time Shared

The configuration of Virtual Machines used in this experiment is as shown in Table:

Configuration of VMs

Virtual Machines	Virtual Machine 1	Virtual Machine 2
RAM(MB)	5024	5024
Processing Power(MIPS)	22000	11000
Processing Element(CPU)	1	1

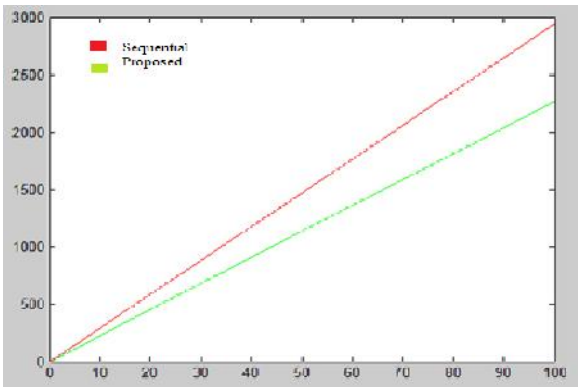
Performance with Respect to Execution Cost

The cost based tasks use greedy approach to minimize the execution cost of individual tasks by selecting the appropriate resource. The tasks execution using the proposed algorithm results in a significant improvement in cost over the sequential allotment as shown in Table . The improvement in cost increases with the increase in number of cloudlets.

Comparison of Execution Cost

No. Of Cloudlets	Proposed Algorithm	Sequential Assignment
25	565.91	735.68
50	1131.82	1471.36
75	1697.73	2207.05
100	2263.6	2942.73

The resulting graph comparing the resultant completion time is shown in figure 5.1 with task on x axis and execution cost on y-axis.



Analysis of Execution Cost

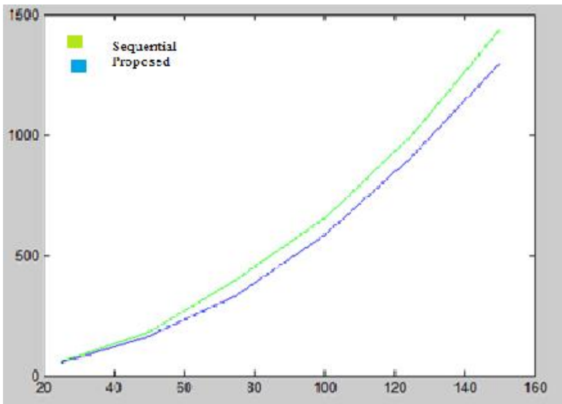
Performance with Respect to Deadline

The experiment is performed repeatedly increasing the number of cloudlets in each trial. It is evident from the results that proposed algorithm gives better completion time of job in comparison to the sequential approach.

Comparison of Completion Time

No. Of Cloudlets	Proposed Algorithm	Sequential Approach
25	52.97	58.97
50	164.66	181.62
75	334.49	399.90
100	584.68	654.03
125	910.04	997.99
150	1298.50	1439.75

The resulting graph comparing the resultant completion time is as shown in figure with task on x axis and completion time on y-axis.



Analysis of Completion Time

8. CONCLUSION:

It is observed that the proposed algorithm improves cost and completion time of tasks as compared to Sequential Assignment. The turnaround time and cost of each job is minimized individually to minimize the average turnaround time and cost of all submitted tasks in a time slot respectively. The results ive with the increase in task count.

9. REFRENCES:

[1] J. Geelan, "Twenty-one experts define cloud computinmprog," Cloud Computing Journal, vol. 2009, pp. 1-5, 2009.

[2] R. Mikkilineni and V. Sarathy. (2009). Cloud Computing and the Lessons from the Past in 18th IEEE International Workshop on Enabling Technologies: Infrastructures for Collaborative Enterprises, WETICE'09,2009.

[3] S. Ma, "A Review on Cloud Computing Development," Journal of Networks, vol. 7, no.2, pp. 305-310, 2012.

[4] P. Mell and T. Grance, "The NIST definition of cloud computing," National Institute of Standards and Technology, vol. 53, no.6, 2009.

[5] I. Foster, Z. Yong, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," 2008, pp. 1-10.

[6] Cloud Computing[Online]: http://en.wikipedia.org/wiki/Cloud_computing

[8] A. Fox and R. Griffith, "Above the clouds: A Berkeley view of cloud computing," Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Rep. UCB/EECS, vol. 28, 2009.

[9] T. J. Velte, et al., Cloud Computing: A Practical Approach: McGraw-Hill Osborne Media, 2009.

[10] V. Hamscher, et al., "Evaluation of job-scheduling strategies for grid computing," Grid Computing—GRID 2000, pp. 191-202, 2000.

[11] Q. Cao, et al., "An optimized algorithm for task scheduling based on activity based costing in cloud computing," 2009, pp. 1-3.