Heuristic Deadline Sensitivity Algorithm

Sumit Dhaka[1], Mrs. Radhika Garg[2]

[1]Dept. of CSE, Vaish College of Engineering, Rohtak
[2]Assistant Professor, Vaish College of Engineering, Rohtak

ABSTRACT

Cloud computing is known as service that allows use of computing, software, platform infrastructure through internet. In Cloud computing user pay for the use of these computing, software etc. In such environment there are plenty of resources at data center but with distributed nature of cloud and less information sharing between client and service provider present task scheduling algorithms become insensitive to task completion before they cross their practical utility, thus it very important to provide efficient utilization of resources and in time delivery of output.

Keyword: Max-min algorithm, min-min Algorithm, RASA, task scheduling, Makespan, deadline sensitivity.

INTRODUCTION

Cloud computing is known as service that allows use of computing, software, platform infrastructure on payment basic through internet. In today`s world cloud computing is growing at a rapid rate. It has also enabled easy access to application and associated data from anywhere around the world. The main purpose of using this technology is to minimize cost and to maximize performance and efficacy. Preparation, timing, and failure management are required to implement the management, scheduling, and responding to demands in minimum time. The main objective of this review paper is to study various scheduling policies for cloud environment and to examine their behavior with respect to deadline. Task scheduling is the basic requirement to make a number of cloud services for an efficient provider while satisfying the client. Task scheduling algorithm is responsible for mapping jobs submitted to cloud environment onto available resource in such a way that total response time and makespan is minimized along with customer satisfaction.

Many task scheduling algorithms are applied by resources manager in cloud computing to optimally allocate resources to tasks .While other scheduling algorithms try to minimize the total completion time. Where the minimization is not necessarily related to the execution time of each single task, but also for minimize overall the completion time of all tasks. But none of the algorithm pays much attention towards the completing task within time limits satisfying user requirement due to lack of information sharing between client and service provider.

Job scheduling is one of the major activities performed in all the computing environments. Cloud computing is one the upcoming latest technology which is developing drastically. To efficiently increase the working of cloud computing environments, job scheduling is one the tasks performed in order to gain maximum profit which in this competitive environment can be achieved with maximum client requirement satisfaction.

CLOUD COMPUTING CHARACTERISTICS

Cloud computing is almost taking advantage of the features that other computing and distributed grids own, but the proper use of the features has made this network superior to the other ones. Cloud computing owns six main characteristics as the following (Mittal and Soni, 2013):

BROAD ACCESS TO NETWORK

Access to cloud resources is possible throughout the network and standard methods are used for the users to access the network.

Supplying Service Based On Demand

Users can have access to their required resources and software without having to interact with cloud computing service providers.

Calculating Service (Pay Per Usage)

One of the key characteristics of cloud computing is calculating system based on the use of services and resources.

Density Of Resources

There are massive amounts of resources in cloud computing which are independent of their physical location via virtualization.

Multiple Users (Tenants) (Shared Resources)

It makes centralizing, increasing the use of unused resources, and sharing resources possible for the users.

Rapid Expansion Ability

Resources in the cloud should be able to expand rapidly and should be unlimited and accessible at any time form the users' point of view.

SERVICE MODELS

The cloud computing service models are Software as a Service (SAAS): In a Software as a Service model, a pre-made application, along with any required software, operating system, hardware, and network are provided. Platform as a Service (PaaS): In PaaS, an operating system, hardware, and network are provided, and the customer installs or develops its own software and applications. Infrastructure as a Service (IAAS): The IAAS model provides just the hardware and network; the customer installs or develops its own operating systems, software and applications.

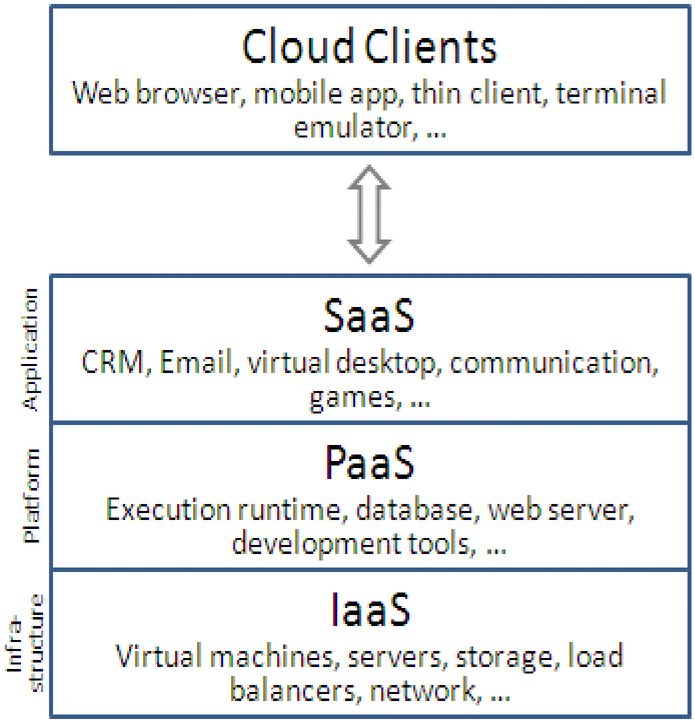

Cloud Clients
Web browser, mobile app, thin client, terminal emulator, ...
Application
SaaS
CRM, Email, virtual desktop, communication, games, ...
Platform
PaaS
Execution runtime, database, web server, development tools, ...
Infra-structure
IaaS
Virtual machines, servers, storage, load balancers, network, ...

Fig. 1

Deployment of Cloud Services:

Cloud services are typically made available via a private cloud, community cloud, public cloud or hybrid cloud.

Public cloud: Generally speaking, services provided by a public cloud are offered over the Internet and are owned and operated by a cloud provider. Some examples include services aimed at the general public, such as online photo storage services, e-mail services, or social networking sites. However, services for enterprises can also be offered in a public cloud.
Private cloud: In a private cloud, the cloud infrastructure is operated solely for a specific organization, and is managed by the organization or a third party.

Community cloud: In a community cloud, the service is shared by several organizations and made available only to those groups. The infrastructure may be owned and operated by the organizations or by a cloud service provider.

Hybrid cloud: A hybrid cloud is a combination of different methods of resource pooling (for example, combining public and community clouds).

NEED FOR SCHEDULING

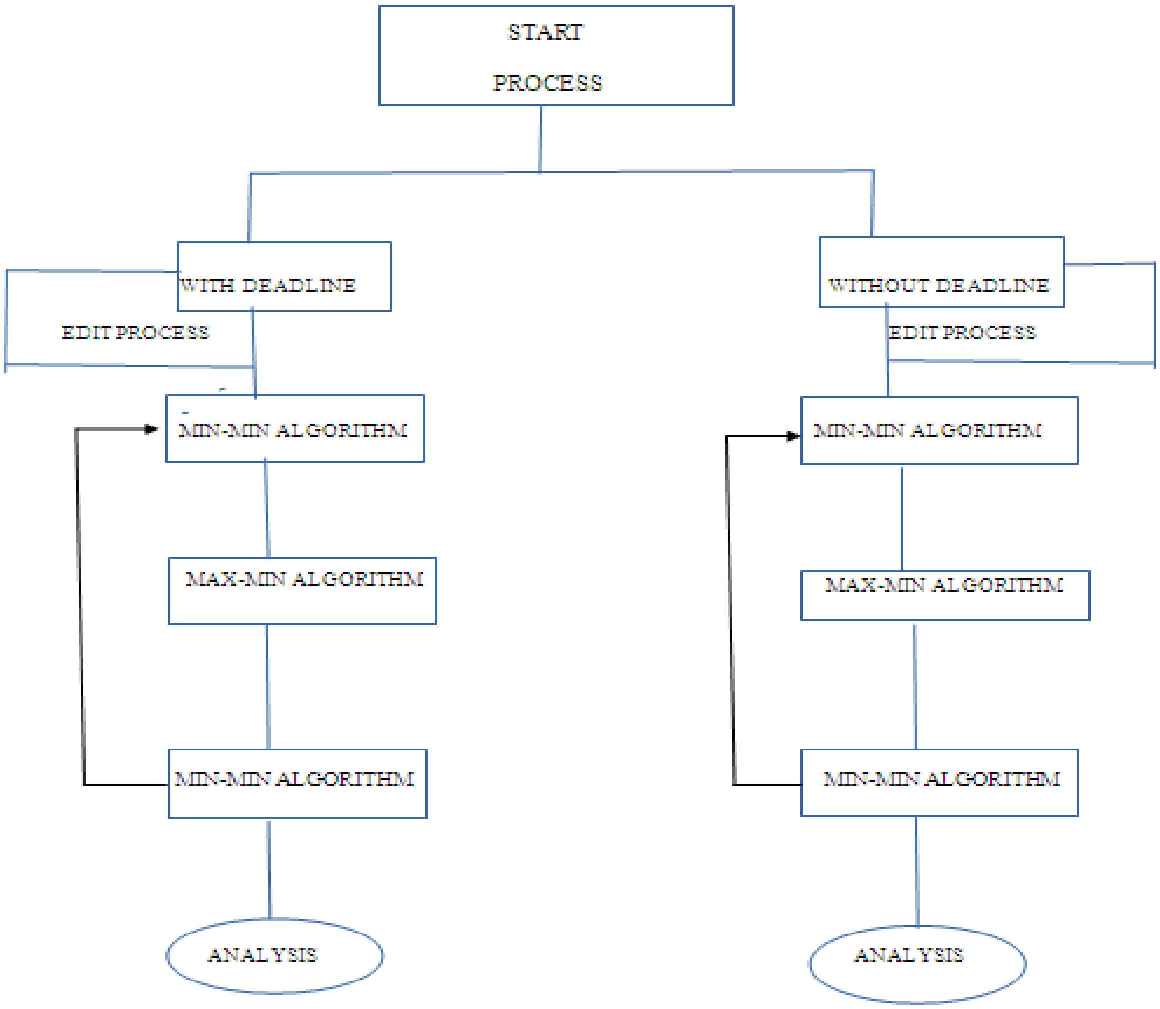
In cloud computing environment, there are plenty of resources at datacenter which are needed to be allocated in efficient and effective manner to get the shortest response time, minimal possible completion time, and utilization of resources.
To achieve all these factors, proper allocation of resources is the key element. There are various existing heuristic algorithms for resource scheduling and allocation in cloud computing. Min –Min and Max-Min are mainly used algorithms, but both of these have their own scope of better resource utilization.

RASA is a new algorithm which uses Max-min and Min-min algorithms. The algorithm determines to select one of these two algorithms, dependent on the standard deviation of the expected completion times of the tasks on each of the resources. RASA algorithm based on static and dynamic environment. But even the use of RASA doesn’t consider the deadline sensitivity of the jobs given to the cloud. All of the running algorithms are an attempt to get the possibly minimal completion time, and the best response time, but none of already existing techniques consider the deadline sensitivity of the job. The jobs to be executed over the cloud can be having different deadlines. Some might be very crucial to be completed before some other task. Some may lose their importance of execution if these are not completed in given time frame.

PROPOSED ALGORITHM

In our proposal we incorporate more information from user making the scheduling client oriented, hence we propose to capture the deadline information from client before we schedule the jobs. This algorithm is an enhancement of RASA algorithm in fields of deadline sensitivity. RASA algorithm which is combination of max-min and min-min algorithm in order to provide maximum makespan and effective utilization of resources fails to consider the deadline before which task has utility for customer.

In our proposal algorithm we proceed in two steps first step is to order the request from user in terms of deadline information provided by client. In the second step we start scheduling by considering a set of 3 request at a time and applying RASA algorithm. Set of 3 jobs is choosen to complete RASA cycle while reducing completion time also to incorporate effect of deadline. So in our work proposal, we consider the deadline of the job and propose a deadline sensitive heuristic algorithm for optimized resource utilization and service execution framework.



ILLUSTRATIVE EXAMPLE

As a simple example, assume there is a cloud environment with three resources. nine tasks T1, T2, T3 ,T4, T5, T6, T7,T8 and T9 with their deadline are in the meta-task Mv and the cloud service provider is required to schedule all the tasks in meta task on three resources M1, M2 and M3. Table 1 represents the task processing time on each resource with their deadline. Applying the data presented in Table 1 , it is possible to calculate the expected completion time of the tasks on each of the resources using already existing RASA algorithm and the proposed sensitivity algorithm and estimating how many task will complete their processing before deadline . The calculated completion time of the tasks are demonstrated in table 2 and table 3 with record of how many task cross there deadline.

TABLE 1

TASK	M1	M2	M3	DEADLINE
1	1	3	5	12
2	2	5	7	8
3	5	8	10	10
4	6	9	9	11
5	7	10	11	17
6	11	11	12	17
7	14	15	16	25
8	15	17	23	32
9	17	19	19	29

TABLE 2

RASA ALGORITHM RESULT:

TASK	COMPLETION TIME	DEADLINE CROSSED
1	1	NO
2	5	NO
3	13	YES
4	24	YES
5	28	YES
6	35	YES
7	28	YES
8	28	NO
9	17	NO

The result of applying RASA on meta-task is exposed in TABLE 2. As shown above, the number of task crossing their deadline after using RASA algorithm is more than 50% i.e. 5 and total execution time is 35 which becomes useless if tasks are not completed within their deadline.

TABLE 3

HEURISITIC DEADLINE SENSITIVE:

TASK	COMPLETION TIME	DEADLINE CROSSED
1	5	NO
2	2	NO
3	8	NO
4	8	NO
5	15	NO
6	17	NO
7	23	NO
8	40	YES
9	32	NO

The result of applying heuristic deadline sensitive algorithm shows that maximum task will complete their processing before their deadline and total completion time is 32 as task 8 having completion time 40 will be disregarded as it crosses its deadline hence the proposed algorithm provides better customer satisfaction and utilization of resources is efficient to fulfill customer need in competitive environment.

CONCLUSION AND FUTURE WORK

RASA is composed of two traditional scheduling algorithms; Max-min and Min-min. RASA uses the advantages of Max-min and Min-min algorithms and covers their disadvantages. The experimental results obtained by applying RASA within the GridSim simulator, shows that RASA is outperforms the existing scheduling algorithms in large scale distributed systems. The above proposed heuristic deadline sensitive algorithm provides better results by satisfying more customer requirements and completing maximum number of task with in time period during which task have utility for the customer . It works on the principle of making cloud environment more client oriented. In our work we have only considered the deadline requirement of client which be extended to many other requirements such as cost , platform specification , and even the work could be extended in fields of green computing making resource allocation energy efficient.

REFERENCES

[1] Yean-Fu Wen and Chih-Lung Chang"Load Balancing Job Assignment for Cluster-Based Cloud Computing", IEEE ,2014
[2] S.DEVIPRIYA "IMPROVED MAX-MIN HEURISTIC MODEL FOR TASK SCHEDULING IN CLOUD"IEEE 2013
[3] M. Malathi, "Cloud Computing Concepts", IEEE, 2011
[4] Paul, M., Sanyal, G., "Survey and analysis of optimal scheduling strategies in cloud environment", IEEE, 2012

[5] Jeyarani, R., Ram, R. Vasanth, Nagaveni, N., "Design and Implementation of an Efficient Two-Level Scheduler for Cloud Computing Environment", IEEE, 2010

[6] Huang Qi-yi, Huang Ting-lei, "An optimistic job scheduling strategy based on QoS for Cloud Computing", IEEE, 2010

[7] Meng Xu, Lizhen Cui, Haiyang Wang, Yanbing Bi, "A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing", IEEE, 2009

[8] Hao Li, Huixi Li, "A Research of Resource Scheduling Strategy for Cloud Computing Based on Pareto Optimality M×N Production Model",, IEEE, 2011

[9] ” RASA: A New Task Scheduling Algorithm in Grid Environment” Saeed Parsa and Reza Entezari-Maleki World Appl. Sci. J., 7 (Special Issue of Computer & IT): 152-160, 2009.