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 7.918 ⁜ Vol. 11, Issue 12, December 2022

DOI: 10.17148/IJARCCE.2022.111235

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. When the credit based task length & priority scheduling algorithm is used to schedule the task without knowing the deadline of the task, it will cause the dead of the task that having least deadline. In this paper a new hybrid approach (Hybrid TLPD) is designed which is a combination of credit based task length & priority algorithm (TLP) and credit based deadline algorithm. In the new hybrid algorithm, the assigning number of resources to the tasks in such a way that there will be minimum execution time and minimum response time is achieved. With the help of Cloudsim and Net beans IDE8.0 the designed algorithm is simulated and analyzed the results.

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

I. INTRODUCTION

Since last few years, cloud computing has been widely adopted into business institutions, research institutions, industries and in academics. The rapid development of cloud computing has brought a bright prospect and more economic benefits to the commercial industries [1].

Recently, cloud computing has become a turning point of resource computing where resources are provided as services on demand as per the user's request. When the highly incoming user's requests are coming for the resources, the scheduler of the cloud computing uses the task scheduling and resource allocation algorithms for efficient and effective load balancing of a workload among cloud resources to enhance the overall cloud system performance. But cloud providers are limited by the amount of resources they have, and are thus compelled to strive to maximum utilization [2].

In this paper, the description of section as follows: Section 1 Introduction Section 2 Concept of scheduling Section 3 Simulation Tool Section 4 Section Traditional task scheduling algorithm 5 Proposed Hybrid TLPD Scheduling Algorithm Section 6 QOS parameters Section 7 Simulation Setup Section 8 Results and Analysis and Section 9 Conclusion.

II. CONCEPT OF SCHEDULING

To achieve the highest degree of performance and resource efficiency, a set of rules and regulations known as tasks scheduling algorithms are used to allocate tasks to the appropriate resources (CPU, memory, and bandwidth).

Project management is a key component of cloud computing systems, and problems with work scheduling are a significant issue that impacts the system's overall performance. Work scheduling is a mapping method that links user's duties with the gathering and utilization of the necessary resources. Realistic and adaptive project scheduling is used. Tasks and task streams may be scheduled to run at predetermined intervals based on business roles, requirements, and preferences. For routine, weekly, monthly, and yearly operations, project streams and procedures can be established in preparation and executed on-demand without assistance from support personnel [3].

III. SIMULATION TOOL

A simulation application called Cloudsim enables you to do cloud computing experiments. CloudSim is a simulation platform that enables seamless modeling, simulation, and testing of cloud computing and application services. Due to a problem, existing distributed system simulators could not be used in the cloud computing environment. Under

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.918 ⁜ Vol. 11, Issue 12, December 2022
DOI: 10.17148/IJARCCE.2022.111235

various device, user, and requirement combinations, it is challenging to evaluate the performance of cloud provisioning rules, facilities, application workload models, and resource performance models. This issue may be resolved with CloudSim [108]. Additionally, CloudSim assists you in modeling the system and behavior of cloud system components including data centers, virtual machines (VMs), and resource allocation strategies. Cloudsim employs generic device provisioning techniques that are easy to modify and take little time [4].

IV. TRADITIONAL TASK SCHEDULING ALGORITHMS

(i) First Come First Serve Algorithm: A fundamental scheduling method of cloud computing scheduling is FCFS (First Come- First Serve). The first process in the queue is executed first is the guiding principle of FCFS. Water enters at one end and exits at the other, acting as a water supply pipe. Because packets are sent from one end and received from the other end in the same order as they were sent, the same work is completed in the cloud [5].

(ii) Shortest Job First Scheduling Algorithm: According to their importance, tasks are classified. According to task duration, tasks are ranked in order of highest importance to lowest task. A scheduling approach called SJF (Shortest Job First), also known as SJN (Shortest Job Next) or SPN (Shortest Process Next), selects the task with the shortest execution time [6]. Jobs are prioritized according to their execution times, with the highest priority being given to those with the quickest execution times. The CPU is allotted to the task with the lowest burst duration using this scheduling technique, which adopts a distinct strategy.

(iii) Task Length & Priority Algorithm: A different approach considers the work length and user priorities. The algorithm makes advantage of the credit system. A credit is assigned to each assignment based on its length and importance. When the task is designed, these credits will be taken into consideration. Prioritization of tasks is crucial for task scheduling. Multiple tasks may have the same value but have varying priorities, which are represented by values assigned to each task. Tasks with equal priorities are treated similarly by the task priority-based scheduling method [7]. This is not an issue with the suggested method since, even if we are allocating credits to each work based on its priority, the final scheduling will be based on total credit, which is dependent on task duration and its priority.

V.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[7]

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
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;

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 7.918 ∺ Vol. 11, Issue 12, December 2022

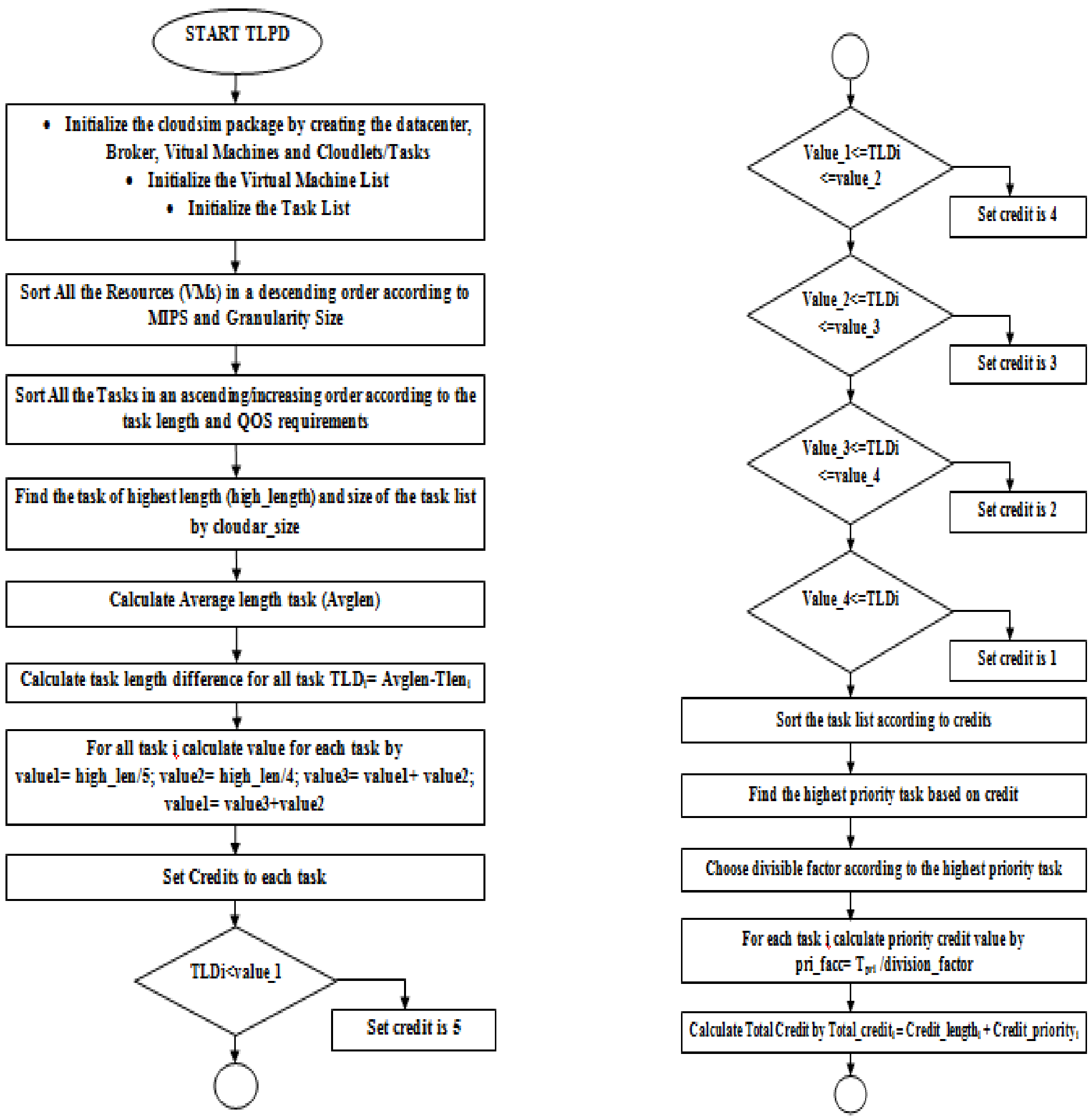
DOI: 10.17148/IJARCCE.2022.111235

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

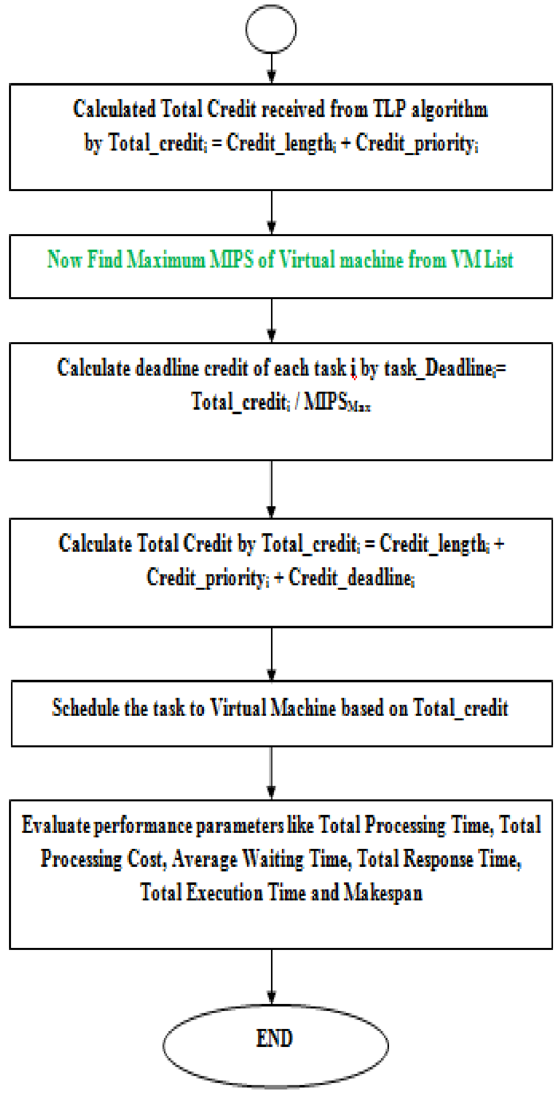


IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ∺ Impact Factor 7.918 ∺ Vol. 11, Issue 12, December 2022

DOI: 10.17148/IJARCCE.2022.111235



VI. 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 [10].

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 [9].

Total Execution Time = cloudletExecStartTime - cloudletFinishTime

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.

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.918 ⁜ Vol. 11, Issue 12, December 2022
DOI: 10.17148/IJARCCE.2022.111235

TABLE -1 SIMULATION PARAMETER VALUES [8]

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

VIII. 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**
[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:

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 7.918 ∺ Vol. 11, Issue 12, December 2022

DOI: 10.17148/IJARCCE.2022.111235

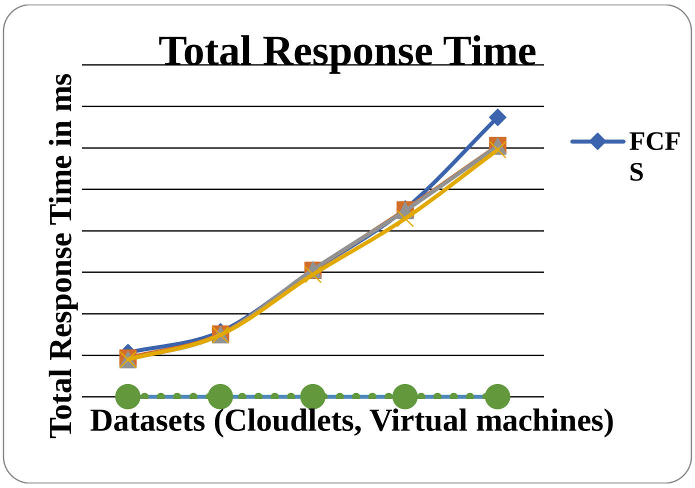


Fig. 1. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Response Time

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.

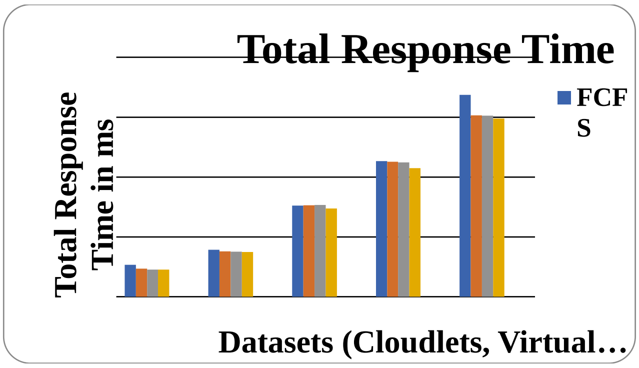


Fig. 2. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Response Time

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.

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.

DOI: 10.17148/IJARCCE.2022.111235

The performance analysis is further illustrated using two different Line chart and PIE chart graphically:

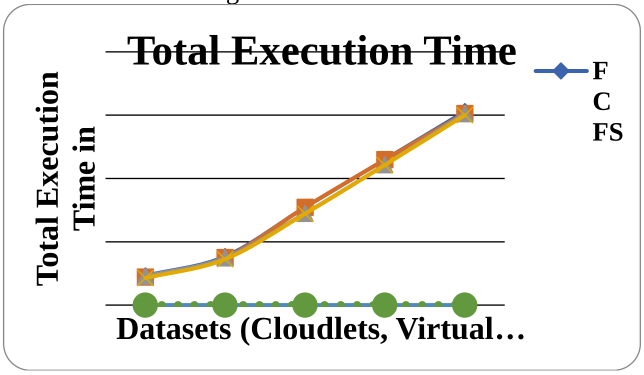


Fig. 3. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Execution Time

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.

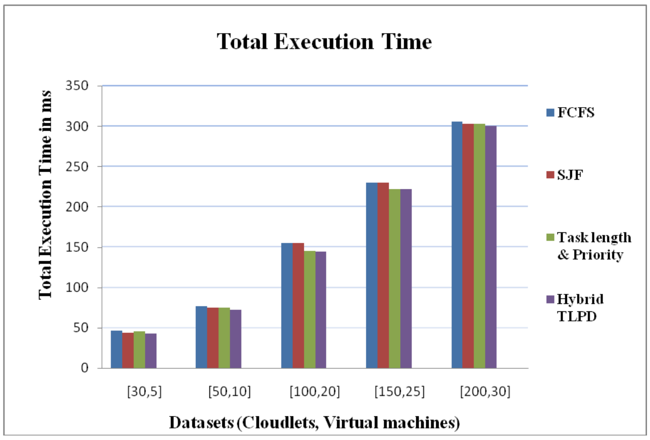


Fig. 4. Comparison of Hybrid TLPD Scheduling Algorithm with Traditional Algorithms – Total Execution Time

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.

IX. CONCLUSION

In this paper, it is concluded that, the proposed hybrid TLPD algorithm works efficiently than the other traditional methods (FCFS, SJF and task length & priority). In Hybrid TLPD scheduling algorithm the QOS parameters Total Response Time and Total Execution Time of the task are lesser when compared with the other traditional 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]. G. Lewis, ―Basics about cloud computing, Software Engineering Institute Carniege Mellon University, Pittsburgh, 2010.
[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]. Zhi Yang, Changqin Yin, Yan Liu. A Cost-based Resource Scheduling Paradigm in Cloud Computing. 2011-12th International Conference on Parallel and Distributed Computing. Applications and Technologies.

[4]. 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.

[5]. 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

[6]. Athokpam Bikramjit Singh, Sathyendra Bhat J., Ragesh Raju, Rio D'Souza," A Comparative Study of Various Scheduling Algorithms in Cloud Computing", 2017 Scientific & Academic Publishing.; 7(2): 28-34

[7]. 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

[8]. 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 ©2017 IEEE, pp 271-275

[9]. 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]

[10]. 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)

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.

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.