JCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-6, Issue-3 E-ISSN: 2347-2693

Dynamic Scheduling Algorithm With Task Execution Time Estimation Method In Cloud

Babita Rani Radwal[1*], Sanjay Kumar [2]

[1*]Dept. Of CSE, Uttarakhand Technical University, Dehradun , India
[2]Dept Of CSE, Uttarakhand Technical University, Dehradun , India

[*]*Corresponding Author:* babliradwal@gmail.com, *Tel.:*7906244088

Available online at: www.ijcseonline.org

Received: 23/Feb//2018, Revised: 28/Feb2018, Accepted: 21/Mar/2018, Published: 30/Mar/2018

Abstract— In this paper, we consider dynamic scheduling algorithm with task execution time estimation method in cloud which aims to satisfy the workflow deadline by consuming the figure of job performance time prospect and typical nonconformity to approximation real task finishing times.

Existing workflow scheduling algorithms in the grid and cloud background concentrated on a number of QoS parameters such as cost, CPU time, makespan and dependability etc. An Effectual Load Balancing based on Resource Utilization is proposed and related algorithm is executed on CloudSim and its toolkit. The results show the effectiveness and decrease the renting cost of the proposed algorithm. The proposed algorithm improves efficiency and response time compared to delay based dynamic scheduling.

Keywords— Cloud computing, scheduling algorithms, workflow scheduling algorithm.

I. INTRODUCTION

Cloud Computing is a latest mode in today's world. It provides on-demand services like software, infrastructure, platform, hardware, and storage etc. Dynamically to the user according to the "pay per use" model by using virtualized resources over the internet. While Cloud computing provides various services like IaaS, PaaS and SaaS etc[1][2], to end users but due to novelty of cloud computing, it also suffers from many types of research issues such as security, performance, database management, virtual machine migration, server consolidation, fault tolerance and workflow scheduling etc[1][2].

After the development of computer and internet, svarious challenges are arising one by one. One of those challenges is increasing in demand for resource handling and connectivity. Just because growth to a global or worldwide scale of every IT infrastructure dynamic resource and access becomes compulsory. To conquer these problems, a very flexible infrastructure is needed intended managed according to rise or falling off in demand. These factors are responsible for the conception of Cloud. It has been said that "Cloud" is a fresh technology in the generation of computers but its market performance shows a totally different picture.

II. RELATED WORK

Scheduling in cloud computing basically whenever we run our application on any system we want the best performance. So, for the efficient performance of any system, there is need for effective scheduling also. In this section most of the workflow scheduling strategies exist in the grid and cloud background has been reviewed briefly with respect to the technique/algorithm description, scheduling parameter considered, and tools/platform used for the implementation for result analysis. In the year 2003 HE Xiaoshan, Xian-He Sun and Gregor von Laszewski2[3] proposed a scheduling algorithm QoS guided Min-Min Heuristic algorithm in which task under lower and higher bandwidth are scheduled parallel and good load balancing .In 2005 Yu, J., Buyya, R. and Tham, C.K.[4] proposed Cost based scheduling on utility grids which Reschedule the unexpected tasks and can meet user's deadline whilst spending less cost . It can also adapt to the intervals of service executions by rescheduling unexecuted tasks to encounter users' deadlines. In the year 2007 Sakellarious, R[5] work on a scheduling algorithm Workflow with budget constraint which was a DAG scheduling performed . It Minimize the execution time and the makespan. In 2007 K Etminani[6] worked on compilation time and load balancing parameters where no longer waiting of processor for larger task and good load balancing when smaller task are more in number then larger task. In 2008 Yun Yang1, and Ke Liu1[7] worked on Transaction-Intensive Cost Constrained algorithm in Which takes price and time as the main concerns with user input on the fly and incorporates the features of cloud computing. The simulation has demonstrated that our algorithm can attain worse cost than others while meeting the user designated deadline. In 2009 MengXu et al[8] Multiple QoS constrained strategy of multi-workflow which is used to schedule the workflow dynamically and used to and minimize the execution time and cost. In 2010 K. Liu, Y. Yang, J. Chen, X. Liu, D. Yuan and H. Jin[9], proposed Innovative transaction intensive cost-constraint scheduling algorithm which decrease the cost under certain user designated deadlines and Permits the negotiations of performance cost and time and which Measures both resource cost and computation performance. Improve the calculation/communication ratio. In 2011 Zhangjun Wu · Xiao Liu · Zhiwei Ni · Dong Yuan Yun Yang[10] worked on Market oriented hierarchical scheduling strategy in which Overall running cost of cloud workflow system will be minimized. It can be used to optimize both makespan and cost simultaneously. In 2012 Ghanbari, Shamsollah, and Mohamed Othman[11] proposed a priority based job scheduling algorithm used to decrease makespan . In the year 2013 Behzad, Shahram, Reza Fotohi, and Mehdi Effatparvar [12] proposed Queue based job scheduling algorithm used to moderate average waiting and response time . In 2014 Agarwal, Dr, and Saloni Jain [13] worked on generalized priority based algorithms used to moderate execution time. In 2014 Theng, D [14] proposed Heterogeneous factorial self-scheduling used to assign low weighted service node to execute task in lowest execution time. In 2015 Aujla, Sumandeep, and AmandeepUmmat[15] proposed Hybrid cuckoo algorithm used to increase resource utilization and to decrease energy consumption . In 2017 ZhichengCai, Xiaoping Li, Ruben Ruiz, Qianmuli[16] worked on delay based dynamic scheduling where algorithm decreases the resource renting cost while assuring the workflow deadline .

III. METHODOLOGY

In the proposed scheduling algorithms referred as Dynamic scheduling algorithm with task execution time estimation method, several notations have been used which are defined in the following fragment:

Cloudlet_List is defined as the list of all the tasks in workflow.

Size is a variable which holds the sum of tasks in the Cloudlet_List.

vmList is a defined as the list of all virtual machine.

VM_ID is defined as virtual machine number.

The proposed Load Balancing Algorithm based on Resource Utilization description is as follows-

Initially, all the virtual machines have no allocations.

Datacenter Controller receives the user request.

The request is passed to the load balancer.

Load Balancer maintains the table which contains Virtual machine ID, speed of the virtual machine, memory resource and the power consumption of the virtual machine.

Load balancer parses the entire table which contains the hardware configurations include speed, memory and the power consumption.

After parsing the table from top to bottom, it calculates the priority of all the virtual machines from top to bottom.

Priority (i) = $s(i) + m(i) + (1/P(i))$, where $i=1<n$ and Priority lies between $[0,1]$

Where s= speed of CPU

m = memory resource P = Power consumption

n= number of virtual machines

If utilization of vm > threshold, then the process will enter into queue, go to step 2.

The highest priority virtual machine is selected and its ID is returned to the Datacenter controller.

Datacenter processes the request and load balancer updates the table accordingly.

The Data Center Controller checks if there are any waiting requests in the queue. If there are, it continues from step 3.

A. Algorithm

```
Start
Step1. Initialize cloud sim package with number of users, calendar and trace flag
Step2. Create Datacentres
        Create host list
        Create Processing Element(Pe) list1 and list2(for dual core machine)
Step3. Create Broker
Step4. Create VMs and cloudlets and send the list to broker
        Loop till all VMs allocated
        If cloudlet exist then
        initialize VMs with resources
        end if
        end loop
Step5. Create Cloudlet
      Loop till all Cloudlets are initialized with id, number of CPUs, data size, output size, and utilization model set owners of the models (giving IDs)
Step6. Submit VM and Cloudlet list to the broker
Step7. Start simulation
        Balance Load
        If resources required in the table then
        If datavcentre0 table <response_time and cost then
Assign space, Start timer and Update table
```

```
        Otherwise, switch to data centre 1
      If response_time and cost are less then required then
        Assign space
        Start timer
        Update table
        otherwise
        put the request in the Queue
        end if
        end if
        end if
        If timer_request is more than ideal_timer then
        Move the resources to another data centre
         Balance the load again of the task queued
         Update table
         end if
  Step8. Display results to validate the performance
 Stop
```

IV. Results and Discussion

The proposed algorithm is implemented through the simulation software like cloudsim and the cloudsim based toolkit. For the implementation of the application, java language is used. Assuming that the application is deployed in single data center which has a number of virtual machines.

Load balancing based on resource utilization dynamically allocates the cloudlets to a particular VM which is determined by VM"s speed, memory and power consumption. After completing the processing of the request, it calculates the start and finish time of the job. The proposed algorithms solve the problem of load imbalance and results in efficient resource utilization, and reduce total execution time.

A. Response Time Calculation

The response time, RT is calculated with the support of the following formula-

$RT = F_{tm} - A_{tm} + TDelay$

where A_{tm} is the arrival time of user request and F_{tm} is the finish time of user request.

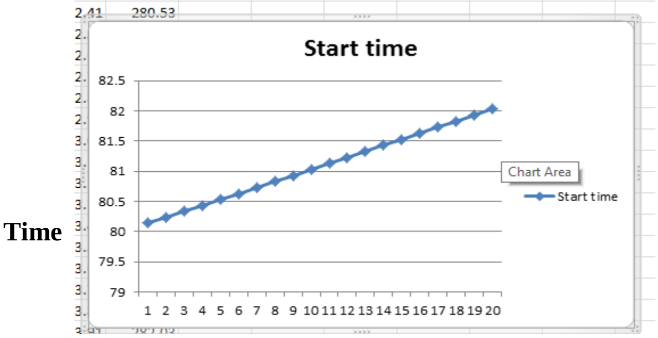
The transmission delay, TDelay is calculated using the following formulas

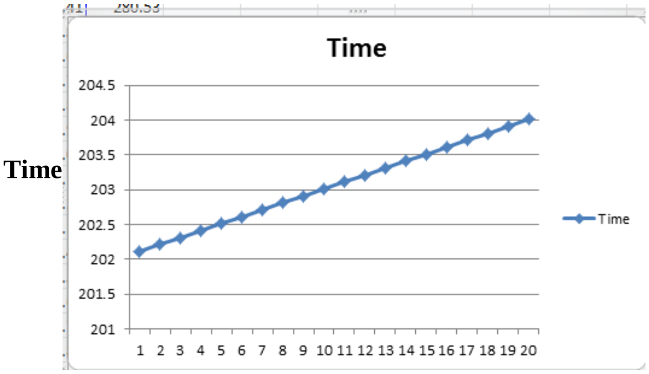
TDelay = Tlatency + Ttransfer

where TDelay is the transmission delay T latency is the network latency.

Ttransfer is the time taken to transfer the size of data from source location to destination of a single request. TDelay is measured to be same in each case and hence it is considered as zero.

B. Result for proposed algorithm





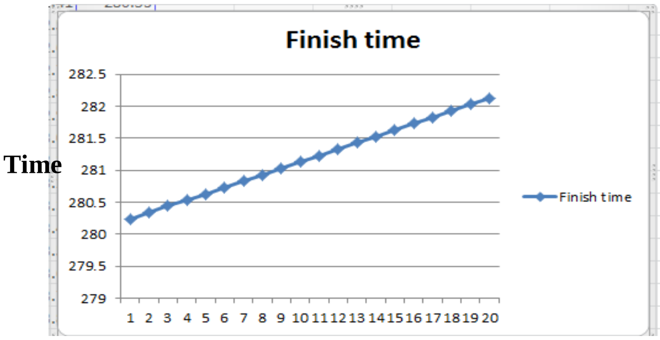


Figure 8.

Figure 9. Graph representations for finish time

C. Results compared with delay based dynamic scheduling:

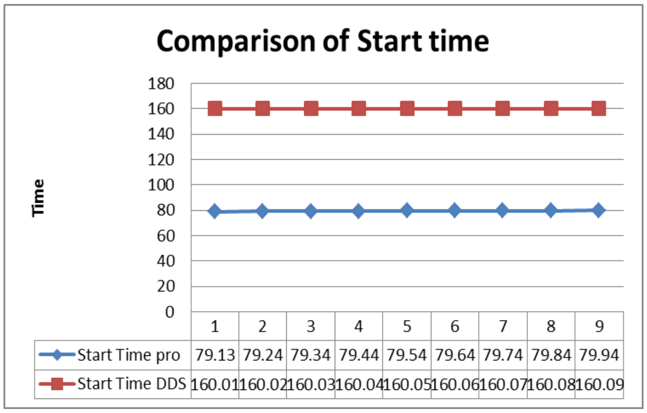


Figure 10. Comparison for start time of DDS and Proposed algorithm

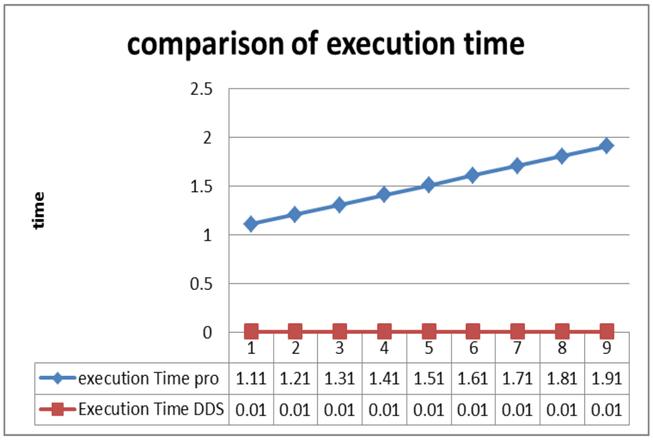


Figure 11. Comparison for Execution time of DDS and Proposed algorithm

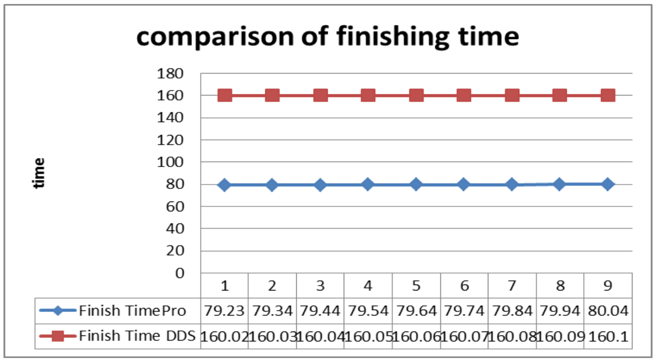


Figure 12. Comparison for finish time of DDS and Proposed algorithm

V. CONCLUSION AND FUTURE SCOPE

Existing scheduling algorithms focused on the time. The main aim of these schedulers is to minimize the overall completion time of the workflow but no analysis was given on the resource utilization. We observe that our algorithm takes minimum time for completion of tasks. The analysis of the results shows that response time of the algorithm is reduced as compared to the other algorithms. In future, we focus to make our algorithms more effective and useful for real time systems.

REFERENCES

[1] P. Mell, and T. Grance, “The NIST Definition of Cloud computing,” National Institute of Standards and Technology, 2009.

[2] Y. Jadeja and K. Modi, “Cloud Computing - Concepts, Architecture and Challenges”, in IEEE proc. of *International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, pp. 877-880, 2012.

[3] HE Xiaoshan, Xian-He Sun, Gregor von Laszewski2 “QoS Guided Min-Min Heuristic for Grid Task Scheduling”(2003).

[4] Yu, J., Buyya, R. and Tham, C.K. ‘‘Cost-based scheduling of scientific workflow applications on utility grids’’, *First Int’l Conference on e-Science* and Grid Computing, Melbourne, Australia, pp. 140–147 (2005).

[5] Sakellariou, R., Zhao, H., Tsiakkouri, E. and Dikaiakos, M.D. ‘‘Scheduling workflows with budget constraints’’, In Integrated Research in GRID Computing, S. Gorlatch and M. Danelutto, Eds Springer- Verlag., pp. 189–202, (2007).

[6] Kobra Etminani, M.Naghibzadeh“A Min-Min Max-Min selective algorihtm for grid task scheduling” in IEEE 3rd *International Conference on Computing, Electronics and Electrical Technologies (ICCEET) in central asia* (2007).

[7] Y. Yang, K. Liu, J. Chen, X. Liu, D. Yuan and H. Jin, An Algorithm in SwinDeW-C for Scheduling Transaction-Intensive Cost-Constrained Cloud Workflows, Proc. of 4th IEEE International Conference on e-Science, 374-375, Indianapolis, USA, December 2008.

[8] M. Xu, L. Cui, H. Wang, Y. Bi, “A multiple QoS constrained scheduling strategy of multiple workflows for cloud computing,” IEEE international symposium on parallel and distributed processing with applications, pp. 629-634, 2009

[9] K. Liu, Y. Yang, J. Chen, X. Liu, D. Yuan and H. Jin, “A Compromised-Time- Cost Scheduling Algorithm in SwinDeW-C for Instance-intensive Cost-Constrained Workflows on Cloud Computing Platform”, International Journal of High Performance Computing Applications, vol.24 no.4 445-456,May,2010.

[10] Z. Wu, X. Liu, Z. Ni, D. Yuan and Y. Yang, “A Market Oriented Hierarchical Scheduling Strategy in Cloud Workflow Systems,” *The Journal of Super Computing*, vol. 63, no. 1, pp. 256-293, Springer US(2011)

[11] Ghanbari, Shamsollah, and Mohamed Othman. "A priority based job scheduling algorithm in cloud computing." Procedia Engineering 50 (2012): 778-785

[12] Behzad, Shahram, Reza Fotohi, and Mehdi Effatparvar. "Queue based Job Scheduling algorithm for Cloud computing." International Research Journal of Applied and Basic Sciences ISSN (2013): 3785-3790.

[13] Agarwal, Dr, and Saloni Jain. "Efficient optimal algorithm of task scheduling in cloud computing environment." arXiv preprint arXiv:1404.2076 (2014).

[14] Theng, D., "Efficient Heterogeneous Computational Strategy For Cross-Cloud Computing Environment" Emerging Research in Computing, Information, Communication and Applications (ERCICA), 2014 Second International Conference on, vol., no., pp.8,17, 1-2 August 2014

[15] Aujla, Sumandeep, and Amandeep Ummat. "Task scheduling in Cloud Using Hybrid Cuckoo Algorithm." International Journal of Computer Networks and Applications (IJCNA) 2.3: 144-15(2015).

[16] Zhicheng Cai, Xiaoping Li, Rubén Ruiz, Qianmu Li "A delay-based dynamic scheduling algorithm for bag-of-task workflows with stochastic task execution times in clouds" Future Generation Computer Systems vol 71, pp. 57–72 (2017)

[17] D. Amalarethinam and F. Selvi, "A minimum makespan grid workflow scheduling algorithm," In *International Conference on Computer Communication and Informatics (ICCCI),* pp. 1-6, IEEE, 2012.

[18] R. N. Calheiros, R. Ranjan, R. Buyya, et al. "Cloudsim: a novel framework for modeling and simulation of cloud computing infrastructures and services", pp. 1-9, 2009.