Volume 6, Issue 7, July 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Novel Scheduling Model for Resource Allocation in Cloud Computing

[1]G. Naga Srikanth, [2]Dr. G. Naga Satish, [3]R. Krishnam Raju Indukuri, [4]Dr. P. Suresh Varma
[1]Lecturer, Aditya Degree College, Kakinada, India
[2] Associate Professor, Aditya Engineering College, Surampalem, India
[3, 4] Department of CSE, Adikavi Nannaya University, Rajahmundry, India

Abstract— Cloud computing provides flexible computing services to access resources of variable capabilities and sizes. The goal of cloud computing is to allocate the cloud resources that are truly needed for the cloud user. Scheduling is the process of allocating jobs to available resources. In cloud computing scheduling is an important activity to improve resource utilization. In this paper we developed a new scheduling model for scheduling in cloud computing by considering only single type of Virtual Machine with different sizes (Small VM, Medium VM and Large VM). Total elapsed time is considered as evaluation metric to evaluate the proposed scheduling model.

Keywords— Cloud computing, scheduling, virtual machine

I. INTRODUCTION

Cloud Computing is an emerging model in Information Technology for enabling convenient, on demand network access to a shared pool of configurable computing resources like networks, servers, storage, services and applications. In Cloud Computing the services can be rapidly provisioned and released with minimum monitoring effort or service provider interaction. The services are customized and delivered in a manner similar to traditional utilities such as water, gas and electricity. In such a model, users can access services based on their requirements without regard to where the services are hosted or how they are delivered. Several computing paradigms have promised to deliver this utility computing vision and these include cluster computing, Grid computing, and more recently Cloud computing.

II. OVERVIEW OF JOB SCHEDULING

Job scheduling problem in Cloud is as "Jobs and Resources need to be allocated and scheduled in such a way that Cloud Users can complete their jobs with minimal time (within the deadline) and cost and maximize the user satisfaction and throughput of Cloud Resource Provider". In a broader sense, the user is expected to complete his/her jobs with minimum time and minimum cost. The following are the major performance metrics of Scheduling Algorithm.

Virtualization enables service providers to increase the infrastructures in terms of energy efficiency and scale the cost of virtual resources. Cloud computing uses virtualization and the modern web to dynamically provide resources of various kinds as services which are provisioned electronically. These services should be available in a reliable and scalable way so that multiple consumers can use them either explicitly upon request or simply as and when required. Virtual Machine (VM) is an emulation of the particular computer system. VMware defines a virtual machine as "a representation of a real machine using software that provides an operating environment which can run or host a guest operating system". The virtual machines themselves are simply files stored on a disk that are loaded and run by the virtual machine host, much like a software program running on an operating system. The VM is the virtualized representation of a physical machine that is run and maintained by the virtualization software.

Peter M.M [1] presented application of multi stage scheduling in production management by building a multi-stage scheduling model which describes the infra-structure of the factory, about the production and how the plant is operated. Anshul Rai [2] discussed generalized resource allocation in the cloud computing and presented a novel approach for resource allocation in cloud computing which helps the user to identify the order of jobs in the schedule in simple way and less complexity also.

Zhen Xiao [3] presented a dynamic resource allocation strategy using virtual machines for cloud computing environment. Luiz F. Bittencourt [4] presented the main characteristics of scheduling in cloud computing, and a brief survey of some of the scheduling algorithms used in these systems. Ji Li [5] presented a greedy based job scheduling algorithm in cloud computing by proposing greedy based algorithm in cloud computing. This algorithm selects local optimum when compared with other methods and reduces the completion time of submitted jobs and increases the user satisfaction. Abhishek Gupta, et al. (2015) presented a theoretical comparison of job scheduling algorithms in cloud computing environment.

Makespan of the schedule means when all the jobs on schedule get finished. Very little work is identified in single stage scheduling in cloud computing, that has driven our research work to be focused on two and multi stage scheduling for resource allocation in cloud computing.

III. SINGLE STAGE SCHEDULING

We consider that n number of user requests have to be processed on Single Type of VM with three different sizes of VM's (Small VM, Medium VM and Large VM). Small VM will allot for the user who require small amount of processing power, RAM and storage space (1 CPU, 4 GB RAM, etc.), Medium VM will allot for the user who require small amount of processing power, RAM and storage space (2 CPU, 10 GB RAM), Large VM will allot for the user who require small amount of processing power, RAM and storage space (4 CPU, 20 GB RAM). The Scheduling model for resource allocation is shown in the Fig. 1.

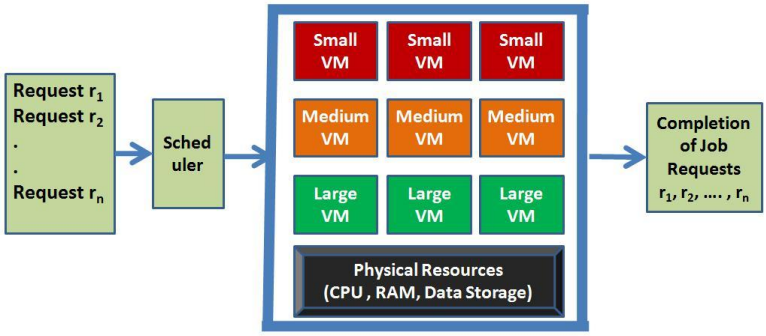

Fig. 1. Scheduling Model for Resource Allocation

User request are selected in order of ascending order of its processing time as similar to Shortest Job First Scheduling. The Scheduling algorithm should meet the following constraints while allocating another size of VM if required size of Virtual Machine is not available. if a user requires Small VM then Scheduler allots an Instance of Small VM if any Small VM is free, otherwise the scheduler will allot Medium VM where Medium VM is free. If Small or Medium VM is not free than the user request is allotted for Large VM. If a user requires Medium VM then Scheduler allots an Instance of Medium VM if any Medium VM is free, otherwise the scheduler will allot Large VM where Large VM is free. If a user requires Large VM then Scheduler allots an instance of Large VM if any Medium VM is free, otherwise the scheduler will allot Large VM where Large VM is free.

The Algorithm-1 describes the single stage scheduling model for resource allocation in cloud computing. This algorithm accepts n number of job request and allocates the required resources in the form of virtual machines. Based on the resources required by the job the scheduler will allocate either Small VM, Medium VM or Large VM. This algorithm identifies optimal allocation of resources to the requested jobs and finally produces the scheduling order with minimum elapsed time of the all schedyles.

Algorithm-1: Pseudo-code for Single Stage Scheduling Model

Input : n number of job requests r_i with processing times
Three types of VM's in sizes Small, Medium and Large
(VM-S, VM-M and VM-L)
p number of instances are available for each type of VM

Output : Optimal scheduling sequence S

```
Initialize VM-S[0..p-1], VM-M[0..p-1], VM-L[0..p-1] as empty;
for each job request r_i with shortest time period among all unprocessed jobs do if r_i requires VM-S Then
  if any of VM-S[j] instance is free then add r_i to the VM-S[j]
  else
    if any of VM-M[j] instance is free then add r_i to the VM-M[j]
    else
      if any of VM-L[j] instance is free then add r_i to the VM-L[j]
      else
        wait until VM-S[j] is free end if;
    end if; end if;
 end if;
 if r_i requires VM-M Then
    if any of VM-M[j] instance is free then add r_i to the VM-M[j]
    else
      if any of VM-L[j] instance is free then add r_i to the VM-L[j]
      else
        wait until VM-M[j] is free end if;
  end if; end if;
 if r_i requires VM-L Then
      if any of VM-L[j] instance is free then add r_i to the VM-L[j]
      else
```


wait until VM-L[j] is free end if;
end if; end for;
Calculate average elapsed time of the entire schedule

IV. RESULT ANALYSIS

Let us consider the 16 number of user requests with three types of VM sizes. Total available Instances of Single Stage Scheduling with Different VM Sizes (Small VM – 3, Medium
– 3, Large VM's – 3) as shown in Table 1..

Table 1. User Request with Processing Time and Type of VM Required for N=16 and P=3

User Request (r_i)	Processing Time (t_i)	Type of VM Req.
0	59	Small
1	86	Medium
2	10	Small
3	90	Small
4	23	Large
5	54	Small
6	49	Small
7	5	Large
8	33	Medium
9	35	Small
10	29	Medium
11	21	Large
12	40	Medium
13	74	Medium
14	25	Small
15	42	Medium

Initially this scheduling model sorts the given job requests in the ascending order of processing times t_i. Now the scheduling sequence will be as follows {r_7, r_2, r_{11}, r_4, r_{14}, r_{10}, r_8, r_9, r_{12}, r_{15}, r_6, r_5, r_0, r_{13}, r_1, r_3}.

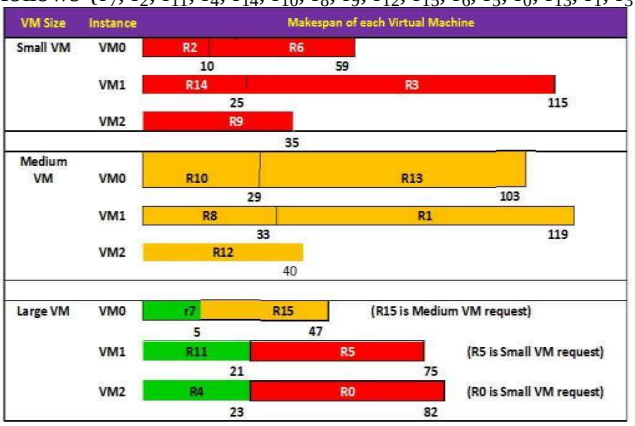

Fig. 1. Scheduling Model for Resource Allocation

Table 2. Average Elapsed Times for N=16 and P=3

Type of VM	Instance-0	Instance-1	Instance-2	Average
Small	58	15	35	69.33
Medium	103	119	40	87.33
Large	47	75	82	68

In the Table 2 elapsed time of individual VM instances are shown. The overall average elapsed time for the entire schedule is 74.88. This mechanism utilizes all available virtual machine instances to reduce the elapsed time of the entire schedule as shown in Fig. 2.

Now we consider n=32 and p=3 as shown in the Table 3, and applied this novel scheduling model by considering total elapsed time of the entire scheduling instance.

Table 3. User Request with Processing Time and Type of VM Required for N=32 and P=3

User Request (r_i)	**Processing Time (t_i)**	**Type of VM Req.**
0	45	0
1	75	2
2	26	1
3	77	2
4	70	0
5	47	2
6	0	2
7	22	0
8	57	0
9	21	1
10	74	2
11	40	0
12	2	1
13	27	2
14	32	0
15	5	1
16	0	2
17	62	2
18	17	2
19	37	2
20	63	1
21	64	1
22	36	0
23	25	2
24	9	1
25	94	0
26	3	1
27	17	2
28	9	1
29	52	2
30	83	1
31	68	1

Table 4. Average Elapsed Times for N=32 And P=3

Type of VM	Instance-0	Instance-1	Instance-2	Average
Small	58	15	35	36.00
Medium	130	85	138	117.67
Large	179	102	115	132.00

The Table 4 shows the elapsed time of each VM instance and how to allocate resources to job with minimum efforts.

V. CONCLUSION

This novel scheduling model has given better performance techniques similar to traditional scheduling models like FCFS and SJF. The average elapsed time of the entire scheduling in this scheduling model for resource allocation in computer science and engineering.

REFERENCES

[1] Peter M.M. Bongers, Bakker B.H, Application of multi-stage scheduling, Proceeding of the 16th European Symposium on Computer Aided Process Engineering, pp 1917-1922, 2006.

[2] Anshul Rai, Ranjita Bhagwan and Saikat Guha, Generalized Resource Allocation for the Cloud, Proceedings of the 3rd ACM Symposium on Cloud Computing, pp 1-12, 2012.

[3] Zhen Xiao, Dynamic Resource Allocation using Virtual Machines for Cloud Computing Environment, IEEE Transaction on Parallel and Distributed Systems, Vol. 24, Issue 6, pp 1107-1117, 2012.

[4] Luiz F. Bittencourt, Edmundo R. M. Madeira, and Nelson L. S. da Fonseca, Scheduling in Hybrid Clouds, IEEE Communications Magazine, pp 42-47, 2012.

[5] Ji Li, Longhua Feng and Shenglong Fangc, An Greedy-Based Job Scheduling Algorithm in Cloud Computing, Journal of Software, Vol. 9, No. 4, pp 921-925, 2014.

[6] Abhishek Gupta, Bhadauria H S, Annapurna Singh and Jagdish Chandra Patni, A Theoretical Comparison of Job Scheduling Algorithms in Cloud Computing Environment, Proceedings of the 1st IEEE International Conference on Next Generation Computing Technologies, pp 16-20, 2015.

