IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 8, August 2022

DOI: 10.17148/IJARCCE.2022.11816

Performance Evaluation of QOS Parameters of Hybrid TLPD Scheduling algorithm in Cloud Computing Environment

Vijay Mohan Shrimal[1], Prof. (Dr.) Y. C. Bhatt[2] and Prof. (Dr.) Y. S. Shishodia[3]

Ph.D. Research Scholar, Jagannath University, Tonk Road, Jaipur[1]

Emeritus Professor, Jagannath University, Tonk Road, Jaipur[2]

Former Pro_Vice-Chancellor, Jagannath University, Tonk Road, Jaipur[3]

Abstract: In recent years, cloud computing has changed the way that resources are used, allowing users to request resources whenever they need them. The scheduler of cloud computing uses task scheduling and resource allocation algorithms for efficient and effective load balancing of a workload among cloud resources to improve the overall performance of the cloud system when the highly incoming user requests are coming for the resources. But cloud providers are limited by the amount of resources they have, and are thus compelled to strive to maximum utilization. In this paper we have designed the hybrid approach of combination of credit based task length & priority algorithm and credit based deadline algorithm as well as compare the results with FCFS, SJF and task length & priority scheduling algorithms. When we use the credit based task length & priority scheduling algorithm to schedule the task without knowing the deadline of the task, it will cause the dead of the least deadline task. The deadline credit is also included so that assigning number of resources to the tasks in such a way that there will be maximum resource utilization and minimum processing time achieved. This paper presents the simulation results of the proposed methodology implemented with the help of Cloudsim and Net beansIDE8.0 and analysis of results.

Keyword: Task length & Priority, Hybrid TLPD, FCFS, SJF, Cloudsim

I. INTRODUCTION

Cloud computing is a distributed computing environment that provides on demand services to the users for deploying their computational needs in a virtualized environment without the knowledge of technical infrastructure [1].

Resource Allocation strategy (RAS) in the cloud is all about the scheduling of tasks or requests by cloud provider in such a manner to balance the load over all the servers and provide high Quality of Service to clients. It also includes the time required to allocate the resources and the resources available. The main aim is to improve the utilization of resources and complete the entire request within the deadline and with least execution time [1].

In this paper we describe in Section 1 Introduction Section 2 Concept of scheduling Section 3 Simulation Tool Section 4 Related Work Section 5 Traditional task scheduling algorithm Section 6 Proposed Hybrid TLPD Scheduling Algorithm Section 7 QOS parameters Section 8 Simulation Setup Section 9 Results and Analysis and section 10 Conclusion.

II. CONCEPT OF SCHEDULING

The key aim of scheduling is to optimize resource efficiency while minimizing the effect on cloud resources. Currently, cloud computing uses the internet to provide complex services such as software, data, memory, bandwidth, and IT services. The dependability and efficiency of cloud services are influenced by a number of factors, including task scheduling. Scheduling can take place at the job, resource, or workflow stage. Scheduling is done based on a number of criteria in order to maximize overall cloud performance [3].

The main focus of this paper is on cloud resource scheduling. Users request services on demand, and the cloud provider is responsible for allocating the necessary resources to the customer in order to prevent Service Level Agreement violations (SLA). The Task Scheduling process instructs the scheduler to obtain tasks from users and requests information from the cloud information service (CIS) on available resources and their assets. The scheduler schedules user-submitted jobs on different resources according to the availability of resources and the Task Scheduling algorithm. The cloud scheduler is in charge of assigning multiple virtual machines (VMs) to various tasks.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816

III. SIMULATION TOOL

Cloudsim is a simulation program that allows you to conduct cloud computing experiments. CloudSim is a simulation framework that allows for seamless cloud computing and application service modeling, simulation, and experimentation. CloudSim also helps you to model cloud system components like data centres, virtual machines (VMs), and resource allocation policies, as well as their system and behaviour. Cloudsim uses generic device provisioning strategies that are simple to extend and require little effort. The datacenter, which acts as the cloud's backbone and includes a variety of hosts and virtual machines, is depicted in the diagram below(s) [5].

IV. RELATED WORK

Antony Thomas et al, Cloud computing and good service practices are synonymous in the modern world. To provide high-quality cloud services, a variety of resources are needed. However, because there are only so many resources those clouds providers can use, they are motivated to make the most of them. [6].

Mokhtar A. Alworafi et al, The newest distributed computing technique is cloud computing. Service Level Agreement determines the distribution method between the service provider and users (SLA). SLA includes Quality of Service (QoS), which is subject to limitations like deadlines in order to satisfy users. The authors of this article suggest using a Deadline-Aware Priority Scheduling (DAPS) model to enhance resource efficiency while reducing average makespan. [7].

V. TRADITIONAL TASK SCHEDULING ALGORITHMS

(A) First Come First Serve Algorithm: FCFS is a cloud resource-saving scheduling policy that is quick, reliable, and error-free. It employs nonpreemptive scheduling, in which tasks are automatically queued and distributed in response to incoming requests. [8].

(B) Shortest Job First Scheduling Algorithm: Tasks are sorted based on their priority. Priority is given to tasks based on tasks lengths and begins from (smallest task = highest priority). Jobs are queued in order of execution time, with the shortest execution time placed first and the longest execution time placed last and given the lowest priority. [9].

(C) Task Length & Priority Algorithm: The credit based method takes into account two factors: task length and user priority. The credit scheme is used in the algorithm. Each assignment is given a credit depending on the duration and priority of the task. These credits will be taken into account when the job is scheduled. The final step in the algorithm is to find out the total credit based on task length and task priority. Finally task having highest credit will be scheduled first. But this scheduling algorithm based on task length and task priority has the problem of treating tasks with similar priority with similar credits [10].

VI. PROPOSED HYBRID TLPD SCHEDULING ALGORITHM & FLOWCHART

Algorithm Hybrid TLPD:

- Initialize the Cloudsim package by creating the datacenter, broker, virtual machines and cloudlets
- Initialize the virtual machines list
- Initialize the task list.
- Sort the virtual machines using QOS parameters (MIPS and Granulaity size).
- Sort the task list using priorities calculated using credits by using following procedure:
- In this credit to task is assigned using 3 parameters which are credits based on task length, priority of the task, deadline of the task.
-

$Total\_Credit_i$ = $Credit\_Length_i$ * $Credit\_Priority_i$ * $Credit\_deadline_i$

Procedure 1: Credit based on Length of task[8]

For all requested tasks in the set; Ti

Task_length_difference (*TLD*) = absolute_value (average_length – task_length)

If $TLD_i \leq$ value1

then credit =5

else if value1 < $TLD_i \leq$ value2

then credit =4

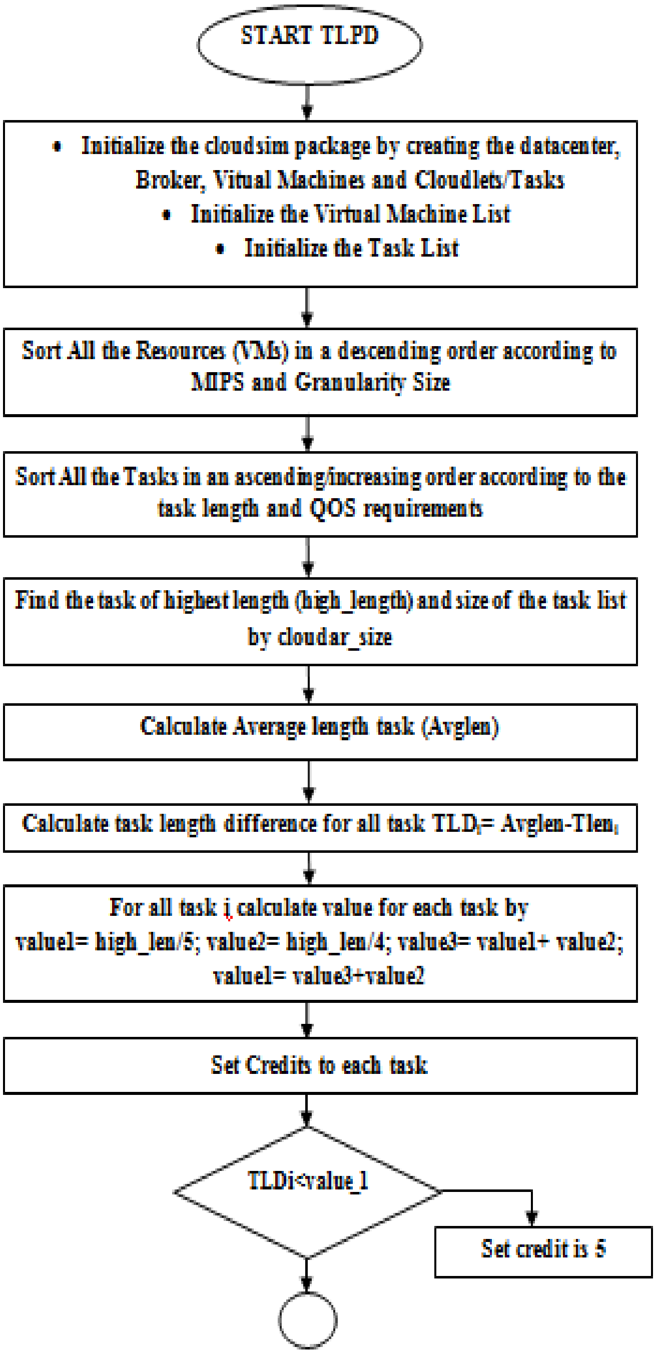
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816

else if value2 < $TLD_i \leq$ value3
then credit =3
else if value3 < $TLD_i \leq$ value4
then credit =2
else value4 > TLD_i
then credit =1
End For
where
value1=high_len / 5;
value2=high_len / 4;
value3=value2+value1;
value4=value3+value2;

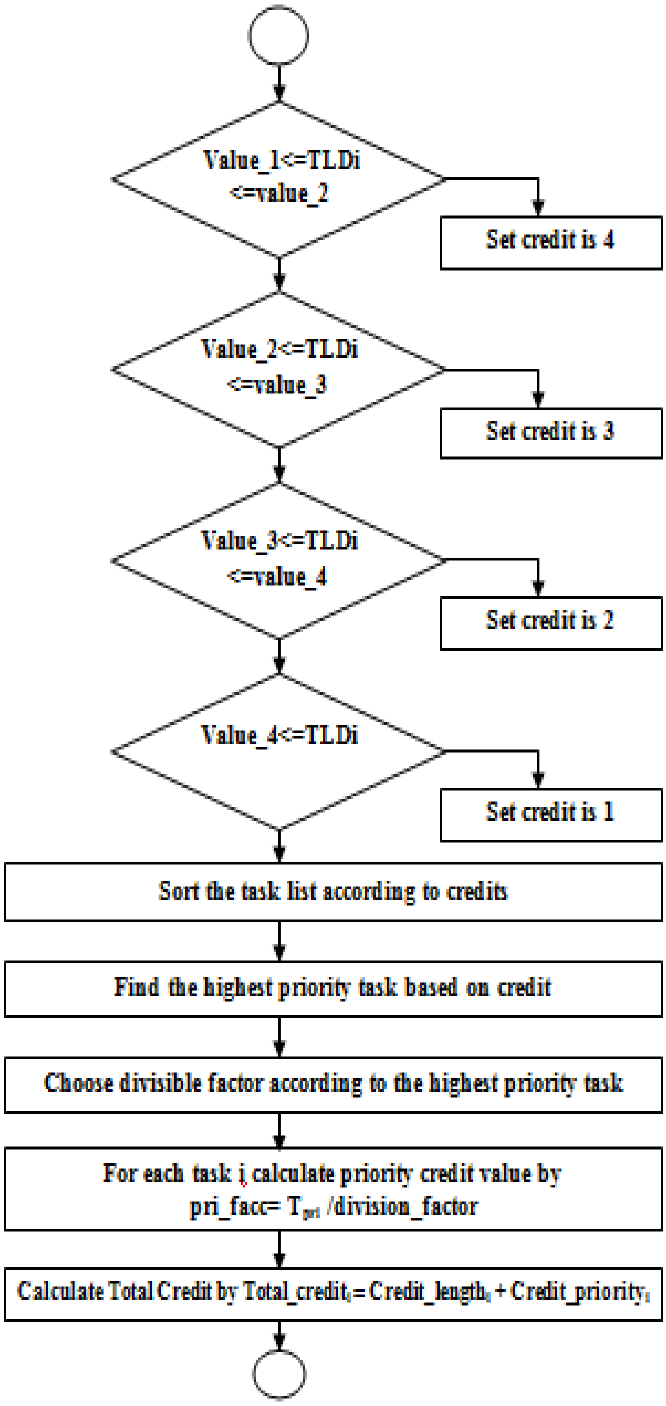
Procedure 2: Priority credits assigning to task[8]
For all requested tasks in the set: Ti
Find out highest priority task (Priority_Number)
Choose division_factor_value
For priority of each task (Tpri)
Calculate Pri_frac_i=Tpri /division_factor
Set priority credit as Pri_frac
End For
End For

Procedure 3: Deadline of the task
For all requested tasks in the set; Ti
Find out MAXMIPS of the VM from the virtual machine list
Deadline_Taski = (Credit_Lengthi * Credit_Priorityi) / MIPSMAX
Calculate $Total\_Credit_i = Credit\_Length_i * Credit\_Priority_i * Credit\_deadline_i$
End For

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816



International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816



International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816

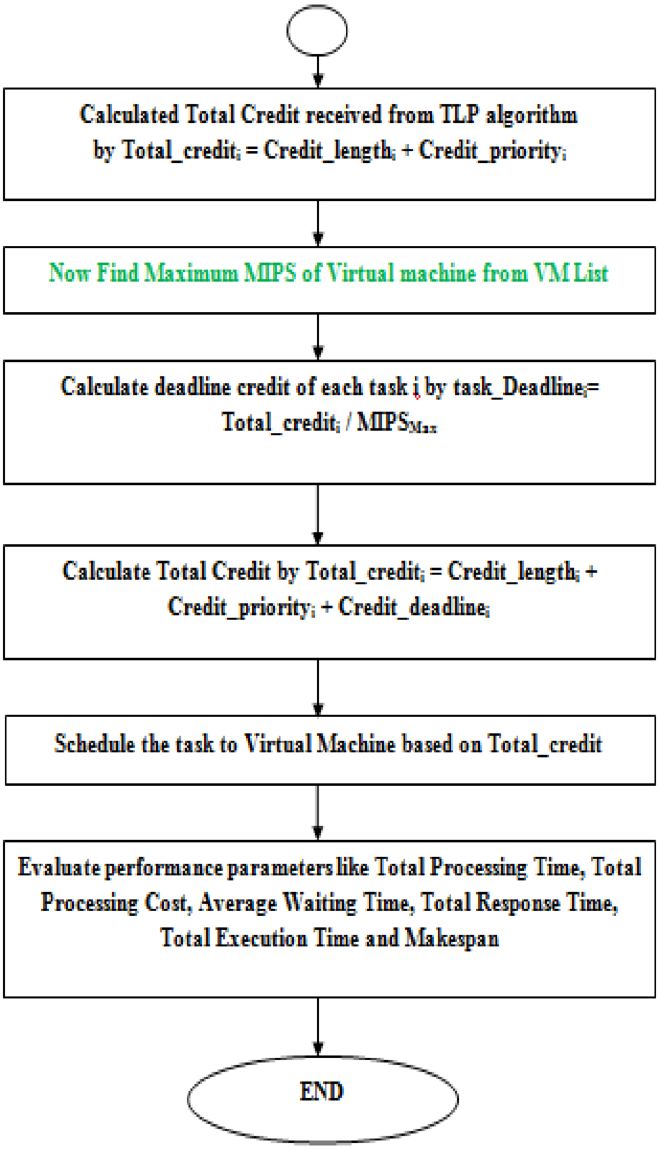


Fig. 1. Flowchart of proposed Hybrid Task length, Priority and Deadline scheduling algorithm (TLPD)

VII. QOS PARAMETERS

In this paper I have considered several metrics/QOS parameters in analyzing the performance of scheduling algorithms. These metrics are as follows.

(a) Total Response Time: The response time of a task refers to the time intervals among tasks to arrive into the system until its completion.

Total Response Time = cloudletSubmissionTime - cloudletFinishTime

(b) Total Execution Time: The CPU time or burst time spent by the computer system for execution of a task is known as execution time, including the time consumed to provide system services for task execution.

Total Execution Time = cloudletExecStartTime - cloudletFinishTime

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⵣ Impact Factor 7.39 ⵣ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816

VIII. SIMULATION SETUP

The configuration of host contains 5 numbers of Hosts, size/processing speed is 5000 (in MIPS), RAM is 5048 (in MB). Configuration of virtual machine contains varying number of virtual machines from 5, 10, 20, 25 and 30 implemented respectively for varying number of cloudlets 30, 50, 100, 150, 200. The details of general simulation parameter are depicted in Table.
Finding Metrics are Total Response Time and Total Execution Time. The experimental data are shown in tables as well as graphs.

TABLE -1 SIMULATION PARAMETER VALUES

S.No.	Parameter	Value
A		
1	Data center architecture	X86
2	Data center OS	Linux
3	VMM	Xen
B		
1	No of Hosts	5
2	MIPS	5000 (in mips)
3	RAM	5048 (in MB)
4	Storage	1000000 (in MB)
5	Bandwidth	500000 (in mbps)
C		
1	No of VMs	5, 10, 20, 25, 30
2	Size/speed of processing	10000 (in mips)
3	MIPS	250 (in mips)
4	RAM	256 (in MB)
5	Bandwidth	1000 (in mbps)
6	No of PEs	1
D		
1	No of Cloudlets	30, 50, 100, 150, 200
2	Length	5000-10000 (in MIs)
3	File Size	100-1000 (in MB)
4	Output Size	300 (in MB)
5	No of PEs	1

IX. RESULT AND ANALYSIS

This section presents the simulation results of the proposed methodology implemented with the help of Cloudsim and Net beansIDE8.0. In this paper, we tested and evaluated the traditional and proposed algorithms using different scenarios where varying number of cloudlets (jobs/tasks) are mapped to varying number of virtual machines (VMs). The performance of the proposed algorithms (TLPD) is evaluated against the traditional algorithm FCFS, SJF and Task Length & Priority and the comparative analysis is described.

1. When number of virtual machines are 5, 10, 20, 25 and 30 and number of cloudlets are 30, 50, 100, 150, 200 assigned respectively. Evaluating Parameter is Total Response Time.

TABLE -2 COMPARISON OF HYBRID TLPD SCHEDULING ALGORITHM WITH TRADITIONAL ALGORITHMS IN DIFFERENT SCENARIOS- EVALUATING PARAMETER TOTAL RESPONSE TIME

Total Response Time				
Dataset	**FCFS**	**SJF**	**Priority**	**Proposed Hybrid TLPD**
[30,5]	**53.339**	**46.987**	**45.0545**	**45.05**
[50,10]	**78.499**	**75.739**	**74.984**	**74.7391**
[100,20]	**152.356**	**152.5779**	**153.1939**	**147.339**
[150,25]	**226.4653**	**225.501**	**224.5271**	**214.952**

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 8, August 2022
DOI: 10.17148/IJARCCE.2022.11816

[200,30]	**336.993**	**302.754**	**302.239**	**297.84**

Table shows the performance analysis of the traditional (FCFS, SJF and Task length & Priority) and proposed scheduling algorithms Hybrid TLPD on the basis of different tasks mapped to different number of virtual machines. The table contains the result value of the parameter "Total Response Time" of the proposed and the traditional scheduling algorithms.

The analysis is done between the available resources (VMs) and requesting task in order to show the scheduling of the task. With the help of resultant values I have designed two types of graphs which represent the different-different perspective of analysis. The performance analysis is further illustrated using two different Line chart and PIE chart graphically:

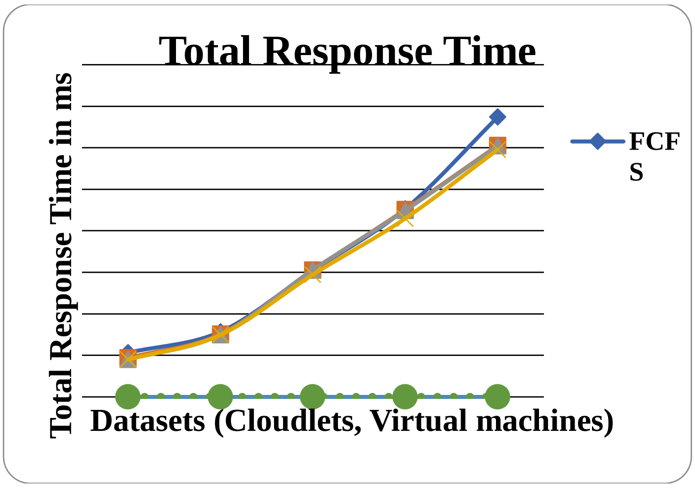


Fig. 2. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Response Time

In this graph we evaluated and analyzed that the total response time is minimum in different scenarios of proposed method compared to traditional methods.

From the analysis of the resultant graph it is cleared shows that the proposed approach performs better result at each steps and evaluated result shows minimum total response time at different scenarios.

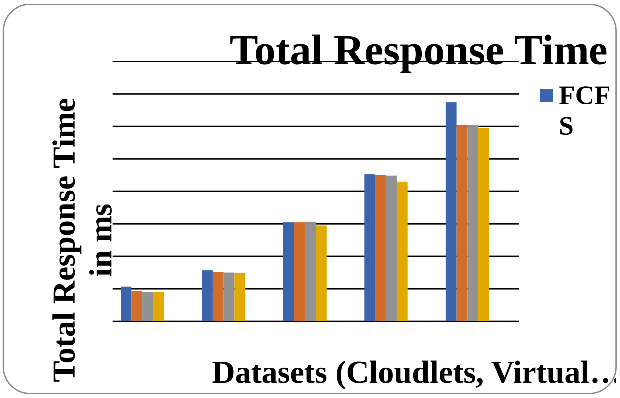


Fig. 3. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Response Time

In this graph we have analyzed that in TLPD scheduling algorithm where we have proposed a hybrid approach with adding the concept of deadline constraints in traditional TLP scheduling algorithm, we found that at starting the total response time of Hybrid TLPD scheduling algorithm is same as traditional algorithms but when task/cloudlets are increased it is minimum total response time compared with TLP and others scheduling algorithm.

2. When number of virtual machines are 5, 10, 20, 25 and 30 and number of cloudlets are 30, 50, 100, 150, 200 assigned respectively. Evaluating Parameter is Total Execution Time.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁘ Impact Factor 7.39 ⁘ Vol. 11, Issue 8, August 2022

DOI: 10.17148/IJARCCE.2022.11816

TABLE -3 COMPARISON OF HYBRID TLPD SCHEDULING ALGORITHM WITH TRADITIONAL ALGORITHMS IN DIFFERENT SCENARIOS- EVALUATING PARAMETER TOTAL EXECUTION TIME

Total Execution Time				
Dataset	**FCFS**	**SJF**	**Priority**	**Proposed Hybrid TLPD**
[30,5]	**46.356**	**44.0826**	**45.578**	**42.615**
[50,10]	**76.786**	**74.998**	**74.755**	**72.58**
[100,20]	**154.521**	**154.635**	**145.0292**	**144.394**
[150,25]	**229.902**	**229.662**	**221.889**	**221.32**
[200,30]	**305.6464**	**302.806**	**302.795**	**300.101**

This table shows the resultant values of the proposed algorithm Hybrid TLPD and traditional algorithms FCFS, SJF and task length & priority. The table contains different datasets of cloudlets and virtual machines.
The performance analysis is further illustrated using two different Line chart and PIE chart graphically:

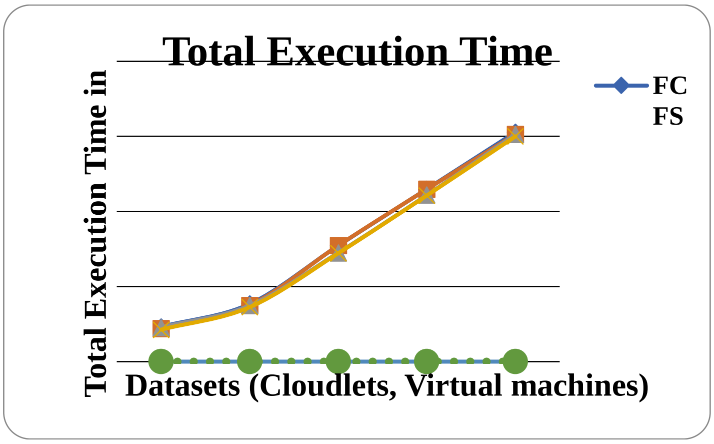


Fig. 4. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Execution Time
In this graph, cloudlets number and virtual machines is represented in the X-axis. In the Y-axis Total Execution Time of cloudlets is represented.
From the analysis of the resultant graph it is cleared shows that the proposed approach performs better result at each steps and evaluated result shows minimum Total Execution Time at different scenarios.

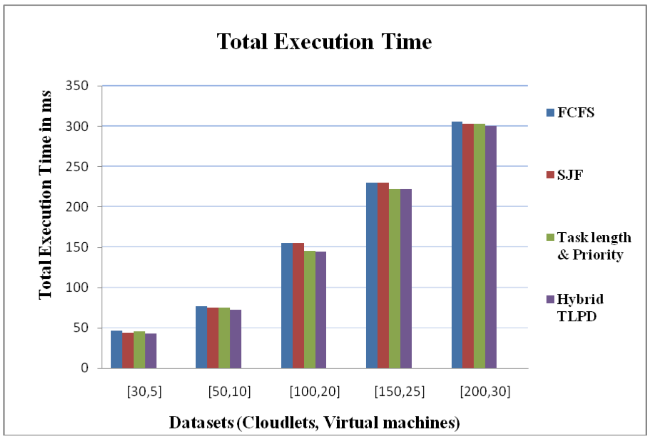


Fig. 5 Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Execution Time

This graph shows the result that the Total Execution Time of proposed algorithm Hybrid TLPD is minimum at different values of datasets compared with the traditional algorithms FCFS, SJF and task length & priority.

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 8, August 2022

DOI: 10.17148/IJARCCE.2022.11816

In this paper, traditional and proposed scheduling algorithms are presented. The traditional algorithms we analyzed the FCFS, SJF and task length & priority in different scenarios. The proposed hybrid approach works on both task length & priority and task deadline (Hybrid TLPD). From the results it is concluded that, the proposed hybrid TLPD algorithm works efficiently than the other traditional methods. Total Response Time and Total Execution Time of the task are lesser when compared with the other algorithms. In future we can add load balancing method for getting more efficient of resources allocation and resources utilization.

ACKNOWLEDGMENT

The Research Scholar is thankful to the Department of Computer science, Jagannath University, Jaipur, Rajasthan, India for providing research facilities and their faculty for being the constant source of inspiration. I would like to thank my Supervisor **Prof. (Dr.) Y.C. Bhatt** and my Joint Supervisor **Prof. (Dr.) Y. S. Shishodia** for his valuable support during the preparation of this paper. I am also thankful to my parents Srimati Nirmala Devi and Shri Mohan Lal Shrimal and my **wife Ms. Neha Shrimal** for their support in my whole work.

REFERENCES

[1]. Rajveer Kaur, Supriya Kinger, "Analysis of Task Scheduling Algorithms in Cloud Computing" International Journal of Computer Trends and Technology (IJCTT) – volume 9 number 7 – Mar 2014 ISSN: 2231-2803 http://www.ijcttjournal.org]
[2]. L. Wang, G. Laszewski, Scientific cloud computing: Early definition and experience, in Proceedings of 10th IEEE International Conference on High Performance Computing and Communications (Dalian, China, 2008), pp. 825–830
[3]. Yogita, C , Bhonsle, M 2012, 'A Study on Scheduling Methods in Cloud Computing', International Journal of Emerging Trends & Technology in Computer Science, vol. 1, no. 3, pp. 12-17.
[4]. Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, César A. F. De Rose, and Rajkumar Buyya CloudSim: A Toolkit for the Modeling and Simulation of Cloud Resource Management and Application Provisioning Techniques" http://dx.doi.org/10.1002/ spe.995
[5]. Prof. S.M. Ranbhise and Prof. K.K.Joshi,"Simulation and Analysis of Cloud Environment", International Journal of Advanced Research in Computer Science & Technology, Vol. 2, Issue 4 (O ct. - Dec. 2014), pp 206-209.
[6]. Syed Arshad Ali and Mansaf Alam, "Resource Aware Min-Min (RAMM) Algorithm for Resource Allocation in Cloud Computing Environment" International Conference on Information and Communication Technologies (ICICT 2014)
[7]. Mokhtar A. Alworafi and Suresha Mallappa," An Enhanced Task Scheduling in Cloud Computing Based on Deadline-Aware Model", International Journal of Grid and High Performance Computing Volume 10 • Issue 1 • January-March 2018
[8]. Sunny Kumar and Shivani Khurana, "Scheduling in Cloud Computing: A Review", International Journal of Advanced Research in Computer Science, Volume 5, No. 1, Jan-Feb 2014 pp 79-81
[9]. Zhang Lufei, Chen Zuoning, "vStarCloud: An Operating System Architecture for Cloud Computing", 2017 the 2nd IEEE International Conference on Cloud Computing and Big Data Analysis, 978-1-5090-4499-3/17/$31.00 ©20 17 IEEE, pp 271-275
[10]. Antony Thomas, Krishnalal G and Jagathy Raj V P, "Credit Based Scheduling Algorithm in Cloud Computing Environment", International Conference on Information and Communication Technologies (ICICT 2014), Procedia Computer Science 46 (2015) 913 – 920

AUTHOR'S BIOGRAPHY

Mr. Vijay Mohan Shrimal is a research scholar of Ph.D. programme of Jagannath University, Jaipur. He completed his Polytechnic Diploma in Computer Science & Engineering from Jodhpur Technical Board, Rajasthan in year 2004 and Bachelor of Engg. in Computer Science from University of Rajasthan in year 2007 and Master degree (M.Tech) from Jaipur National University, Jaipur in year 2012. His area of interests lies in the field of Cloud computing, Machine Learning & Artificial Intelligence, Embedded System and Principles of Programming Languages and many other areas. He has 15 years Academic/Research experience. He is a Member of Computer Science of Teachers Association Collaborate with Association of Computing Machinery, International Association of Computer science and Information technology and International Association of Engineers. He has published 4 patents, 3 books and many research papers in National and International Journals.

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 8, August 2022

DOI: 10.17148/IJARCCE.2022.11816

Prof. (Dr.) Yogesh Chandra Bhatt is an emeritus professor of Jagannath University, Jaipur, Rajasthan. He did his research work at Max Planck Institute of Materials Science Stuttgart Germany 1978-80 and was awarded Doctoral Degree by University of Stuttgart Germany in the year 1980. Professor Bhatt obtained M.Sc. (Physics) degree from University of Rajasthan in year 1965. He was a Former Professor of Physics and Former Dean, Research and Development at Malaviya National Institute of Technology, Jaipur. He was Former Chairman, Board of Governors, Engineering College Bikaner and Engineering College Ajmer. He was also Former Director, Jagannath Gupta Institute of Engg. & Tech., Jaipur. He has guided several Ph.D. He has published many research papers in National and International Journals in India and Abroad.

Prof. (Dr.) Y. S. Shishodia was a Former Pro_Vice-Chancellor of Jagannath University, Jaipur, Rajasthan. Professor Shishodia obtained Bachelor's Degree (B.Sc.) in 1964, Master's Degree in 1966 (M.Sc. Physics) and Doctorate degree in 1972 from University of Rajasthan, Jaipur. He is recipient of Gold Medals from Maharaja College Jaipur for B.Sc and from University of Rajasthan for M.Sc Degree. Prof. Shishodia has also been awarded FIRST PRIZE, by the American Association of Physics Teachers, at the Eleventh Biennial Apparatus Competition held in New York, USA in 1979. Professor Shishodia has been recipient of SAREC (Swedish Agency for Research and Cooperation) Fellowship in 1972-73 and SIDA (Swedish International Development Authority) Fellowship in 1984-85 for post-doctoral work at University of Uppsala, Sweden. He is also a recipient of UNU-ICTP (United Nations University - International Centre for Theoretical Physics) fellowship for post-doctoral work as University of Malaya, Malaysia. Professor Shishodia has been a consultant to SBBJ (State Bank of Bikaner and Jaipur) and Bank of Rajasthan (now ICICI) for their all IT related matters. He has also been a Member (94-2006) of Governing Council of RAJCOMP, which is chaired by Chief Secretary, Govt of Rajasthan.