Integration of Deadline and Credits for Cloud Task Scheduling

Pankaj Kumar Chauhan*, Poonam Dabas

Department of Computer Science and Engineering, UIET, KUK, Kurukshetra, India

Abstract: Cloud Computing (CC) is the developing technology tradition for day to day trade operations in today's information technology industries. It provide user with platform, software, attributes of service, virtualization and infrastructure as convenience which is effortless accessible via the internet. In cloud there are infinite tasks demands to be executed by the extant resources to attain the finest performances, achieve deadline, shortest response time etc. Scheduling is a sophisticated task in CC environment where the ambition is to schedule the task completely to reduce turnaround time and improved the overall performance. Recently existing of scheduling algorithm may work proficiently in some perspective. But they are unable to achieve deadline of task and attain maximum profit. In this paper, new mechanism is proposed which take care of deadline as well as assign credit to task based on task length and deadline, then schedule task according to need. Based on the concept of space-shared scheduling policy, this work presents deadline credit based task scheduling. To validate the performance of proposed algorithm we use make-span as criteria to compare with existing algorithm.

Keywords: Cloud Architecture, Cloud Computing, Credit, Deadline, Task Scheduling

I. INTRODUCTION

Cloud computing is trend to provide service as resources including hardware, software, network etc. Every service is provided over network that require high speed of network and persistence connection. Where its services are distributed over the networks according to the architecture and geo-location. It depends on matrices like usability, durability, cost, load etc. So that consumer does not need to buy any hardware, software etc. The main goal of cloud computing is to achieve higher throughput, availability, scalability, consistency guarantees, usability, fault tolerance etc. using distributed resources [1]. Cloud computing resources should able to solve large scale of the computation problems. Cloud computing uses characteristics of Client–server model, Grid computing, Peer-to peer, Mainframe computer, Utility computing to provide better services like gaming, tons of computation, message passing, network etc. Cloud computing has an advantage of delivered a flexible, very high performance, pay-as-you-go, on-demand service. Google adopts Map-Reduce scheduling mechanism scheduling algorithms are relatively simple (First fit etc.). FIFO, default algorithm performs not so well for short jobs. In fact, tasks scheduling in cloud is a NP complement problem with time limit. To improve performance of cloud computing, efficient task scheduling and resource management is required.

A. Cloud Computing

According to the R. Buyya " Collection of the inter-connected and virtualized computers in parallel and distributed computing system" are provisioned dynamically and based on service-level agreement (SLA) and presented as unified computing resources established by negotiation between the service providers of cloud and users is known as Cloud[2].

A.1 Deployment Models of Cloud Computing:

It represents a specific type of the cloud environment, firstly distinguished by ownership, size and access. The basic types of cloud deployment models are defined below:

a) *Private Cloud:* This type of cloud provides its service to department of large organization (single organization) which is managed by either third party or by same organization. It is hosted either internally or externally to consumer organization. Right evaluation can improve business but there can be lot of security issues.

b) *Public Cloud:* This type of cloud is organized by a cloud service selling organization over network to provide service as per pay-as-go model. For example Amazon AWS, Microsoft Azure and Google play are clouds and accessed via internet.

c) *Hybrid Cloud:* This type of cloud is composition of Private community or Public cloud. This cloud is used as per type of service requirements, means to access services a private setup is connected to the public setup.

B. Cloud Computing Architecture

Front and back end are the most significant components of Cloud computing architecture. End which is visible to the user of cloud is front end. It has applications and computer that user uses to access the cloud. Storage devices and computers are the back end of the cloud computing. Figure 1 shows the graphical view of Cloud Computing architecture.

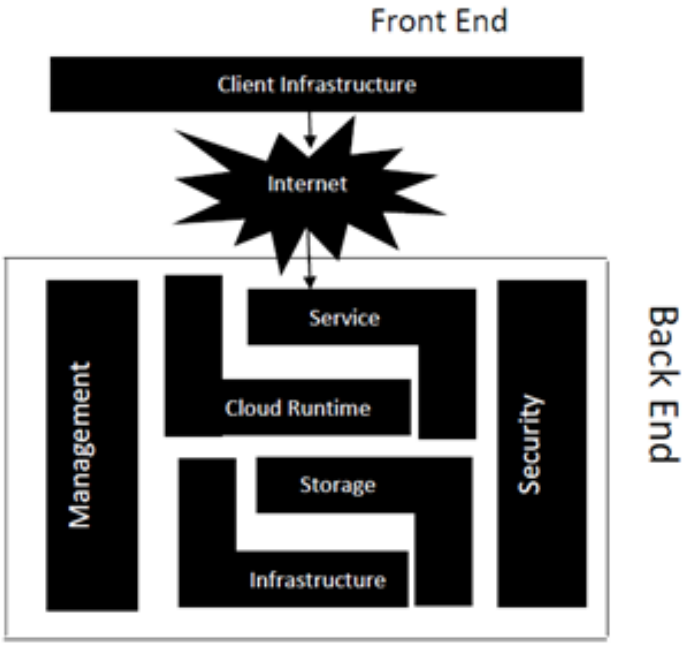

Figure 1: Cloud Computing Architecture

Cloud Architectures address key difficulties related to processing of large amount of data. It is really a hard work to allocate and co-ordinate a large scale job on different-different machines, run procedures on them, and supplies another machine to recuperate if one machine flops during the operation. Applications that built above Cloud Architectures run in-the-cloud where the real

physical location of the substructure is determined by the cloud providers and basically take advantage of modest APIs of Internet-accessible services that are able to scale on-demand, where the difficult logic of scalability and consistency of the services remains executed and hidden inside the huge cloud.

II. TASK SCHEDULING

Task scheduling: It play key roles in the CC system. In CC system the scheduling of the task not will be done on the base of specific benchmark. It will be done under rules as well as regulations which can be described as an argument among provider and user of cloud. To provide best quality of service to the client or user according to argument it is the critical task for the providers at the equivalent time there is huge no. of the tasks running at the side of provider. Scheduling is a method by which data flow and threads are given access to the system resources [3].

A. Task Scheduling Types

Task Scheduling is the method by which threads, processes or data flows are given access to system resources. Scheduling is a essential operating system function, nearby all computer resources are scheduled before use. There are various types of task scheduling which are discussed below [4].

a) *Preemptive Scheduling*: New process selected to run also when an interrupt occur when new processes become ready. Tasks are usually allotted with priorities. At periods it is compulsory to run a definite task that has a advanced priority before another task even though it is running. Consequently, the running task is intermittent for some time and continued later when the priority task has done its execution or implementation.

b) *Non-preemptive Scheduling:* New process is selected to run either a process terminates and explicit system request causes a wait state. In sort in non-preemptive scheduling, a task which are running is executed till completion.

c) *First Come-First Serve Scheduling(FCFS):* By far the simplest and easiest CPU-scheduling algorithm is the first come- first serve scheduling algorithm, By using help of this algorithm, processes are assigned the CPU in the order request it. Basically, there is a single queue of ready or prepared processes. Relative importance of jobs measured or calculated just by arrival time (poor choice). The implementation or execution of the FCFS policy is easily managed with a FIFO queue.

d) *Round Robin Scheduling(RR):* The round robin scheduling algorithm is planned or designed specifically for time-sharing systems or structures. It is related to FCFS scheduling, but pre-emption is additional to switch in the middle of processes.

e) *Priority-Based Scheduling:* A priority is allocated for each process. The prepared list contains an entry for the each processes ordered by its priority. The process at the creation of the list (highest priority) is picked first.

III. RELATED WORK

Cloud computing is an emerging technology, which changes things like resource allocation, data processing, monitoring etc. Scalability, availability and consistency are its keen concepts. Its challenges are like internet speed, data portability and interoperability, heterogeneous environment,

security etc. A number of researches are provided on its challenges to provide efficient way for handling resources. In the review of literature, let's discuss some review papers of task scheduling in Cloud Computing environment.

Zhong and Li [5], proposed schedule constraint which based on Quality of Services constraints, mainly cost, utilization of resources etc. planned the one task hierarchical model for related task scheduling which consider certain application specification in CC. It consider the parallel structure of sub directed acyclic graph, scheduled hierarchical task graph to decomposed associate tasks which reduce the execution cost as well as improve the tasks execution concurrency. To execute all task which are related in the distinct delay-bound which scheduled conceit of the corresponding calculation method, tasks processing capacity.

Bansal et al. [6], analyzed the Cost performance of Quality of Services Driven task scheduling in CC. There were so many parameters which considers in Quality of Services driven like make-span and latency. But the allocation cost of the parameter was not considered in Quality of Services driven the scheduling-algo. Minimized total allocation costs is essential issue in the CC. Cost was calculated Quality of Services driven task scheduling algorithm. This analyzed Quality of Services scheduling algorithm very efficient algorithm parameters except cost introduced.

Chiang et al. [7], proposed that CC a latest pattern for distributing distant computing resources over a networks. A function of cost has been refined in which system cost congestion as well as server start-up all taken into review. An EGC algo. firstly propose the solving constrained of the expansion problems as well as making performances/cost trade-offs in the entire system.

Luo et al. [8], proposed that Deadline guarantee is an important Quality of Services requirement for some critical scientific workflow applications. The resource heterogeneity and the unpredictable workloads make it the difficult for grid systems to provides efficient deadline-guaranty service for those applications. IaaS providers, such as Amazon's EC2, provides the virtualized on the demands computing which can be aggregated to the existing grid resource pool to enhanced the deadline-guaranty of scientific workflow.

Gao et al. [9], proposed the Cloud computing has attracted significant attention due to the increasing demand for low-cost and energy-efficient computing and high performance. Profit maximization for the cloud service provider a key objective in the large-scale, multi-user and heterogeneous environment of a cloud system. In the envisioned cloud environment, users can constructs their own application and services based on the available set of virtual machines, but are relieved from burden of the resource provisioning and task scheduling. Cloud service provider will then exploit the data parallelism in user workloads to creates an energy platform and deadline aware cloud platform.

Thomas et al. [10], proposed that Credit Based Scheduling Algorithm in CC Environment. In order to achieve good services from a cloud, the need for a number of resources arose but cloud providers are limited by the amount of resources have and are thus compelled to strive to maximum utilization. Min-Min algorithm used to reduce the make span of tasks by considering the task length.

Xue et al. [11], main goal was to put forward a task scheduling algorithm in cloud computing with the goal of the maximum load balancing degree, minimum completion time and minimum energy consumption using improved differential evolution algorithm. To improve global search ability in the earlier stage and the local search ability in the later stage, have adopted the adaptive zooming factor mutation strategy and adaptive crossover factor increasing strategy. . In the process of simulation

performed the functional verification of the algorithm and compared with the other representative algorithms.

IV. PROPOSED WORK

It considers 2 parameter: first is Task-Length and second is User-Priorities. Algo. is base upon Deadline-aware-credit –system where each of the task assigns a credit-based on their task-length and priorities.

A. Task-length-credit

CC systems executes the task which has the distinct length. Whenever task is arranged base on increasing order of length firstly the task which have the short length will be occupied at starting of the array and the tasks which having the peaked length will present at the end. Algo. should be take tasks from both the back as well as front gives it a little more strength.

Task priority credit base system based on task-length will work as follows:

1st Step Find the length of the each task (Tleni).

In second Step Calculates average of the tasks length (lenavg).

In 3rd Step it start with calculated difference in the length as regards to the lenavg.

Let firstly tasks will set T1, T2, T3 and so on. Equation second is used to finds difference in the length with avg. length. Our proposed algorithm do not takes the tasks with smaller and larger length. It will be takes the each task from their middle.

$$TLDi = lenavg - Tleni \quad \ldots\ldots\ldots (2)$$

Where TLDi is task-length differences of the task i. After finds differences in the task lengths of each the task, the credits assigns to the each task. In this algo. There are five credits as well as these credits are given to each task for distinct conditions. Before these steps four distinct values found from length array. These Four values forms the condition for assiging the credits. We cannot simply choose four values.

$$value\_1 = high\_len/5 \quad \ldots\ldots\ldots\ldots\ldots (3)$$

$$value\_2 = high\_len/4 \quad \ldots\ldots\ldots\ldots\ldots (4)$$

$$value\_3 = value\_2 + value\_1 \quad \ldots\ldots (5)$$

$$value\_4 = value\_3 + value\_2 \quad \ldots\ldots\ldots (6)$$

Here the high_len is the highest-value of task-lengths. This can be founded by the Pseudo-code which mention down.

For all submitted tasks in the set; Ti

| TLDi = lenavg - Tleni|
If TLDi≤ value_one

then the credit =Five
else
if the value_one <TLDi≤ value_two
then the credit =four
else
if the value_two <TLDi≤ value_three
then the credit =three
if the value_three <TLDi≤ value_four
then the credit =two
else
value_four>TLDi
then the credit =one
End

This algo. adds the credits-base on task-length. Later this step each of the task will be associated with a credit (Credit-Lengthi)

B. Task-priority-credit-based on deadline

Priority of a task can be computed by equation (7). Scheduling algo. Base on the task priorities has problem to handle the tasks with their related priorities. In proposed approach this cannot arise as a problem because even though we are giving the credits to the each tasks base on their priorities. Final scheduling is based on the total-credit which are based on the task-length and its priorities. In this algo. Distinct tasks assigned with distinct credits. The credit value is generated base on the priorities which is assigned to each task. Let us suppose there are ten tasks then there will be ten distinct credits and there will be twenty credits when deal with twenty tasks. The credit value will be change based on priorities that is assigned by the user.

Pseudo-code is presented down:

Assign task priority (max user pay) accord to prediction model:

pi= k*Li/Di (Li-length of task i, Di-deadline of task i, priority of task i)…….. (7)

For the all submitted tasks in set Ti,
then
Find out the task with maxPi=highest- priority (Priority pi)
then
Choosed division-part = (10^numdigits(maxPi))
For the each task with the priority-Tpri
Then
findPri-frac(i)=Tpri /division-factor
set the credit as Pri-frac
End

Main step in above algo. is finding highest priorities numbers. 2nd step is choosing division factor for finding Pri-frac for each tasks. Computed by the dividing-priority value of each tasks with the division-factor of correlative task. And lastly this value (Pri-frac) will be assigned to the each task as the priority-credit.

Total-Crediti= Credit-Lengthi*Credit-Priorityi …… (8)

2 credits calculated separately. Last step in algo. is to find out total-credit-based on task-length as wll as the task priorities. Total-credit is determined by using the equation 8. Lastly task which having the highest credit will be scheduled 1st.

C. Experiment Setup

The performance measurement gathered by using the matlab on a Dell Laptop with Intel® core i3 2.20 GHz 32-bit processor, 4GB RAM and 80GB HDD. The goal of these experiments is to show the ability of proposed task scheduling approach to meet the deadlines in conditions where the other algorithms could not able to meet deadlines. To accomplish this, we recommend following situation. We label identical datacenters composed of hosts. Each processing element has speed of 1000 MIPS. An available bandwidth of 10Gb it/s. A storage capacity of 1TB. A Random Access Memory (RAM) of 2048MB. We generate 100 random cloudlets with length between 1000 and 3000 and 20 virtual machines on host of one datacenter. We have validate out algorithm with Himani et al. proposed algorithm [12].

C.1 Makespan

Makespan is the completion time of the schedule. Minimum make span is considered as best in scheduling. In Figure 2 results indicates that makespan in our approach is minimum compared to existing policy. In the given figure 2, x-axis algorithms and y-axis shows the makespan of the schedule.

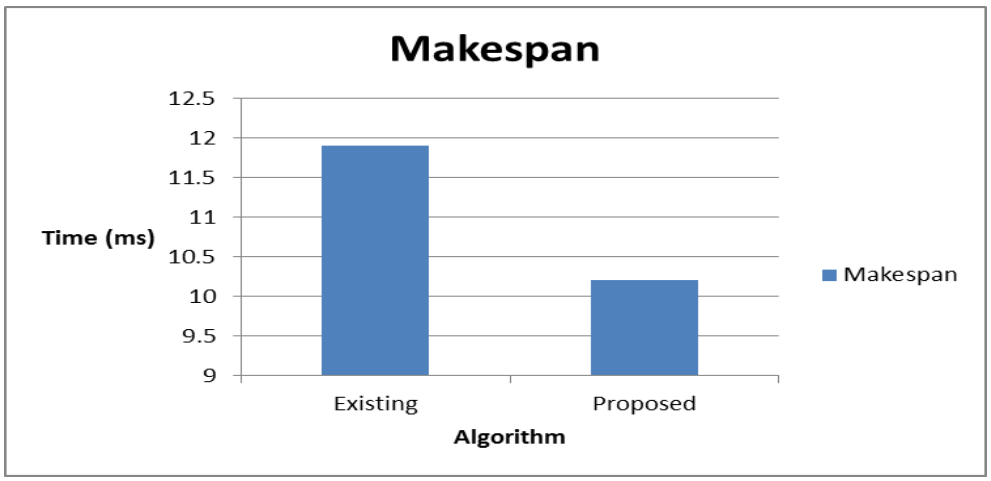

Fig. 2: Comparison of the Makespan

V. CONCLUSION AND FUTURE SCOPE

In this proposed algorithm we have assign a rank for each task to sort them and choose min-min schedule for task schedule. This model give more advantages than existing work in terms of task arrangement based on deadline using composite credit based on task length and deadline of the task. We perform analysis of the proposed model with existing technique to predict the improvement regard make-span, net profit, task penalty and missed deadlines. In future, we can try to optimize the schedule using optimization algorithms like GA, PSO etc.

Conflict of interest: The authors declare that they have no conflict of interest.

Ethical statement: The authors declare that they have followed ethical responsibilities.

REFERENCES

[1] Yashpal Singh Jadeja and Kirit Modi, "Cloud Computing - Concepts, Architecture and Challenges", in the proceedings of 4th International Conference on Computing, Electronics and Electrical Technologies, 2012.

[2] R.Buyya, C.S.Yeo, S.Venugopal, J.Broberg, and I.Brandic, Cloud Computing and emerging IT Platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems, 2009, pp. 599-616.

[3] Tayal, Sandeep. "Tasks scheduling optimization for the Cloud Computing systems." International Journal of Advanced Engineering Sciences and Technologies, Vol. 2, 2011, pp. 111-115.

[4] Kamal Kc, and Kemafor Anyanwu. "Scheduling hadoop jobs to meet deadlines." in the proceedings of IEEE 2nd International Conference on Cloud Computing, 2010, pp. 388-392.

[5] Yingchi Mao, Haishi Zhong and Xiaofang Li Hierarchical Model-based Associate Tasks Scheduling with the Deadline Constraints in the Cloud in the proceedings of IEEE International Conference on Information and Automation (2015) 268-273.

[6] Nidhi Bansal, Amitab Maurya, Tarun Kumar, Manzeet Singh and Shruti Bansal, Cost performance of QoS Driven task scheduling in cloud computing in the proceedings of 3rd IEEE International Conference on Recent Trends in Computing (2015) 126-130.

[7] Chun-ling Cheng, Ying Wang and Jun Li, "An Energy-Saving Task Scheduling Strategy Based on Vacation Queuing Theory in CC" in ISSN Vol.20(1) (2015) 28-39.

[8] Huimin Luo, Chaokun Yan and Zhigang Hu, An Enhanced Workflow Scheduling Strategy for Deadline Guarantee Hybrid Grid Infrastructure in proceedings of Journal of Applied Science and Engineering, Vol. 18(1) (2015) 67-78.

[9] LizhYue Gao, Yanzhi wang, Sandeep K. Gupta and Massoud Pedram "An Energy and Deadline Aware Resource Provisioning, Scheduling and Optimization Framework for Cloud Systems" in the proceedings of IEEE, pp. 1417-1426, 2013.

[10] Antony Thomas, Krishnalal and Jagathy Raj "Credit Based Scheduling Algorithm in Cloud Computing Environment" in the proceedings of International Conference on Information and Communication Technologies, pp. 913-920, 2014.

[11] Jing Xue, Liutao Li, SaiSai Zhao and Litao Jiao "A Study of Task Scheduling Based On Differential Evolution Algorithm in Cloud Computing" in the proceedings of 6th IEEE International Conference on Computational Intelligence and Communication Networks, pp. 637-640, 2014.

[12] Himani and Harmanbir Singh Sidhu, Cost Deadline based task scheduling in Cloud Computing in the proceedings of 2nd International Conference on Advances in Computing and Communication Engineering (2015) 273-279.

[13] Pankaj Kumar Chauhan and Poonam Dabas "A Review of Prioritize Task Scheduling in Cloud Computing" International Journal of Advanced Engineering Research and Applications (IJA-ERA), ISSN-2454-2377,Vol.2(1), pp. 7-13,2016.

