Task Scheduling for Utilization of Resources using Cloud Computing

Shivani
Swami Vivekanand Institute of Engineering & Technology

Harkomalpreet Kaur
Swami Vivekanand Institute of Engineering & Technology

ABSTRACT

Cloud Computing if a form of Internet –based computing that provides shared processing resources and data to computers and other devices on demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., computer networks, server, storage, application and services) which can be rapidly provisioned and released with minimal management effort. Basically, Cloud computing allow the users and enterprises with various capabilities to store and processes their data in either privately owned cloud, or on a third party server in order to make data accessing mechanisms much more easy and reliable. Data Centers that may be located far from the user –ranging in distance from access a city to access the world .Cloud computing relies on sharing of resources to achieve coherence and economy of scale , similar to a utility (like the electricity grid) over an electricity network. In cloud computing, usually there are number of jobs that need to be executed with the available resources to achieve optimal performance, least possible total time for completion, shortest response time, and efficient utilization of resources etc.

Keywords

Cloud Computing, Task Scheduling, Algorithms

1. INTRODUCTION

In computer science, scheduling is the method by which threads, processes or data flows are given access to system resources (e.g. processor time, communications bandwidth). This is usually done to load balance a system effectively or achieve a target quality of service. The need for a scheduling algorithm arises from the requirement for most modern systems to perform multitasking (execute more than one process at a time) and multiplexing (transmit multiple flows simultaneously).

2. SCHEDULING ALGORITHMS

Scheduling disciplines are algorithms used for distributing resources among parties which simultaneously and asynchronously request them. Scheduling disciplines are used in routers (to handle packet traffic) as well as in operating systems (to share CPU time among both threads and processes), disk drives (I/O scheduling), printers (print spooler), most embedded systems, etc. Scheduling algorithms are listed and described below:

2.1 First In, First Out

FIFO is an acronym for First In, First Out, which is an abstraction related to ways of organizing and manipulation of data relative to time and prioritization. This expression describes the principle of a queue processing technique or servicing conflicting demands by ordering process by first-come, first-served (FCFS) behavior: where the persons leave the queue in the order they arrive, or waiting one's turn at a traffic control signal.

2.2 Shortest Job First

Shortest job next (SJN), also known as Shortest Job First (SJF) or Shortest Process Next (SPN), is a scheduling policy that selects the waiting process with the smallest execution time to execute next. SJN is a non-preemptive algorithm. Shortest remaining time is a preemptive variant of SJN.

2.3 Fixed-priority pre-emptive scheduling

Fixed-priority pre-emptive scheduling is a scheduling system commonly used in real-time systems. With fixed priority pre-emptive scheduling, the scheduler ensures that at any given time, the processor executes the highest priority task of all those tasks that are currently ready to execute.

2.4 Round-robin scheduling

Round-robin (RR) is one of the simplest scheduling algorithms for processes in an operating system. As the term is generally used, time slices are assigned to each process in equal portions and in circular order, handling all processes without priority (also known as cyclic executive). Round-robin scheduling is simple, easy to implement, and starvation-free. Round-robin scheduling can also be applied to other scheduling problems, such as data packet scheduling in computer networks.

2.5 Multilevel feedback queue

A multilevel feedback queue is a scheduling algorithm. It is intended to meet the following design requirements for multimode systems:

- Give preference to short jobs.
- Give preference to I/O bound processes.
- Separate processes into categories based on their need for the processor

3. LITERATURE REVIEW

There are many scheduling algorithms available that are used in the cloud computing environment for better scheduling of resources. Scheduling is basically assigning number of resources to the tasks in such a way that there will be maximum resource utilization, minimum waiting time and minimum total processing time.

Liang Luo, at el. [1] , the VM assigns a varying (different) amount of the available processing power to the individual application services. These VMs of different processing powers, the tasks/requests (application services) are assigned or allocated to the most powerful VM and then to the lowest and so on. we have optimized the given performance parameters such as response time and data processing time, giving an efficient VM Load Balancing algorithm i.e. Weighted Active Load Balancing Algorithm in the Cloud Computing environment.

G. Guo-Ning and H. Ting-Lei [2]an optimized algorithm for task scheduling based on genetic simulated annealing algorithm. This considers the QoS requirements like completion time, bandwidth, cost, distance, reliability of different type tasks. Here annealing is implemented after the selection, crossover and mutation, to improve local search ability of genetic algorithm.

Jasmin James, Dr. Bhupendra Verma,[3]] a new VM Load Balancing Algorithm is Weighted Active Monitoring Load Balancing Algorithm using CloudSim tools, for the Datacenter to effectively load balance requests between the available virtual machines assigning a weight, in order to achieve better performance parameters. Here VMs of different processing powers and the tasks/requests are assigned or allocated to the most powerful VM and then to the lowest and so on.

Pawar, C. S., & Wagh, R. B.[4] proposes a priority based dynamic resource allocation in cloud computing. This paper considers the multiple SLA parameter and resource allocation by pre-emption mechanism for high priority task execution can improve the resource utilization in cloud. The main highlight of the paper is that it provides dynamic resource provisioning and attains multiple SLA objectives through priority based scheduling. Since cost is the important aspect in cloud computing.

4. PROPOSED WORK

4.1 Proposed SA Scheduling Algorithm

1. All the tasks are submitted in the cloud.
2. After the tasks are submitted they are sorted in ascending order of the length.
3. After sorting is done groups (let's say combining 3 jobs) are created of the tasks according to the available resources.
4. Calculation of Standard deviation for all the tasks is done according to their respective lengths.
5. Calculation of Mid value for total number of available lengths is done.
6. We can perform allocation of resource to the task in the following manner

   If SD is greater than the value at mid of the total numbers of tasks then tasks in groups with the longer length will be allocated first to the resources in descending MIPS.
   Else the tasks in groups with the minimum length will be allocated first to the resources with the resource in ascending MIPS.

7. All the resources will be deleted at end.

   Method

- In the proposed scheduling algorithm we are dealing with the Shortest Job First algorithm we are just combining two algorithms by which tasks will be executed according to their length. First of all sort all the tasks according to their length.
- Then make their groups according to the requirement. Making group means we are combining the number of cloudlets and sending it to the single resource for the execution.
- Now calculate the Standard deviation of the cloudlets by using the formula.

$$S=\sqrt{\Sigma\,(X-\bar{X})^2}/N$$

Where S is Standard Deviation
$\sum$ Is the "sum of"
X is the each value
$\bar{X}$ is the mean of all values
N is the number of values

- Now if SD is greater than the value at mid of the total numbers of tasks, then the tasks in groups with the longer length is allocated to the resources for execution.

5. RESULT AND DISCUSSION

Cloud user can deploy the large scale application over the real cloud without taking any responsibility for resource management and resource provisioning. CloudSim toolkit provides the modeling and simulation of cloud computing system and application provisioning policy implementation. This Simulation tool provides the repeatable and controlled environment to setup a new virtual cloud computing environment with different cloud component properties. CloudSim Toolkit provides the flexibility to the user to implement his own resource provisioning policy.

CloudSim enables seamless modeling, simulation, and experimenting on Cloud computing infrastructures. It is a self-contained platform that can be used to model datacenters, service brokers, and scheduling and allocation policies of large scale Cloud platforms.

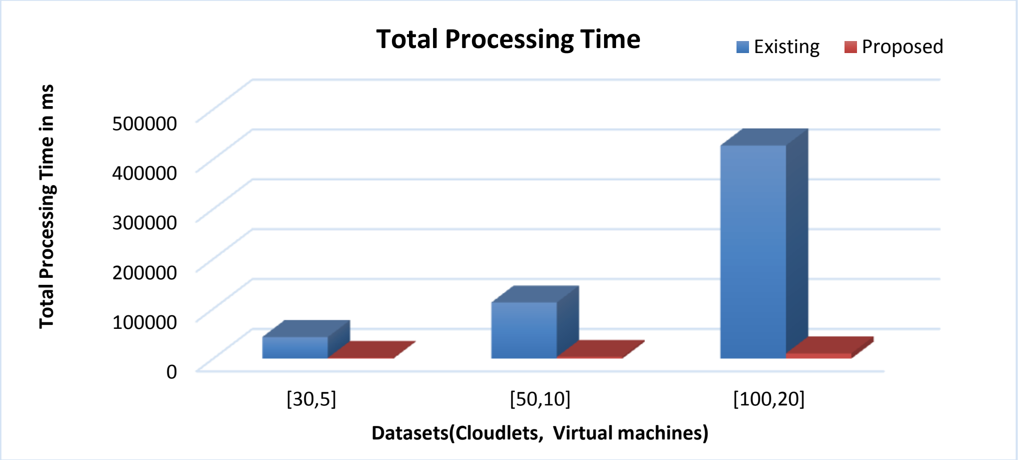

Figure 1: Showing the total processing time comparison of the proposed scheme with the existing on various scenarios

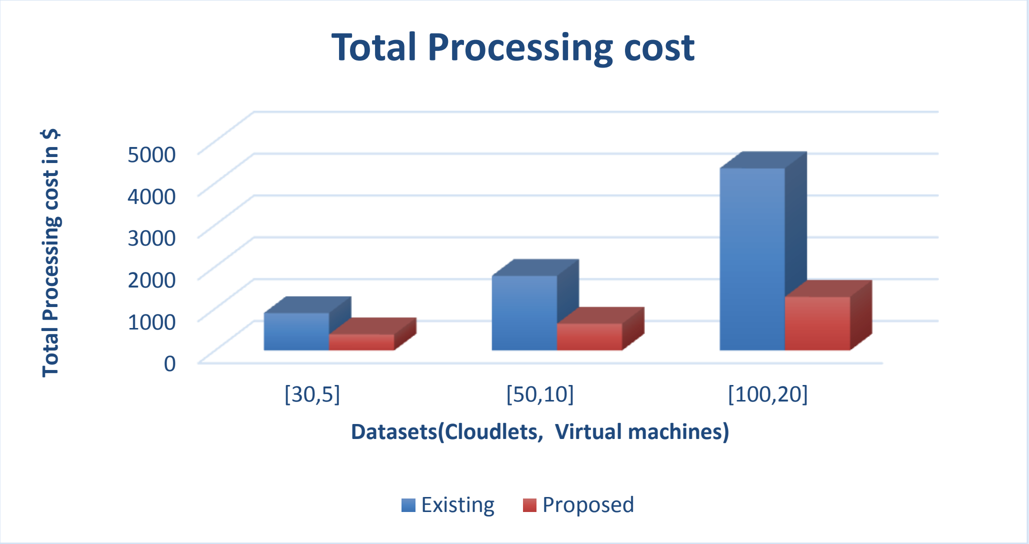

Figure 2: Showing the total processing cost comparison of the proposed scheme with the existing on various scenario

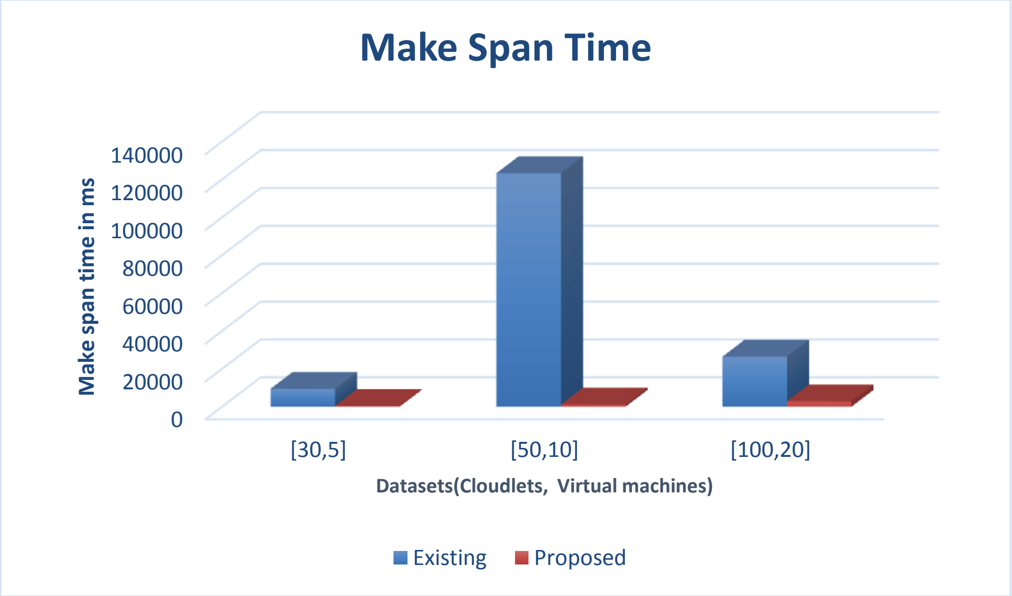

Figure 3: Showing the Make span time comparison of the proposed scheme with the existing on various scenarios

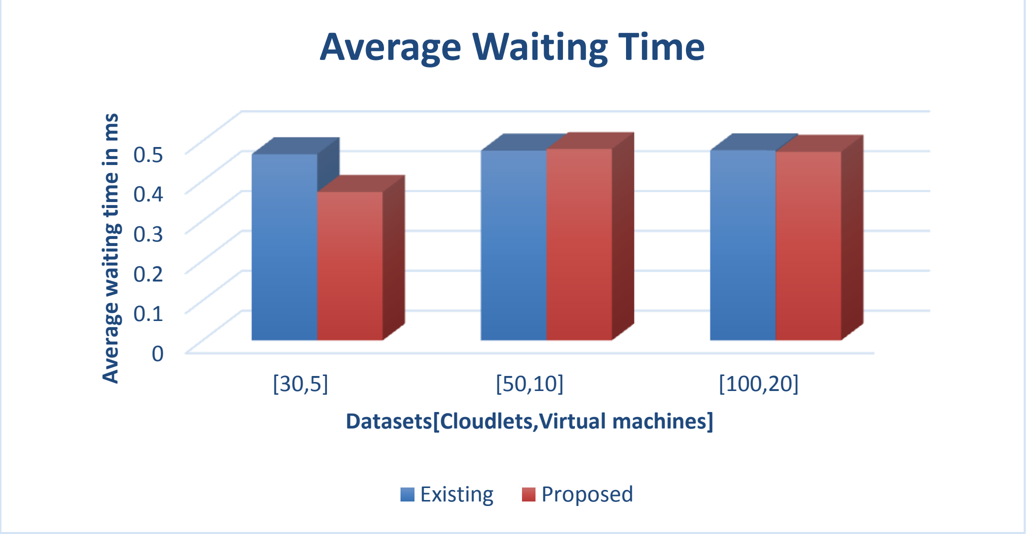

Figure 4: Showing the Average waiting time comparison of the proposed scheme with the existing on various scenarios

From all the above figures it is clearly show that the results of the proposed approach is better than the existing approach with respect to the total processing time, processing cost, average waiting time and make span time.

6. CONCLUSION AND FUTURE SCOPE

The proposed algorithm considers three parameters i.e. Total processing cost, total processing time and average waiting time. Load balancing technique is used to guarantee balancing of load and RASA is used for proper resource utilization. RASA is hybrid approach of Max-Min and Min-Min algorithms. The parameters considered in this research work are tested with various input values and the results depicts that the proposed method achieves better results than the existing method. The work can be further extended in future aiming to achieve more efficient performance results. The proposed work is using RASA for resource allocation. Also, resource allocation can be performed with the improved versions and evolutionary algorithms with implementation in real time scenario.

7. REFERENCES

[1] Tarek Z, Zakria M,.Omara F A, “PSO Optimization algorithm for Task Scheduling on The Cloud Computing Environment”, ISSN 2277-3061, International Journal of Computers And Technology Vol. 13, No. 9

[2] Mathukiya E.S, Gohel P.V, “Efficient Qos Based Tasks Scheduling using Multi-Objective Optimization for Cloud Computing”, International Journal of Innovative Research in Computer and Communication Engineering Vol. 3, Issue 8, August 2015

[3] Lakra. A.V, Yadav. D.K,” Multi-Objective Tasks Scheduling Algorithm for Cloud Computing Throughput Optimization ,” International Conference on Intelligent Computing , Communication & Convergence , 2015 [4]

[4] Brar S. S., Rao S., “Optimizing Workflow Scheduling using Max-Min Algorithm in Cloud Environment”, International Journal of Computer Applications (0975 – 8887)Volume 124 – No.4, August 2015

[5] Bhoi U, Ramanuj P.N, “Enhanced Max-min Task Scheduling Algorithm in Cloud Computing”, International Journal of Application or Innovation in Engineering and Management, Volume 2, Issue 4, April 2013

[6] Kaur N., Kaur K, “Improved Max-Min Scheduling Algorithm”, IOSR Journal of Computer Engineering (IOSR-JCE)e-ISSN: 2278- 0661,p-ISSN: 2278-8727, Volume 17, Issue 3, Ver. 1 (May – Jun. 2015), PP 42-49 Sun H, Chen S.P, Jin C, Guo K, “Research and Simulation of Task Scheduling Algorithm in Cloud Computing”, TELKOMNIKA, Vol.11, No.11, November 2013, pp. 6664~6672e-ISSN: 2087-278X

[7] Chen H, Wang F, Dr Helian N, Akanmu G, “User-Priority Guided Min-Min Scheduling Algorithm for Load Balancing in Cloud Computing”

[8] Etminani K, Naghibzadeh M, “A Min-Min Max-Min selective algorithm for grid task scheduling”, DOI: 10.1109/CANET.2007.4401694 · Source: IEEE Xplore

[9] Liu J, Luo X. G, Zhang X.M, Zhang F and Li B.N, “Job Scheduling Model for Cloud Computing Based on Multi-Objective Genetic Algorithm”, IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 3, January 2013

[10] Pandey S, Wu L, Guru S M, Buyya1R, “A Particle Swarm Optimization-based Heuristic for Scheduling Work flow Applications in Cloud Computing Environments”

[11] Tripathy L, Patra R.R, “Scheduling In Cloud Computing”, International Journal on Cloud Computing: Services and Architecture (IJCCSA) Vol. 4, No. 5, October 2014 [12] Agarwal A, Jain S, “Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment”, International Journal of Computer Trends and Technology (IJCTT) – volume 9 number 7– Mar 2014

[12] Ghanbaria S, Othman M, “A Priority based Job Scheduling Algorithm in Cloud Computing”, International Conference on Advances Science and Contemporary Engineering 2012(ICASCE 2012)