e-ISSN : 0976-5166
p-ISSN : 2231-3850

Cloud Task Scheduling using Static Time Quantum Approach

Puneet Sharma

Research Scholar, CEA, GLA University,
Mathura, (UP) 201406, India
puneetsharmasgi@gmail.com
http://<gla.ac.in>

Manoj Kumar

Professor, CEA, GLA University,
Mathura, (UP) 201406, India
manoj.kumar@gla.ac.in
http://<gla.ac.in>

Ashish Sharma

Professor, CEA, GLA University,
Mathura, (UP) 201406, India
Sharma.asb@gmail.com
http://<gla.ac.in>

Abstract

Cloud Computing is the delivery of various services via internet namely hardware, software, servers, databases, network and data storage. Cloud Computing is the technology that allows users to access resources through Internet. Storing data on data cloud is growing more popular among individuals and businesses who need bulk storage space. In our daily life Cloud Computing plays a vital role. Cloud Computing is differentiated from conventional computing by its Scalability, Accessibility, Reliability, and On-demand pay-as-you-go services. At present Cloud Computing technology is used by Google, Amazon, Yahoo. In Cloud Computing Process Scheduling is an important aspect. Minimizing Waiting Time (WT), Minimizing Turnaround Time (TAT), Minimizing Response Time (RT) and Improves Resource Utilization are the important parameter for Task Scheduling. As an outcome various strategies have been suggested and implemented in order to improve scheduling performance. Because of its simplicity and fairness Round Robin (RR) is treated as the best scheduling algorithm.
Our proposed algorithm named Enhanced Round Robin Algorithm for Cloud Scheduling (ERACS). The proposed algorithm reduces Average Waiting Time (AWT), Average Turnaround Time (ATAT), Average Response Time (ART) and Number of Context Switches (NCS). Comparative Analysis is being done with existing algorithms and get better result of our prosed algorithm in terms of AWT (28.2), ATAT (44.2), ART (24.2) and NCS (7).

Keywords: **Cloud Computing; Scheduling; Round Robin; Waiting Time; Turnaround Time; Context Switching.**

1. Introduction

The success of the internet and the advanced technology is defined as Cloud Computing. Cloud Computing introduced new model whose aim is to reshape the world of computing. Cloud consists of parallel and distributed framework which consists of interrelated and virtualized computers. Recently many educational institutions and business organization for storing data and other computing hardware and software resources over the internet. The customer can access information of the cloud through internet by using mobile phones, PDA from anywhere in the world. It allows the usage of servers, networks and applications anytime safely, securely and at low cost. [1]
The use of Cloud Computing brings many advantages that includes unlimited storage capacity, lower cost of setting infrastructure, backup and restore data, multi-tenancy, service in pay as per use basis, unlimited storage capacity, provisioning of resources on demand, data security, excellent accessibility, low hardware and software maintenance cost, scalability, cost effective, improved performance and remote access to resources. Along with

various advantages Cloud Computing suffers from various challenges as it internet connection to access data from cloud, migrating data from one cloud platform to another is also time consuming, designing, developing and maintaining cloud system is quite complicated. Service and Deployment models are shown in Fig 1. There are three prominent services models namely, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS). IaaS provides computational resources whenever it is required. PaaS provides a framework that developers use to quickly and easily build application. SaaS is the software in which the applications are hosted by cloud service provider, user can access this application with the help of internet connectionist works on private and hybrid cloud. Private cloud provides services to the organization. The combination of public and private cloud is hybrid cloud.

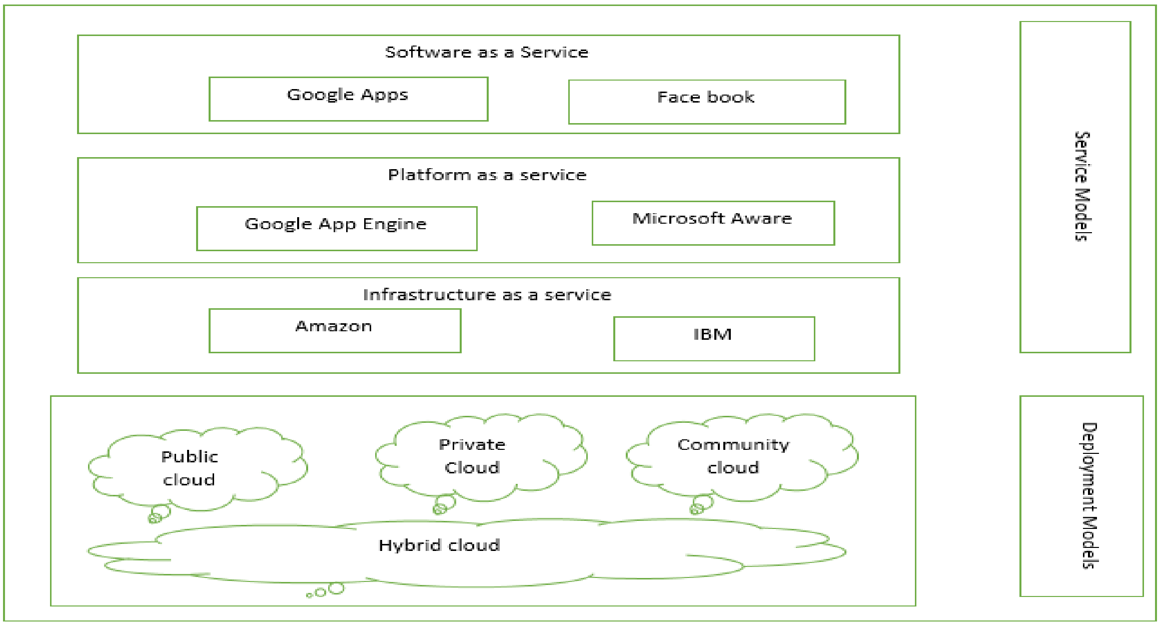

Fig. 1. Service and Deployment Models.

The major issue in Cloud Computing environment is Task Scheduling. Fig 2 shows the working of Task Scheduling. Therefore, there is a need for an effective task scheduling algorithm, which is considered key for the performance of the system [2] Scheduling is the process of allocation of available resources on the basis of need and task qualities. [3] Scheduling can be done at two level namely task level and resource level. Many researchers proposed various algorithms related to resource allocation, these algorithms scale in terms of waiting time, turnaround time, throughput, response time and context switching but still these algorithms require more improvements to fulfill requirement in the cloud environment.

Three common classification of Task Scheduling Algorithms in Cloud Computing environment namely traditional algorithm such as First come First serve, shortest job first, largest job first, Shortest Remaining Time First(SRTF), Earlier Deadline First(EDF), Round Robin(RR) [4], Heuristic Algorithm such as Min-Min and Max-Min Algorithm [5], Meta Heuristic Algorithm such as Ant Colony Optimization algorithm [6]and Particle Swarm Optimization [7].

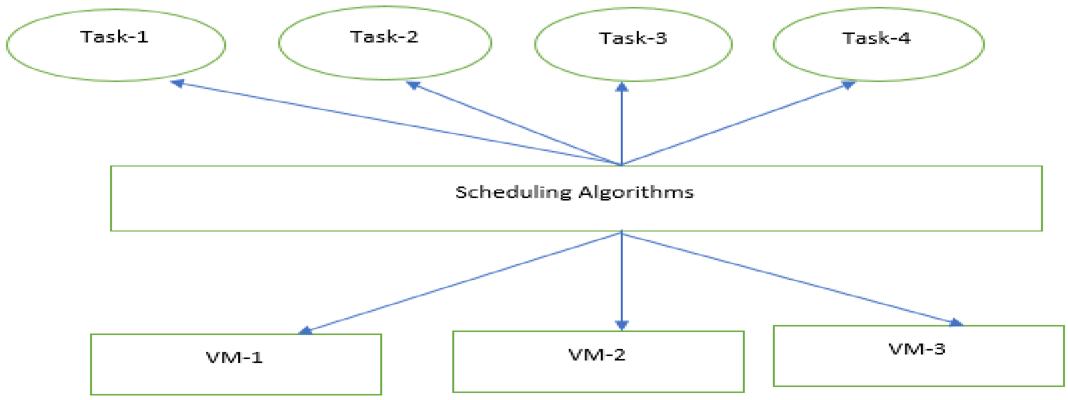

Fig. 2. Task Scheduling.

There are various parameters of Scheduling Algorithms as Context Switching, Throughput, CPU Utilization, WT, Tat and RT as shown in Table 1. In Time Sharing system RR algorithm is commonly used. [8]. It is divided into two parts Static RR and Dynamic RR. In Static RR TQ is fixed till end of the execution but in Dynamic RR TQ vary iteration by iteration and also improve the performance of the system. The efficiency of RR algorithm depends on TQ, to improve the performance of algorithm, the choice of TQ is a critical issue. If the size of quantum is too large RR becomes a First come First Serve algorithm, moreover if size of quantum is too small it poorly performs due to context switching. The Time Quantum (TQ) is an important characteristics of RR algorithm. Some authors proposed static time quantum in round robin algorithm but it not always give better solution. Another alternate solution is dynamic time quantum in which the time slice changes on ready queue for execution. In RR scheduling algorithm the time slice lies between 10 to 100 ms, also called TQ. [9]. The time quantum decides the performance of scheduling algorithms.

Table 1. Criteria of Round Robin Algorithm.

S. No	Parameter	Description
1.	Context Switch (minimize)	It is the process of switching the status from one process to another
2.	Turnaround time (minimize)	It is the time interval or time duration from process submission to process completion
3.	Waiting Time (minimize)	The time waiting in the ready queue
4.	Response Time (minimize)	It is the time at which cloudlet in the ready queue is given first chance for execution in the virtual machine
5.	CPU utilization (maximize)	The amount of work done by CPU
6.	Throughput (maximize)	It is the measurement of how many numbers of units a system can be processed at a given interval of time

The major emphasis of this work comprises:

- ERACS architecture is being proposed.
- For evaluation we compared our result with other algorithms and get better results in terms of AWT, ATAT, ART and NCS.

This paper proposed a method to optimize scheduling. The proposed algorithm is implemented and give improved result. The paper is divided in following sections as Literature Survey presents in Section-II, Section-III and IV covers Proposed Methodology and Experimental-Setup, Results discussed in Section V and Section VI concludes the papers.

2. Literature Survey

The selection of quantum time or time quantum is key to success in RR, many studies have been made to improve the efficiency of RR algorithm. The literature survey of existing algorithms is shown in Table 2. The author in [10] remove the limitations of existing algorithms that consider the priorities of the process. Initially arrange the process based on priorities Then in first iteration the time quantum is fixed in this algorithm, in second iteration rearrange the process based on remaining Burst Time (BT). It shows good result than conventional algorithms in context of AWT, ART and NCS. The proposed algorithm by S. Saeidi in [11] suggest novel idea to calculate TQ of RR algorithm that minimize AWT of the process. In [12] Author proposed Improved Shortest Remaining Burst RR (ISRBRR). Time quantum is calculated as the mean of all process. This method performs well than another traditional RR algorithm. In [13] author proposed Half Life Variable Quantum Time Round Robin algorithm, this method reduces the number of context switching. In [14] author suggested an Improved RR CPU Scheduling Algorithm with Varying TQ, the proposed method experimentally perform better as compared to other conventional RR algorithm that minimize WT, TAT, NCS and maximize CPU utilization. Author proposed an algorithm in [15] that shows better results in minimizing AWT, ART and NCS. Author in [16] proposed an algorithm that improve TQ in the RR algorithm. This new approach gives better results in comparison with other conventional RR algorithm in terms of decreasing the AWT, ATAT and NCS Hani and M. Dorgham in [17] proposed a novel idea that improves the RR algorithm by using the geometric median. The suggested method gives better result in context of AWT and ATAT.Author. in [18] suggested the proposed approach gives better result in minimizing AWT, ATAT and NCS. In [19] Author proposed a modification on RR algorithm where TQ is equal to the BT of the first process. The proposed method reduces WT and TAT. Srinivasu in [20] suggesting a novel approach based on genetic algorithm that reduces the WT. Author in [21] proposed a new idea of RR. It takes highest burst time as time quantum. It minimizes AWT, ATAT and NCS A novel idea proposed by author in [22]. The suggested method decreases NCS. Author in [23] suggested a new idea that remove the limitation of traditional RR algorithm. In proposed approach the TQ is based on mean and the median of BT. This approach minimizing WT and TAT. Author in [24] introduced an Amended Dynamic RR (ADRR) that enhance RR algorithm. The TQ is based on least BT. It gives better result in comparison with another conventional algorithm. In [25] Author proposed a method that combine SJF and RR algorithm. Median is used in this technique as a time quantum. it achieves better result in minimizing the WT and RT. In [26] Author proposed an improved RR algorithm. The result is experimented on CloudSim. It obtained better result in comparison with RR, it minimizes the WT, TAT and NCS. In [27] Author proposed a slight modification on RR algorithm The TQ is calculated on the basis of mean BT of all process. The result is experimented on java and shows better result in context of WT and TAT. In [28] Author proposed an algorithm in which the average BT of all process is treated as TQ which changes dynamically in every iteration. It gives better result in reducing the WT, TAT and NCS. The author in [29] proposed a method that enhance the performance of the RR algorithm The suggested method resulting in better utilization of resources. A review study is suggested in [30]. The proposed algorithm gives better result in optimizing the WT. In [31] author proposed an algorithm which enhance allocation of resources in cloud environment. In [32] author proposed an algorithm in which the TQ is selected on the basis of even and odd number of processes in cloudlet it gives better result in terms of AWT, ART and NCS.

Table 2. Literature Survey of Existing RR Algorithms.

Reference	Year	Algorithms	Parameters	Description
[10]	2012	Priority based RR algorithm	Priority based RR algorithm	Better results as compared to traditional RR algorithms
[11]	2012	Nonlinear mathematical model in RR	Nonlinear mathematical model in RR	Better performance as compared to other algorithms
[12]	2012	ISRBRR	ISRBRR	Better performance as compared to other algorithms
[13]	2013	SRBRR	SRBRR	Minimizing AWT, ATAT and NCS
[14]	2014	HLVQTRR	HLVQTRR	Proposed method reduces context switches
[15]	2014	IRRVQ	IRRVQ	Proposed method performs better than conventional RR
[16]	2015	Improved dynamic RR scheduling algorithm	Improved dynamic RR algorithm	Minimizing AWT, ATAT and NCS
[17]	2016	Improved RR algorithm	Improved RR algorithm	Minimizing AWT and ATAT
[18]	2016	Dynamic Time slice RR algorithm	Dynamic Time slice RR algorithm	Minimizing AWT, ATAT and NCS
[19]	2016	Modified RR algorithm	Modified RR algorithm	Minimizing WT and TAT
[20]	2016	Genetic Algorithm	Genetic algorithm	Minimizing WT
[21]	2017	Efficient Dynamic RR algorithm	Efficient dynamic RR algorithm	Minimizing AWT, ATAT and NCS
[22]	2017	Modified RR with vigorous TQ	Modified RR with vigorous TQ	Better results as compared to traditional algorithms
[23]	2017	Best TQ RR CPU Scheduling algorithm	Best TQ RR CPU scheduling algorithm	Minimizing AWT, ATAT and NCS
[24]	2017	Amended Dynamic RR (ADRR)	Amended dynamic RR(ADRR)	Better performance as compared to other algorithms
[25]	2017	Hybrid dynamic SJF and RR algorithm	Hybrid dynamic SJF and RR algorithm	Minimizing WT and RT
[26]	2017	A new Cloudlet Scheduling Algorithm	A new cloudlet scheduling algorithm	Minimizing WT, TAT and NCS
[27]	2017	Improved RR Scheduling	Improved RR scheduling	Better results in terms of WT and TAT
[28]	2018	Priority Based Efficient RR Algorithm	Priority based efficient RR algorithm	Reduces AWT, ATAT and NCS
[29]	2018	Dynamic Calculated TQ for each VM	Dynamic calculated TQ for each VM	Better utilization of resources
[30]	2018	Modified RR algorithm	Modified RR algorithm	Optimizing WT
[31]	2018	DTQRR	DTQRR	Better results in tems of AWT, ATAT and NCS
[32]	2021	ETQRRA	ETQRRA	Better results in tems of AWT, ATAT, ART and NCS

The calculation of TQ, Gantt Chart, WT, AWT.TAT.ATAT of RR, CRR, ISRBRR, HLVQTRR, IRRVQ, DTQRR, ETQRRA is shown below:

In RR algorithm the TQ is fixed. Here in this algorithm, we assume TQ = 10 for all processes. The Gantt Chart of RR algorithm is shown below:

$$TQ = 10$$

J1	J2	J3	J4	J5	J2	J3	J4	J5	J5

0 10 20 30 40 50 52 57 67 77 80

The WT, TAT and RT is being calculated for each process. The WT is calculated by Eq. (1) and Table 3 shows the calculation of WT and TAT by RR.

$$Waiting\ Time = Completion\ of\ process - CPU\ Burst \quad (1)$$

The WT for Job P1 is (0), WT for Job P2 is (40), WT for Job P3 is (42), WT for Job P4 is (47) and WT for Job P5 is (57). The total WT for all 5 jobs is 186, then the AWT is being calculated by Eq. (2).

$$Average\ Waiting\ Time = \frac{Total\ Waiting\ Time\ of\ Process}{Total\ Number\ of\ processes} \quad (2)$$

$$AWT = \frac{186}{5} = 37.2$$

The TAT is calculated by Eq. (3).

$$Turnaround\ Time = WT + BT \quad (3)$$

The TAT for job P1 is (10), TAT for job P2 is (52), TAT for job P3 is (57), TAT for job P4 is (67) and TAT for job P5 is (80). The total TAT for all 5 jobs is 266, then the AWT is being calculated by Eq. (4).

$$Average\ Turnaround\ Time = \frac{Total\ Turnaround\ Time\ of\ process}{Total\ Number\ of\ processes} \quad (4)$$

The RT is calculated by Eq. (5).

$$Response\ Time = Time\ at\ which\ process\ first\ get\ CPU - Arrival\ Time \quad (5)$$

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (20), RT for job P4 is (30) and RT for job P5 is (40). The total RT for all 5 jobs is 100, then the ART is being calculated by Eq. (6).

$$Average\ Response\ \ Time = \frac{Total\ Response\ Time\ of\ process}{Total\ Number\ of\ processes} \quad (6)$$

$$ART = \frac{100}{5} = 20$$

Table 3. Calculation of WT, TAT and RT by RR.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	40	52	10
Job 3	42	57	20
Job 4	47	67	30
Job 5	57	80	40

In CRR algorithm the TQ is decided on the basis of average of highest burst time and median. The formula for TQ is shown in "Eq. (7)"

$$TQ = \frac{Highest\ BT + Median}{2} \tag{7}$$

$$TQ = \frac{23+15}{2} = 19$$

The Gantt Chart od CRR is shown below:

J1	J2	J3	J4	J5	J4	J5

0 10 22 37 56 75 76 80

The WT, TAT and RT is being calculated for each process by Eq. (1), Eq. (3) and Eq. (5). The Table 4 shows the calculation of WT and TAT by CRR. WT for job P1 is (0), WT for job P2 is (10), WT for job P3 is (22), WT for job P4 is (56) and WT for job P5 is (57). The total WT for all 5 jobs is 145, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{145}{5} = 29$$

The TAT for job P1 is (10), TAT for job P2 is (22), TAT for job P3 is (37), TAT for job P4 is (76) and TAT for job P5 is (80). The total TAT for all 5 jobs is 266, then the AWT is being calculated by Eq. (4).

$$ATAT = \frac{225}{5} = 45$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (22), RT for job P4 is (37) and RT for job P5 is (56). The total RT for all 5 jobs is 125, then the ART is being calculated by Eq. (6).

$$ART = \frac{125}{5} = 25$$

Table 4. Calculation of WT, TAT and RT by CRR.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	10	22	10
Job 3	22	37	22
Job 4	56	76	37
Job 5	57	80	56

In ISRBRR algorithm the TQ is calculated by Eq. (8).

TQ = SQRT (Σ xi2 / n) (8)

$$TQ = 17$$

The Gantt Chart of ISRBRR is shown below:

J1	J2	J3	J4	J5	J4	J5

0 10 22 37 54 71 74 80

The WT, TAT and RT is being calculated for each process by Eq. (1), Eq. (3) and Eq. (5). Table 5 shows the calculation of WT and TAT by ISRBRR.The WT for job P1 is (0), WT for job P2 is (10), WT for job P3 is (22), WT for job P4 is (54) and WT for job P5 is (57). The total WT for all 5 jobs is 143, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{143}{5} = 28.6$$

The TAT for job P1 is (10), TAT for job P2 is (22), TAT for job P3 is (37), TAT for job P4 is (77) and TAT for job P5 is (80). The total TAT for all 5 jobs is 266, then the ATAT is being calculated by Eq. (4).

$$ATAT = \frac{223}{5} = 44.6$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (22), RT for job P4 is (37) and RT for job P5 is (54). The total RT for all 5 jobs is 123, then the ART is being calculated by Eq. (6).

$$ART = \frac{125}{5} = 25$$

Table 5. Calculation of WT, TAT and RT by ISRBRR.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	10	22	10
Job 3	22	37	22
Job 4	54	74	37
Job 5	57	80	54

In HLVQTRR algorithm the TQ is calculated in two parts, first and second half by partition the process BT into two equal halves.in this scenario the TQ is different for each process. For P1(5,5), for P2(6,6), for P3(7,8), for P4(10,10), for P5(11,12). The Gantt chart for this algorithm is shown below:

J1	J2	J3	J4	J5	J1	J2	J3	J4	J5

0 5 11 18 28 39 44 50 58 68 80

The WT, TAT and RT is being calculated for each process by Eq. (1), Eq. (3) and Eq. (5) moreover Table 6 shows the calculation of WT and Tat by HLVQTRR.The WT for job P1 is (34), WT for job P2 is (38), WT for job P3 is (43), WT for job P4 is (48) and WT for job P5 is (57). The total WT for all 5 jobs is 220, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{220}{5} = 44$$

The TAT for job P1 is (44), TAT for job P2 is (50), TAT for job P3 is (58), TAT for job P4 is (68) and TAT for job P5 is (80). The total TAT for all 5 jobs is 302, then the AWT is being calculated by Eq. (4).

$$ATAT = \frac{302}{5} = 60.4$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (5), RT for job P3 is (11), RT for job P4 is (18) and RT for job P5 is (28). The total RT for all 5 jobs is 62, then the ART is being calculated by Eq. (6).

$$ART = \frac{62}{5} = 12.4$$

Table 6. Calculation of WT, TAT and RT by HLVQTRR.

JOBS	WT	TAT	RT
Job 1	34	44	0
Job 2	38	50	5
Job 3	43	58	11
Job 4	48	68	18
Job 5	57	80	28

In IRRVQ algorithm the TQ is different for different process. TQ is calculated on the basis of first BT. for P1(10), for P2(2), for P3(3), for P4(5), for P5(3). The Gantt chart for this algorithm is shown below:

J1	J2	J3	J4	J5	J2	J3	J4	J5	J3	J4	J5	J4	J5	J5	
0	10	20	30	40	50	52	54	56	58	61	64	67	72	77	80

The WT, TAT, RT is being calculated for each process by Eq. (1), Eq. (3) and Eq. (5). The Calculation of WT and TAT is shown in Table 7 The WT for job P1 is (0), WT for job P2 is (40), WT for job P3 is (46), WT for job P4 is (52) and WT for job P5 is (57). The total WT for all 5 jobs is 195, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{195}{5} = 39$$

The TAT for job P1 is (10), TAT for job P2 is (52), TAT for job P3 is (61), TAT for job P4 is (72) and TAT for job P5 is (80). The total TAT for all 5 jobs is 275, then the AWT is being calculated by Eq. (4).

$$ATAT = \frac{275}{5} = 55$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (20), RT for job P4 is (30) and RT for job P5 is (40). The total RT for all 5 jobs is 100, then the ART is being calculated by Eq. (6).

$$ART = \frac{100}{5} = 20$$

Table 7. Calculation of WT, TAT and RT by IRRVQ.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	40	52	10
Job 3	46	61	20
Job 4	52	72	30
Job 5	57	80	40

e-ISSN : 0976-5166
p-ISSN : 2231-3850

In DTQRR algorithm the TQ is calculated on the basis of average BT of all processes as shown in Eq. (9).

$$TQ = \frac{Sum\ of\ process}{Total\ number\ of\ processes} \tag{9}$$

TQ = 16

The Gantt Chart for DTQRR is shown below

J1	J2	J3	J4	J5	J4	J5

0 10 22 37 53 69 73 80

The WT, TAT and RT is being calculated for each process by Eq. (1), Eq. (3) and Eq. (5) moreover Table 8 shows the calculation of WT and TAT by DTQRR.The WT for job P1 is (0), WT for job P2 is (10), WT for job P3 is (22), WT for job P4 is (53) and WT for job P5 is (57). The total WT for all 5 jobs is 142, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{142}{5} = 28.4$$

The TAT for job P1 is (10), TAT for job P2 is (22), TAT for job P3 is (37), TAT for job P4 is (73) and TAT for job P5 is (80). The total TAT for all 5 jobs is 222, then the AWT is being calculated by Eq. (4).

$$ATAT = \frac{222}{5} = 44.4$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (22), RT for job P4 is (37) and RT for job P5 is (53). The total RT for all 5 jobs is 122, then the ART is being calculated by Eq. (6).

$$ART = \frac{122}{5} = 24.4$$

Table 8. Calculation of WT, TAT and RT by DTQRR.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	10	22	10
Job 3	22	37	22
Job 4	53	73	37
Job 5	57	80	53

In ETWRRA algorithm the TQ is calculated on the basis of number of processes (even or odd). For even average remaining execution time approximation method is used, for odd number of processes mid approximation method is used.

For TQ for first iteration there are odd number of processes (5) so by mid approximation method TQ=15, TQ for second iteration is calculated by average remaining execution time approximation method because there are even number of processes (2) so TQ is calculated by Eq. (10).

$$TQ = \frac{\Sigma\ Time}{n} \tag{10}$$

$$TQ = 6$$

So TQ = 6 for second iteration and left BT is 2 so at last TQ = 2

TQ= (15,6,2), for first, second and third iteration. The Gantt chart of this algorithm is shown below:

J1	J2	J3	J4	J5	J4	J5	J5

0 10 22 37 52 67 72 78 80

The WT, TAT and RT is being calculated for each process by Eq. (1), Eq. (3) and Eq. (5). The calculation of WT and TAT of ETWRRA is shown in Table 9. The WT for job P1 is (0), WT for job P2 is (10), WT for job P3 is (22), WT for job P4 is (52) and WT for job P5 is (57). The total WT for all 5 jobs is 151, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{151}{5} = 30.2$$

The TAT for job P1 is (10), TAT for job P2 is (22), TAT for job P3 is (37), TAT for job P4 is (72) and TAT for job P5 is (80). The total TAT for all 5 jobs is 222, then the AWT is being calculated by Eq. (4).

$$ATAT = \frac{221}{5} = 44.2$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (22), RT for job P4 is (37) and RT for job P5 is (52). The total RT for all 5 jobs is 121, then the ART is being calculated by Eq. (6).

$$ART = \frac{121}{5} = 24.2$$

Table 9. Calculation of WT, TAT and RT by ETWRRA.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	10	22	10
Job 3	22	37	22
Job 4	52	72	37
Job 5	57	80	52

3. Proposed Methodology

To select the Time quantum in RR algorithm is a great issue. Inappropriate time quantum declines the performance in cloud environment. If the size of quantum is very large RR becomes a First come First serve algorithm, moreover if size of quantum is very small it poorly performs due to context switching. The Time quantum is an important characteristics of round robin algorithm. Context switching implies the method of switching the Virtual Machine (VM) from one cloudlet to another cloudlet.Fig.3. shows the proposed framework of ERACS So the proposed method enhances the performance of the system moreover reduces the WT, TAT, RT and NCS. The TQ is calculated by using TQ = $[M +1 / 2]^{th}$ term. Initially sort the cloudlet in increasing order as per their BT, then according to the TQ apply on the proposed algorithm and prepare Gantt chant.

Proposed Algorithm

Step 1: START
Step 2: Process arrived in the cloud
Step 2: Assign all the task to the Ready Queue
  Initially Ready Queue == Empty
  Assigning new cloudlet to the Ready Queue on the basis of First Come First Serve.
Step 3: Rearrange Ready Queue Cloudlets in ascending order based on their execution time.
Step 4: Find TQ by using formula:
  TQ = $[M +1 / 2]^{th}$ term
Step 5: Assigning same TQ to all the processes reside in Ready Queue

Step 6: Compute all the processes
Step 7: if TQ = 0
Then
Terminate the process
Step 8: if TQ! =0
Then
Continue till all processes are completed
Step 9: Repeat above steps until there is no cloudlet remaining
Step 10: Calculate WT, TAT, RT and NCS
Step 11: STOP

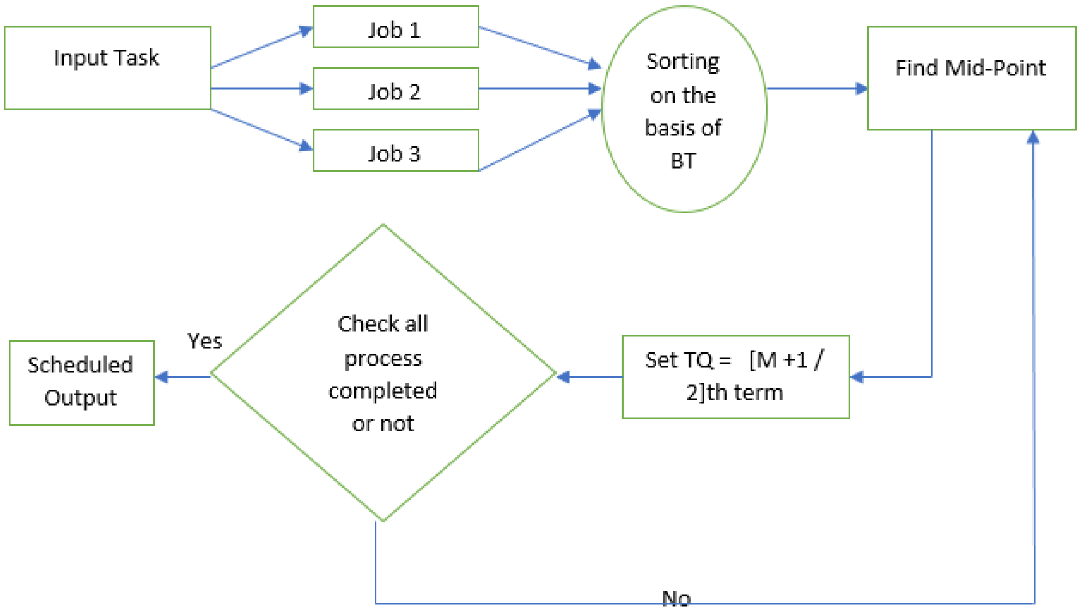

Fig.3. Proposed framework of ERACS.

In our proposed algorithm the TQ is calculated by the formula as shown below in “Eq. (11)”.

$$TQ = [M + 1 / 2]^{th} \text{ term} \quad (11)$$

Where,

M= number of jobs

$= [5+1 / 2]^{th}$ term

$= [6 / 2]^{th}$ term

$= 3^{th}$ term

=15

So, we take
$TQ = 15$, for our proposed method

The Gantt Chart of our proposed algorithm is:

J1	J2	J3	J4	J5	J4	J5

0 10 22 37 52 67 72 80

The WT, TAT and RT is being calculated for each process of our proposed method by Eq. (1), Eq. (3) and Eq. (5), the calculation of WT, TAT and RT of our proposed method is shown in Table 10. The WT for job P1 is (0), WT for job P2 is (10), WT for job P3 is (22), WT for job P4 is (52) and WT for job P5 is (57). The total WT for all 5 jobs is 142, then the AWT is being calculated by Eq. (2).

$$AWT = \frac{141}{5} = 28.2$$

The TAT for job P1 is (10), TAT for job P2 is (22), TAT for job P3 is (37), TAT for job P4 is (72) and TAT for job P5 is (80). The total TAT for all 5 jobs is 222, then the AWT is being calculated by Eq. (4).

$$ATAT = \frac{221}{5} = 44.2$$

The RT is calculated by Eq. (5).

The RT for job P1 is (0), RT for job P2 is (10), RT for job P3 is (22), RT for job P4 is (37) and RT for job P5 is (52). The total RT for all 5 jobs is 121, then the ART is being calculated by Eq. (6).

$$ART = \frac{121}{5} = 24.2$$

Table 10. Calculation of WT, TAT and RT of our proposed Method.

JOBS	WT	TAT	RT
Job 1	0	10	0
Job 2	10	22	10
Job 3	22	37	22
Job 4	52	72	37
Job 5	57	80	52

- Arrival Time: Arrival time is defined as the time at which process entered into the ready queue for execution

$$Arrival\ Time = Completion\ Time - Turnaround\ Time \quad (13)$$

- Burst Time: It is defined as the required time for executing the process.

$$Burst\ Time = Completion\ Time - Waiting\ Time \quad (14)$$

- Starting Time: The initial time process is allocated to VM for execution.

$$Starting\ Time = 0, When\ VM\ is\ free$$
$$Starting\ Time > 1, When\ VM\ consists\ some\ process$$

- Execution Time: Execution time is defined as the time taken to complete execution by the cloudlet.

$$Execution\ Time = \frac{Size\ of\ Cloudlet}{Speed\ of\ VM} \quad (15)$$

Size of cloudlet is measured in Million Instruction and speed of cloudlet is measured in Million Instructions per second

- Turnaround Time: The time interval or time duration from process submission to process completion.

$$Turnaround\ Time = WT + BT \tag{16}$$

- Waiting Time: The amount of time waiting in the ready queue

$$Waiting\ Time = Completion\ of\ process - CPU\ Burst \tag{17}$$

- Response Time: Response time is the time at which cloudlet in the ready queue is given first chance for execution in the virtual machine.

$$Response\ Time = Time\ at\ which\ process\ first\ get\ CPU - Arrival\ Time \tag{18}$$

- Make span: It is defined as sum of execution of all the cloudlets.

- Throughput: It is the measurement of how many number of units a system can be processed at a given interval of time.

4. Experimental Setup

This research is implemented on workstation with specification as core i7 processror,32 GB RAM,1TB HDD,8GB NVidia GPU,2080 RTX and windows 10 enterprise edition. The proposed method is implemented using cloudsim. Cloudsim is java-based platform consisting set of in-built classes and libraries.it allow the user to create data centers, host, VM in a single standalone machine. It performs basic operations like Cloudlet creation, cloudlet scheduling, host creation, virtual machine creation. Cloudlet (C) is defined as small scale cloud datacenter located at edge of the internet. Some of the VM parameters and C- parameters are shown in Table 11 and Table 12.

Table 11. VM Parameters with specifications.

S. No	Criteria	Specification
1	VM-ID	Distinctive ID for virtual Machine
2	VM-User ID	Distinctive ID for owner of virtual machine
3	Capacity	Capacity in Million Instructions per second
4	Number of pes	Number of processing elements of CPU
5	Random Access Memory	Input file size before process
6	Bandwidth	Output file size before process
7	SIZE	Size of Memory
8	TYPE	Type of virtual machine monitor
9	Scheduler	Specification of scheduling policy

Table 12. C- Parameters with specifications.

S. No	Criteria	Specifications
1	C-ID	Unique ID for Cloudlet
2	C- length	Number of executable instructions in cloudlet
3	Number of Pes	Number of processing elements of CPU
4	C-file size	Input file size before process
5	C-output size	Output file size before process
6	Utilization Model 1	Utilization model for CPU
7	Utilization Model 2	Utilization model for RAM
8	Utilization Model 3	Utilization model for Bandwidth

The simulation is done by taking five cloudlets of different size and execution time. Table 13 shows the list of cloudlets with its Cloudlet ID and Burst Time. In cloudlet Table we are taking five jobs along with BT (10,12,23,20 and 15) and Table 14 shows the cloudlet sorted table in which all five jobs are sorted according to their BT. The processes are sorted in (10,12,15,20, and 23) sequence.

Table 13. Cloudlet Original Table.

S. No	Cloudlet ID	CPU Burst
I	Job 1	10
II	Job 2	12
III	Job 3	23
IV	Job 4	20
V	Job 5	15

Table 14. Cloudlet Sorted Table.

S. No	Cloudlet ID	CPU Burst
I	Job 1	10
II	Job 2	12
III	Job 5	15
IV	Job 4	20
V	Job 3	23

5. Results

ERACS method gives better result and the comparative study is shown in Table 16. It shows AWT, ATAT, ART and NCS incurred by existing algorithms.

On the basis of performance our proposed method results better in comparison with other conventional algorithms. In context of AWT our proposed method has less AWT (28.2) in comparison with Round Robin (37.2), Classic RR Algorithm (29), Improved Shortest Remaining Burst RR Algorithm (28.6), Half Life Variable Quantum Time RR Algorithm (44), Improved RR Varying Quantum Algorithm (39), Dynamic Time Quantum RR (28.4) and Enhanced Time Quantum based Round Robin Algorithm (30.2) we are getting. The graphical representation of ERACS based on WT is shown in Fig 4. In terms of ATAT our proposed method has less ATAT (44.2) in comparison with Round Robin (53.2), Classic RR algorithm (45), Improved Shortest Remaining Burst RR Algorithm (44.6), Half Life Variable Quantum Time Round Robin Algorithm (60.4), Improved Round Robin Varying Quantum Algorithm (55), Dynamic Time Quantum Round Robin (44.4) and Enhanced Time Quantum based Round Robin Algorithm (44.2). The graphical representation of ERACS based on TAT is shown in Fig 5. In context of ART our proposed method has less ART (24.2) in comparison with Round Robin (20), Classic RR Algorithm (25), Improved Shortest Remaining Burst RR Algorithm (24.6), Half Life Variable Quantum Time RR Algorithm (12.4), Improved RR Varying Quantum Algorithm (39), Dynamic Time Quantum RR (20) and Enhanced Time Quantum based Round Robin Algorithm (24.2) we are getting. The graphical representation of ERACS based on RT is shown in Fig 6. we are calculated. In terms of NCS our proposed method has less NCS (7) in comparison with Round Robin (10), Classic RR algorithm (7), Improved Shortest Remaining Burst RR Algorithm (7), Half Life Variable Quantum Time Round Robin Algorithm (10), Improved Round Robin Varying Quantum Algorithm (15), Dynamic Time Quantum RR (7) and Enhanced Time Quantum based Round Robin algorithm (8) is being calculated. The graphical representation of ERACS based on NCS is shown in Fig. 7.

The TQ is calculated for all existing and proposed algorithms as shown in Table 15.

Table 15. The Time Quantum of various algorithms.

S. No	Algorithm	Time Quantum
1	RR	10
2	CRR	19
3	ISRBRR	17
4	HLVQTRR	(5,6,7,10,11) AND (5,6,8,10,12)
5	IRRVQ	10,2,3,5,3
6	DTQRR	16
7	ETQRRA	15,6,2
8	PROPOSED	15

Table 16. The Quantitative Comparison of various algorithms based on different parameters.

S. No	Algorithm	AWT	ATAT	ART	NCS
1	RR	37.2	53.2	20	10
2	CRR	29	45	25	7
3	ISRBRR	28.6	44.6	24.6	7
4	HLVQTRR	44	60.4	12.4	10
5	IRRVQ	39	55	20	15
6	DTQRR	28.4	44.4	24.4	7
7	ETQRRA	30.2	44.2	24.2	8
8	PROPOSED	28.2	44.2	24.2	7

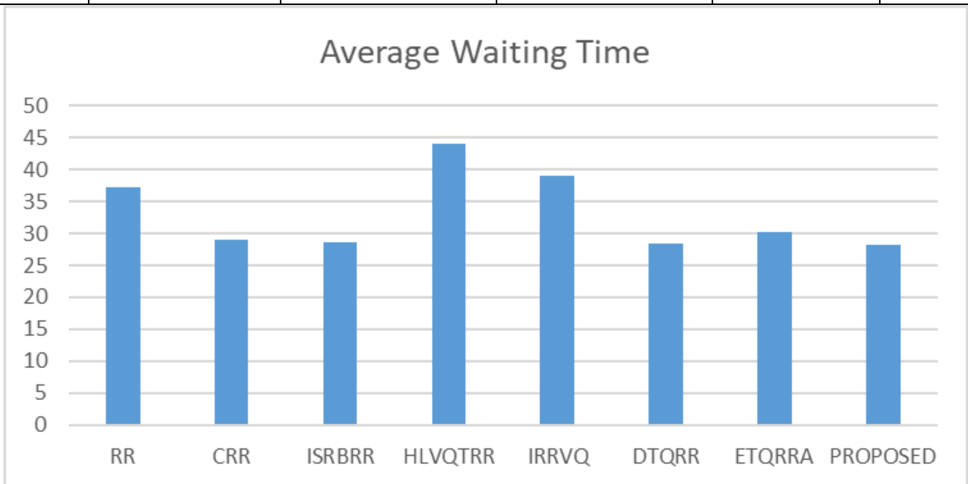

Fig. 4. The graphical representation of ERACS based on WT.

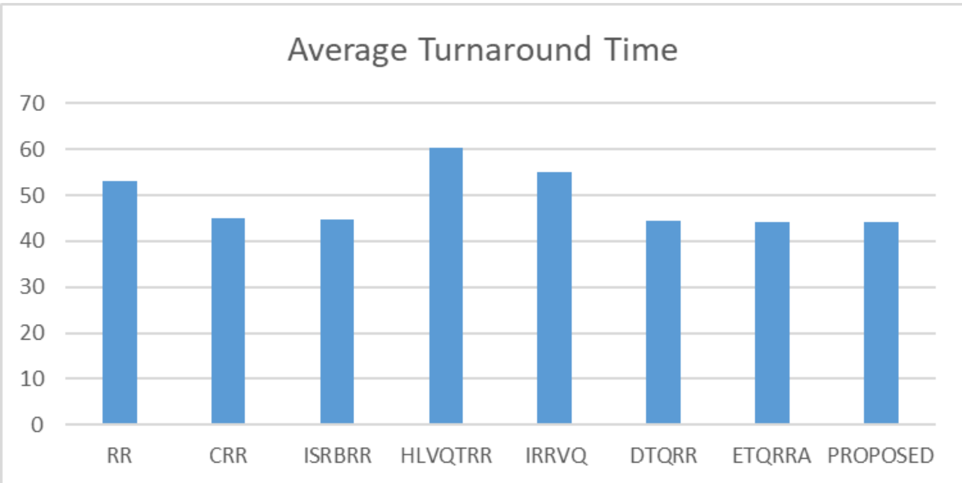

Fig. 5. The graphical representation of ERACS based on TAT.

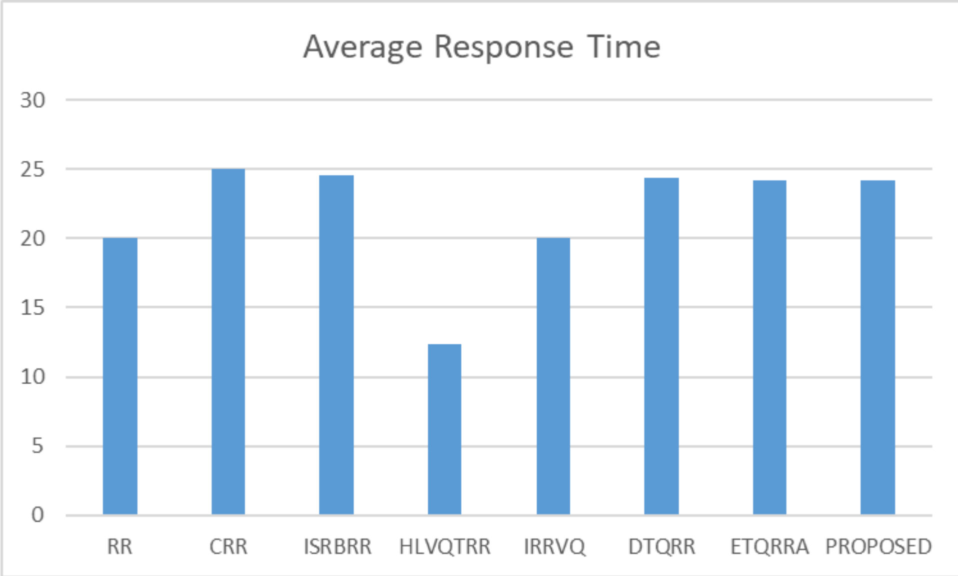

Fig. 6. The graphical representation of ERACS based on ART.

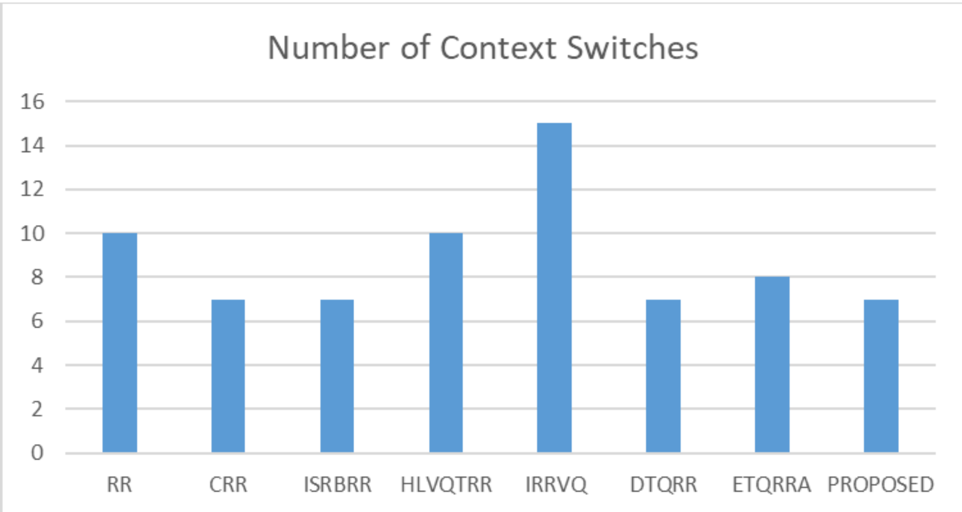

Fig. 7. The graphical representation of ERACS based on NCS.

6. Conclusion

The Time Quantum plays an important role in RR Algorithm. To choose the TQ is complicated task. If the size of quantum is very large Round Robin becomes an FCFS algorithm, moreover if size of quantum is very small it badly performs due to context switching that cause more overhead This study gives a brief concept regarding cloud computing, virtualization and scheduling. The comparative study of proposed method with other existing algorithms namely RR, CRR, ISRBRR, HLVQTRR, IRRVQ, DTQRR, ETQRRA is being made. WT, TAT and NCS is being calculated. This method gives better result as compared to other traditional algorithms. The comparative study gives better performance of our prosed algorithm in terms of AWT (28.2), ATAT (44.2), NCS (7) that helps to improve system performance.

Further, enhance has been made in RR by finding the novel idea for Time Quantum that improves the efficiency of the algorithm and also using modern techniques as fuzzy logic and neural network to determine the appropriate TQ of task.

Acknowledgments

Thanks to our families and colleagues who support us morally. The authors have no relevant financial and non-financial interests to disclose.

References

[1] Razaque, A.; Vennapusa, N.R.; Soni, N.; Janapati, G.S.; Vangala, K.R. (2016): Task scheduling in Cloud computing. In Proceedings of the IEEE Long Island Systems, Applications and Technology Conference (LISAT), Farmingdale, NY, USA, 29 April 2016;Volume 8.
[2] Lynn, T.; Xiong, H.; Dong, D.;Momani, B.; Gravvanis, G.A.; Filelis-Papadopoulos, C.; Elster, A.;Muhammad ZakiMurtaza Khan,M.;Tzovaras, D.; Giannoutakis, K.; et al. (2016).CLOUDLIGHTNING: A framework for a self-organising and self-managing heterogeneouscloud. In Proceedings of the CLOSER 2016, 6th International Conference on Cloud Computing and Services Science, Rome, Italy, Volume 1, pp. 333–338.
[3] Jangra, A., & Saini, T. (2013). Scheduling Otimization in Cloud Computing. Internation Journal of Advance Research in Computer Science and Software Engineering, Volume 3, Issue 4, 62-65.
[4] Alhaidari, F.; Balharith, T.; Al-Yahyan, E.(2019). Comparative analysis for task scheduling algorithms on cloud computing. In Proceedingsof the International Conference on Computer and Information Sciences, Sakaka, Saudi Arabia, 3–4 April 2019;pp. 1–6
[5] Li, X.; Zheng, M.C.; Ren, X.; Liu, X.; Zhang, P.; Lou, C. (2013).An improved task scheduling algorithm based on potential games in cloudcomputing. In Pervasive Computing and the Networked World; Zu, Q., Vargas-Vera, M., Hu, B., Eds.; ICPCA/SWS; Lecture Notes inComputer Science; Springer: Cham, Switzerland,; Volume 8351
[6] Tawfeek, M.; El-Sisi, A.; Keshk, A.; Torkey, F. Cloud task scheduling based on ant colony optimization. Int. Arab J. Inf. Technol., 12, 129–137
[7] Masdari, M.; Salehi, F.; Jalali, M.; Bidaki, M.(2015). A Survey of PSO-Based Scheduling Algorithms in Cloud Computing. J. Netw. Syst. Manag. 2017, 25, 122–158
[8] P. S. Varma, (2013).Available Online at www.jgrcs.info A FINEST TIME QUANTUM FOR IMPROVING SHORTEST REMAINING BURST ROUND ROBIN (SRBRR) ALGORITHM,‖ vol. 4, no. 3, pp. 10–15.
[9] Hitoshi Matsumoto, Yutaka Ezaki.(2011).“ Dynamic Resource Management in Cloud Environment”, July, FUJITSU science & Tech journal, Volume 47, No: 3, page no: 270-276.
[10] I. S. Rajput(2012).A Priority based Round Robin CPU Scheduling Algorithm for Real Time Systems,‖ vol. 1, no. 3, pp. 1–11.
[11] S. Saeidi,(2012).Determining the Optimum Time Quantum Value in Round Robin Process Scheduling Method,‖ no. September, pp. 67–73.
[12] P.Surendra Varma. (2012). “Improved Shortest Remaining Burst Round Robin (ISRBRR) using RMS as its Time Quantum”, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol.1, Issue8.
[13] P. S. Varma(2013).Available Online at www.jgrcs.info A FINEST TIME QUANTUM FOR IMPROVING SHORTEST REMAINING BURST ROUND ROBIN (SRBRR) ALGORITHM,‖ vol. 4, no. 3, pp. 10–15
[14] Simon Ashiru , Salleh Abdullahi , Sahalu Junaidu. (2014.“Half Life Variable Quantum Time Round Robin (HLVQTRR)”, International Journal of Computer Science and Information Technologies, Vol.5(6).
[15] Manish Kumar Mishra and Dr. Faizur Rashid,. (2014).“An Improved Round Robin CPU Scheduling Algorithm with Varying Time Quantum”, International Journal of Computer Science, Engineering and Applications (IJCSEA), Vol.4, No
[16] A. Alsheikhyl, R. Ammar, and R. Elfouly.(2015).An Improved Dynamic Round Robin Scheduling Algorithm Based on a Variant Quantum Time,‖ pp. 98–104,
[17] O. Hani and M. Dorgham.(2016).Improved Round Robin Algorithm : Proposed Method to Apply SJF using Geometric Mean,‖ vol. 5, no. 11, pp. 112–120.
[18] A. Muraleedharan, N. Antony, and R. Nandakumar,.(2016).Dynamic Time Slice Round Robin Scheduling Algorithm with Unknown Burst Time,‖ vol. 9, no. February.
[19] P. Pradhan,(2016).Modified Round Robin Algorithm for Resource Allocation in Cloud Computing,‖ vol. 85, no. Cms, pp. 878–890.
[20] N. Srinivasu,.(2016).A DYNAMIC APPROACH TO TASK SCHEDULING IN CLOUD COMPUTING USING GENETIC ALGORITHM,‖ vol. 85, no. 2.
[21] M. U. Farooq, A. Shakoor, and A. B. Siddique,.(2017).An Efficient Dynamic Round Robin Algorithm for CPU scheduling,‖ no. March,.
[22] B. Dave, P. S. Yadev, M. Mathuria, and Y. M. Sharma,.(2017).Optimize Task Scheduling Act By Modified Round Robin Scheme with Vigorous Time Quantum,‖ 2017 Int. Conf. Intell. Sustain. Syst., no. Iciss, pp. 905–910.
[23] D. Khokhar and A. Kaushik. (2017) ―BEST TIME QUANTUM ROUND ROBIN CPU,‖ no. 5, pp. 3–7.
[24] U. Shafi, M. A. Shah, and Q. Javaid.(2019).A Novel Amended Dynamic Round Robin Scheduling Algorithm for Timeshared Systems,‖ no.
[25] S. Elmougy, S. Sarhan, and M. Joundy.(2017).A novel hybrid of Shortest job first and round Robin with dynamic variable quantum time task scheduling technique.
[26] S. Banerjee, M. Adhikari, and U. Biswas.(2017).Design and analysis of an efficient QoS improvement policy in cloud computing,‖ Serv. Oriented Comput. Appl., vol. 11, no. 1, pp. 65–73.
[27] P. Sangwan, M. Sharma, and A. Kumar.(2017).Improved Round Robin Scheduling in Cloud Computing,‖ vol. 10, no. 4, pp. 639–644.
[28] S. N. Rao, R. Jayaraman, and S. V. N. Srinivasu.(2018).Efficient Pimrr Algorithm based on Scheduling Measures for Improving Real Efficient PIMRR Algorithm Based on Scheduling Measures for Improving Real Time Systems.
[29] S. Banerjee, A. Chowdhury, S. Mukherjee, and U. Biswas.(2018).An Approach Towards Development of a New Cloudlet Allocation Policy with Dynamic Time Quantum 1,‖ vol. 52, no. 3, pp. 208–219.
[30] R. A. B. U. Khurma, H. A. L. Harahsheh, and A. Sharieh.(2018).TASK SCHEDULING ALGORITHM IN CLOUD COMPUTING BASED ON MODIFIED ROUND ROBIN ALGORITHM,‖ vol. 96, no. 17, pp. 5869–5888.
[31] B. Fataniya and M. Patel.(2018).Dynamic Time Quantum Approach to Improve Round Robin Scheduling Algorithm in Cloud Environment,‖ vol. 4, no. 4, pp. 963–969.
[32] D.Gritto and Dr.P.Muthulakshmi.(2021). Enhanced Time Quantum based Round Robin Algorithm for Cloudlet Scheduling in Cloud Environment, Vol.12 No.10 (2021), 5629-5638.

Authors Profile

Puneet Sharma, P. hD Research Scholar in department of Computer Engineering and Applications at GLA University. I have completed my MCA from Anand Engineering College (UPTU) in 2012 with 75% marks and BCA from Barkatullah University Bhopal in 2006. I have 10 years of Teaching experience in Engineering and Degree Colleges as an Assistant Professor. I am published more than 10 papers in journals and conferences. My research area includes Cloud Computing and Machine Learning. I am doing research in Resource Scheduling and optimization in cloud computing.

Manoj Kumar, Professor in GLA University received the Ph.D. degree in Computer Engineering and Applications from GLA University, Mathura, India in 2016. His total teaching experience is more than 21 years. Currently, he is Professor at GLA University. His research interest areas are computer vision and image processing. He has published more than 35 papers. Also, he has published 2 books and 9 patents and 2 patent has been granted. Currently, he is guiding 8 PhD students. Also, running 3 different government projects.

Ashish Sharma, Professor and Dean Academics in GLA university, received the Ph.D. degree in Computer Engineering and Applications from MNNIT Allahabad, M. Tech from UPTU and BE from Shivaji university, Kolhapur. His research interest area is Software engineering, Distributed computing and cloud computing. He has published several research papers in journal and conferences of repute. Also, he has published 4 books and 8 patents and 3 patent has been granted. Currently, he is guiding 10 PhD students.