IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-5 E-ISSN: 2347-2693

Comparative Performance Analysis of Optimized Performance Round Robin Scheduling Algorithm(OPRR) with AN Based Round Robin Scheduling Algorithm using Dynamic Time Quantum in Real Time System with Arrival Time

Pallab Banerjee[1*] and Anita Kumari[2],Puja Jha[3]

Department of Computer Science and Engineering, Cambridge Institute of Technology, Ranchi

Received: May/02/2015 Revised: May/10//2015 Accepted: May/24/2015 Published: May/30/ 2015

Abstract- Round Robin is the most oldest and widely used scheduling algorithm but it has certain limitation because of static time quantum. Time quantum must be large so that context switching becomes reduces and it also effect the response time .So, in this paper we proposed a new algorithm called Optimized Performance Round Robin(OPRR) in which we focused on dynamic time quantum which give result as a very less context switching as well as average waiting time and average turnaround time. It also reduces the overhead of the CPU by adjusting the time quantum according to the highest burst time of the processes in the ready queue.

Keywords-Operating System, Round Robin, Best Performance Round Robin, Turnaround time, Waiting time, Context Switch.

I. INTRODUCTION:

An operating system is a system software which makes interface between user and computer hardware so that system perform in an efficient manner. Scheduling is the process of arranging, controlling and optimizing work and workload in a production process. Operating system follows a predefined procedure for selecting processes from memory and allocates resources as per their requirement. Modern operating system and time sharing system are more complex , they have evolved from a single task to multitasking environment in which processes run in synchronized manner.CPU scheduling algorithm decides which of the process in the ready queue are to be allocated to the CPU[1].

II. SCHEDULING ALGORITHMS

In the First-Come-First-Serve (FCFS) algorithm, process that arrives first is immediately allocated to the CPU based on FIFO policy. In Shortest Job First (SJF) algorithm, process having shortest CPU burst time will execute first. If two processes having same burst time and arrive simultaneously, then FCFS procedure is applied. Priority scheduling algorithm, provides priority (internally or externally) to each process and selects the highest priority process from the ready queue. In case of Round Robin (RR) algorithm, time interval of one time quantum is given to each process present in the circular queue[1].

III. PERFORMANCE METRICS

The proposed algorithm is designed to meet all scheduling criteria such as maximum CPU utilization, maximum throughput, minimum turnaround time, minimum waiting time and context switches. Here we are considering three performance criteria in each case of our experiment[4].

Turnaround Time (TAT)=Finish Time–Arrival Time. Average Turnaround Time should be less.

Waiting Time (WT)= Start Time- Arrival Time. Average Waiting Time should be less.

Context Switch The number of context Switch should be less.

IV. RELATED WORK

In the last few years different approaches are used to increase the performance of Round Robin scheduling like Adaptive Round Robin Scheduling using Shortest Burst Approach Based on Smart Time Slice, Multi-Dynamic time Quantum Round Robin (MDTQRR)[14], Maximum Performance Round Robin (MPRR)[15], High Performance Round Robin (HPRR)[6], Average Max Round Robin Algorithm (AMRR)[2],Even Odd Round Robin (EORR)[5].Mid Average Round Robin(MARR)[8], Average Mid Max Round Robin Algorithm(AMMRR)[4].

V. PROPOSED APPROACH:

The burst time of the processes is taken as sorted increasing order so that it will give better waiting time and turnaround time. In round robin algorithm the performance is based upon the size of static Time Quantum(TQ). If the Time Quantum is too small the then there will be many context switching between the processes. So our approach solved this problem by taking optimized dynamic time quantum TQ. Where TQ is the average of the last two burst time.

AVG=(summation of last two burst time of processes)/2.

TQ= AVG.

VI. PROPOSED ALGORITHM

In our proposed algorithm, processes are already present in the Ready Queue (RQ). The number of processes is (N) and CPU Burst Time (BT) are accepted as input and Average Turnaround Time (ATT), Average Waiting Time (AWT) and number of Context Switch (CS) are produced as output.

```
1. Initialize CS=0, AWT=0, ATT=0.
   //ATT=Average Turnaround time.
   //AWT=Average waiting time.
   //CS=Number of context switch.

2. Sort the process in the ready Queue in
   ascending order of their BT.

3. while(RQ != NULL)
   {
   // N= Number of Processes in the ready queue.
   // BT= Burst Time of the Processes.
   // RQ= Ready Queue.
   //TQ=Time quantum.
   a.) Calculate the average of BT of the last two
       processes and put them in Avg.
         Initialize sum=0;
         For (i=N-1 to N loop)
         sum=sum+ BT(Pi);
        //(Pi)=Process
         Avg=sum/2;
      //Avg=Average of last two process BT.
         TQ = Avg;

   b.) If remaining BT <=TQ then
   Execute that process first with that BT

   c.) If two processes are there then the TQ is
       equal to the average of BT of the two
       processes.

   d.) If one process is there then after calculation
       TQ is equal to BT itself.

   e.) for i=1 to N loop
       // Assign TQ to (1 to N)processes.
          {
            Pi->TQ
   //Assign TQ to all the available processes.
          }
         //End of for.
       }
   //End of while.

4. If (new process arrived and BT!=0 Or new process is
arrived and BT==0 Or new process is not arrived and
BT!=0) then go to step 2. Else go to step 5.

5. Calculate ATT,AWT,CS
6. End
```

VII. EXPERIMENTAL ANALYSIS

In every case we will compare the result of the proposed Optimized Performance Round Robin (OPRR) method with AN Based Round Robin (ANRR) scheduling algorithm and simple Round Robin scheduling Algorithm. Here we have taken 10 as the static time quantum (TQ) for RR algorithm.

CASE 1:-

(With Zero Arrival Time) We consider five processes with Burst time (P1=9, P2=15, P3=19 ,P4=45,P5=120) and Arrival Time =0 and Priority(P1=1,P2=2,P3=3,P4=4,P5=5)as shown in the Table 1. Table 2 shows the output using RR , ANRR and OPRR algorithm. Figure 1,2 and 3 shows Gantt chart of RR ,ANRR and OPRR algorithm respectively.

Table 1:Process with Arrival Time and Burst Time

Process	Arrival time	Burst time	Priority
P1	0	9	1
P2	0	15	2
P3	0	19	3
P4	0	45	4
P5	0	120	5

Table 2: Comparison between RR algorithm ,ANRR algorithm and our new proposed OPRR algorithm(Case 1).

Algorithm	Time quantum	Turnaround time	Waiting time	Context switch
RR	10	92.4	53.8	14
ANRR	42,41,37	82.8	41.2	6
OPRR	83,37	74.4	32.8	5

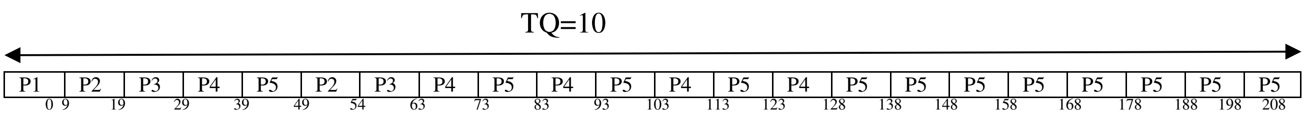

Fig1 : Gantt chart of RR form Table1(CASE 1)

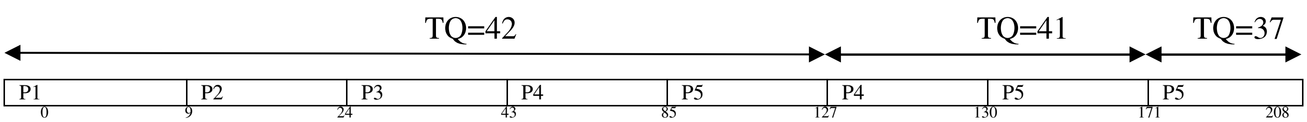

Fig2: Gantt chart of ANRR from Table1

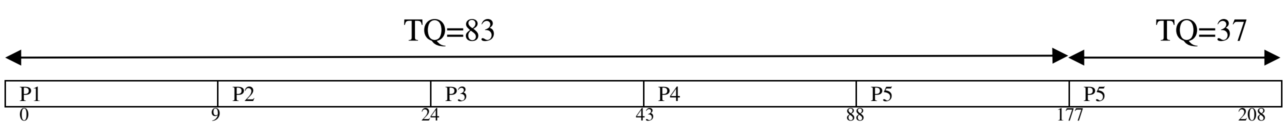

Fig 3: Gantt chart of OPRR from Table1

CASE 2:-

(With Zero Arrival Time) We consider five processes with Burst time (P1=7, P2=14, P3=28 ,P4=56,P5=112) and Arrival Time =0 and Priority(P1=1,P2=2,P3=3,P4=4,P5=5)as shown in the Table 3. Table 4 shows the output using RR , ANRR and OPRR algorithm. Figure 4, 5 and 6 shows Gantt chart of RR ,ANRR and OPRR algorithm respectively.

Table 3:Process with Arrival Time and Burst Time

Process	Arrival time	Burst time	Priority
P1	0	7	1
P2	0	14	2
P3	0	28	3
P4	0	56	4
P5	0	112	5

Table 4: Comparison between RR algorithm ,ANRR algorithm and our new proposed OPRR algorithm(Case 2).

Algorithm	Time quantum	Turnaround time	Waiting time	Context switch
RR	10	103.8	61.2	17
ANRR	43,41,28	88.4	45	6
OPRR	84,28	79.8	36.4	5

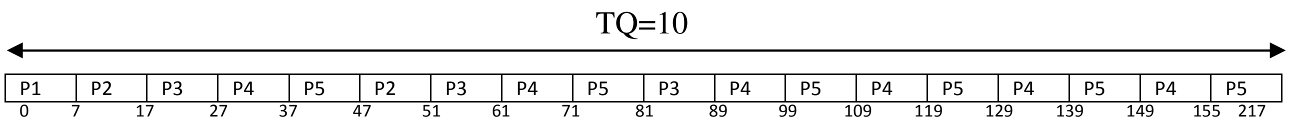

Fig4 : Gantt chart of RR form Table3(CASE 2)

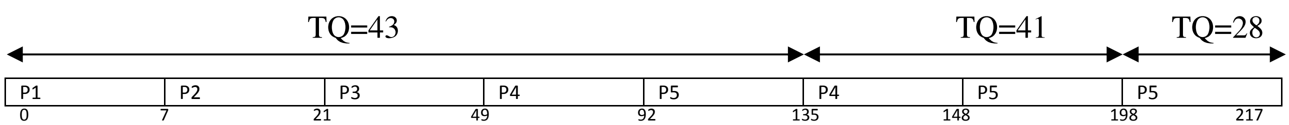

Fig5 : Gantt chart of ANRR form Table3(CASE 2)

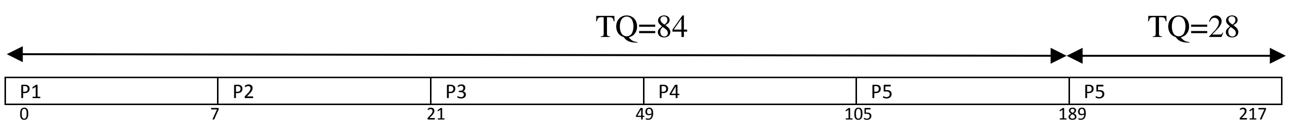

Fig 6: Gantt chart of OPRR from Table3(CASE 2)

CASE 3:-

(With Zero Arrival Time) We consider five processes with Burst time (P1=41, P2=42, P3=43 ,P4=44,P5=45) and Arrival Time =0 and Priority(P1=1,P2=2,P3=3,P4=4,P5=5)as shown in the Table 5. Table 6 shows the output using RR , ANRR and OPRR algorithm. Figure 7,8 and 9 shows Gantt chart of RR ,ANRR and OPRR algorithm respectively.

Table 5:Process with Arrival Time and Burst Time

Process	Arrival time	Burst time	Priority
P1	0	41	1
P2	0	42	2
P3	0	43	3
P4	0	44	4
P5	0	45	5

Table 6: Comparison between RR algorithm ,ANRR algorithm and our new proposed OPRR algorithm(Case 3)

Algorithm	Time quantum	Turnaround time	Waiting time	Context switch
RR	10	207	168	24
ANRR	43,2	136	93	6
OPRR	45	127	84	4

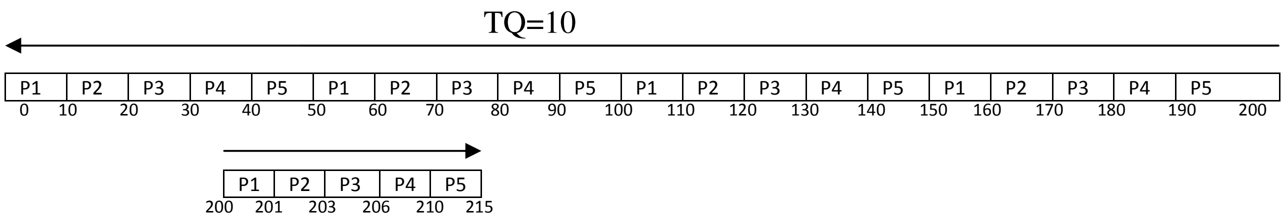

Fig7 : Gantt chart of RR form Table5(CASE 3)

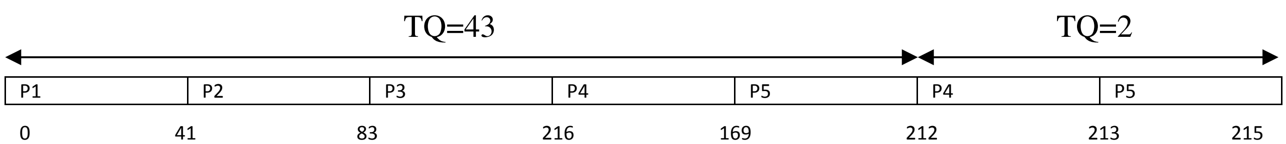

Fig8 : Gantt chart of ANRR form Table5(CASE 3)

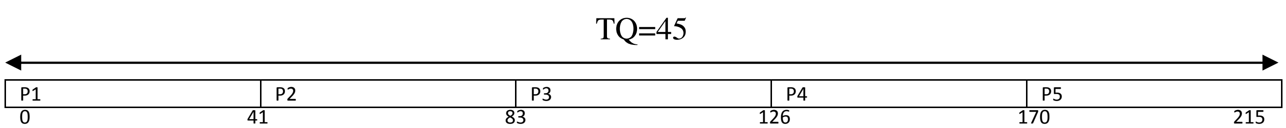

Fig9: Gantt chart of OPRR from Table5(CASE 3)

CASE 4:-

(With Arrival Time) We consider five processes with Burst time (P1=9, P2=17, P3=25 ,P4=38,P5=47) and Arrival Time (P1=0,P2=7,P3=13,P4=20,P5=25) and Priority(P1=1,P2=2,P3=3,P4=4,P5=5)as shown in the Table 7. Table 8 shows the output using RR , ANRR and OPRR algorithm. Figure 10,11 and 12 shows Gantt chart of RR ,ANRR and OPRR algorithm respectively.

Table 7:Process with Arrival Time and Burst Time

Process	Arrival time	Burst time	Priority
P1	0	9	1
P2	7	17	2
P3	13	25	3
P4	20	38	4
P5	25	47	5

Table 8: Comparison between RR algorithm ,ANRR algorithm and our new proposed OPRR algorithm(Case 4)

Algorithm	Time quantum	Turnaround time	Waiting time	Context switch
RR	10	69	55	13
ANRR	27,18,4	54	40	6
OPRR	43,4	49	35	4

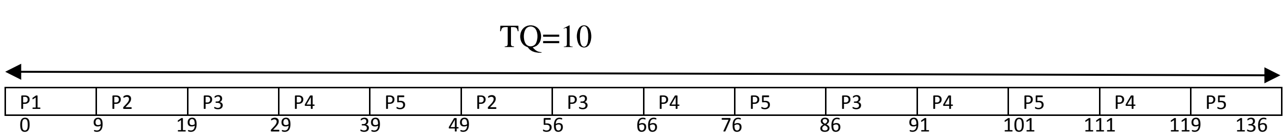

Fig10 : Gantt chart of RR form Table7(CASE 4)

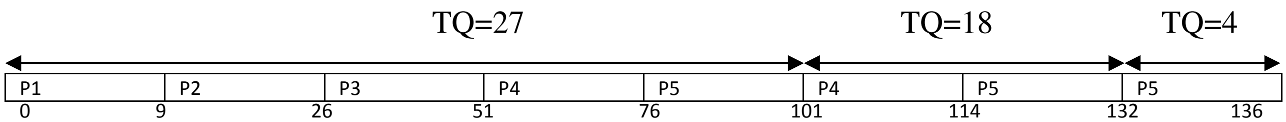

Fig11 : Gantt chart of ANRR form Table7(CASE 4)

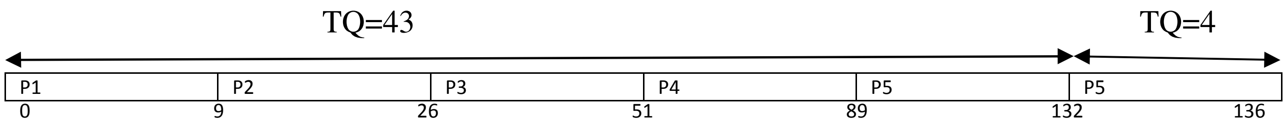

Fig12: Gantt chart of OPRR from Table7(CASE 4)

CASE 5:-

(With Arrival Time) We consider five processes with Burst time (P1=5, P2=13, P3=25 ,P4=37,P5=49) and Arrival Time (P1=0,P2=4,P3=15,P4=17,P5=22) and Priority(P1=1,P2=2,P3=3,P4=4,P5=5)as shown in the Table 9. Table 10 shows the output using RR , ANRR and OPRR algorithm. Figure 13,14 and 15 shows Gantt chart of RR ,ANRR and OPRR algorithm respectively.

Table 9:Process with Arrival Time and Burst Time

Process	Arrival time	Burst time	Priority
P1	0	5	1
P2	4	13	2
P3	15	25	3
P4	17	37	4
P5	22	49	5

Table 10: Comparison between RR algorithm ,ANRR algorithm and our new proposed OPRR algorithm(Case 5)

Algorithm	Time quantum	Turnaround time	Waiting time	Context Switch
RR	10	63	49	13
ANRR	26,17,6	49	34	6
OPRR	43,6	43	29	4

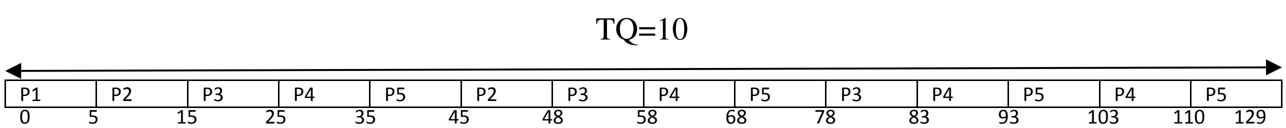

Fig13 : Gantt chart of RR form Table9(CASE 5)

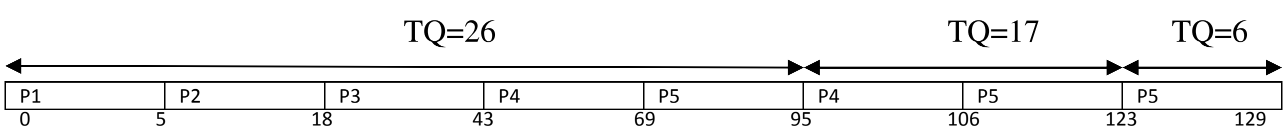

Fig14 : Gantt chart of ANRR form Table9(CASE 5)

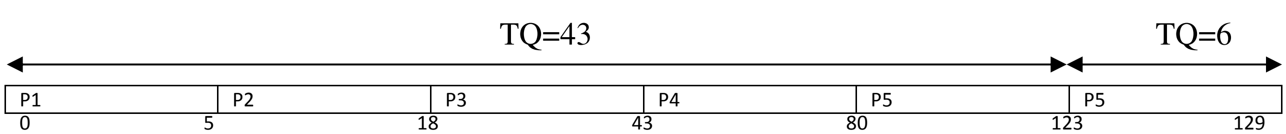

Fig 15: Gantt chart of OPRR from Table9(CASE 5)

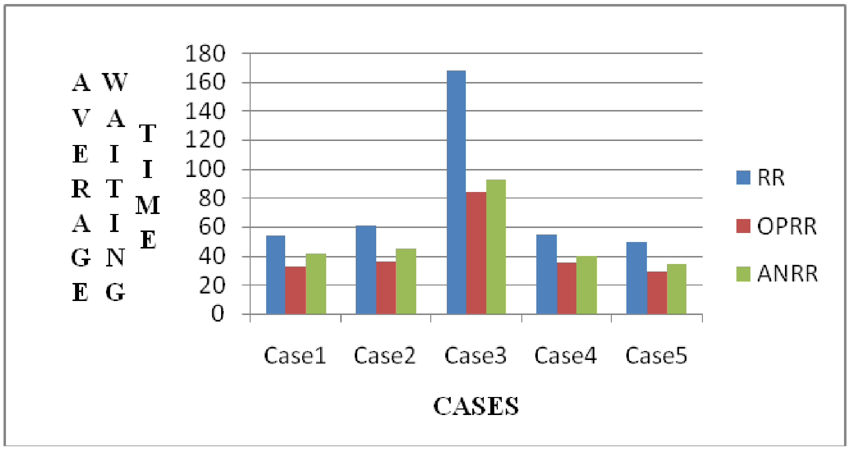

Fig.16: Comparison of average Waiting Time of RR, OPRR and ANRR taking arrival time into consideration.

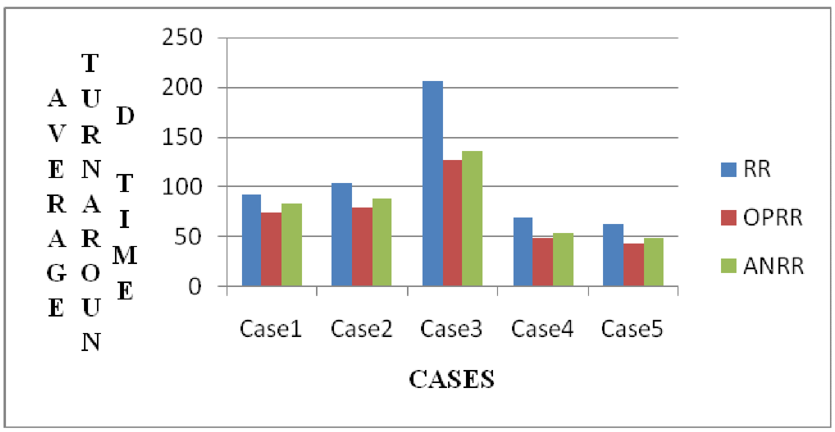

Fig.17: Comparison of average Turnaround Time of RR, OPRR and ANRR taking arrival time into consideration.

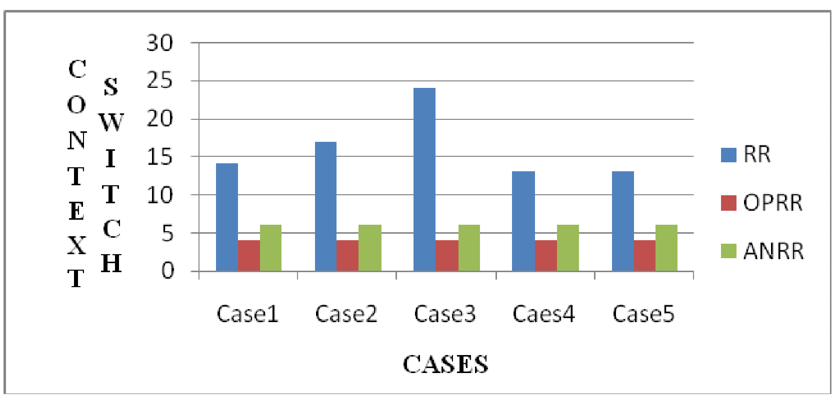

Fig.18: Comparison of Context Switch of RR,OPRR and ANRR taking arrival time into consideration

VIII. CONCLUSION

CPU is one of the most important component of the computer resources.CPU scheduling involves careful examination of waiting processes to determine the most efficient way to service the request. In this paper an optimized performance round robin scheduling algorithm is proposed. This proposed OPRR CPU scheduling algorithm always gives better performance than RR. This is achieved by increasing Time Quantum Dynamically with decreasing the total Turnaround Time, Average Waiting Time and Number of Context Switching.

REFERENCE

[1] "Silberschatz, A., P.B. Galvin and G. Gagne, 2008"Operating Systems Concepts. 7th Edn., John Wiley and Sons, USA, ISBN: 13: 978-0471694663 , pp: 944.

[2] Pallab banerjee, probal banerjee, shweta sonali dhal, " Comparative Performance Analysis of Average Max Round Robin Scheduling Algorithm (AMRR) using Dynamic Time Quantum with Round Robin Scheduling Algorithm using static Time Quantum ",IJITEE, ISSN: 2278-3075, Volume-1, Issue-3, August 2012.

[3] "Tanebaun, A.S., 2008" Modern Operating Systems. 3rd Edn., Prentice Hall, ISBN: 13:9780136006633, pp: 1104.

[4] Pallab banerjee, probal banerjee, shweta sonali dhal, "Performance Evaluation of a New Proposed Average Mid Max Round Robin (AMMRR) Scheduling Algorithm with Round Robin Scheduling Algorithm",IJARCSSE,ISSN:2277-128X, Volume-2, Issue-8, August 2012.

[5] Pallab banerjee, probal banerjee, shweta sonali dhal,"Comparative Performance Analysis of Even Odd Round Robin Scheduling Algorithm (EORR) using Dynamic Time Quantum with Round Robin Scheduling Algorithm using static Time Quantum" IJARCSSE,ISSN: 2277-128X,Volume-2, Issue-9, August 2012.

[6] Pallab banerjee, probal banerjee, shweta sonali dhal,"Improved High Performance Round Robin Scheduling Algorithm(HPRR)using Dynamic Time Quantum" International Journal of Computer Information System,ISSN: 2277-128X,Volume-5, No-3, 2012.

[7] Sarojhiranwal and D.r. K.C.Roy"Adaptive Round Robin Scheduling using Shortest Burst Approach Based on Smart Time Slice".volume 2,No. 2,July-Dec 2011,pp. 319-32.

[8] Pallab banerjee, probal banerjee, shweta sonali dhal, "Comparative Performance Analysis of Mid Average Round Robin Scheduling Algorithm (MARR) using Dynamic TimeQuantum with Round Robin Scheduling Algorithm having static Time Quantum",IJECSE,ISSN: 2277- 1956, Volume-1,Issue-4, August 2012.

[9] H. S. Behera, R. Mohanty, and D. Nayak, "A New Proposed Dynamic Quantum with Re-Adjusted Round Robin Scheduling Algorithm and Its Performance Analysis," vol. 5,no. 5, pp. 10-15, August 2010.

[10] Sanjay Kumar Panda and Saurav Kumar Bhoi, "An Effective Round Robin Algorithm using Min-Max Dispersion Measure" ISSN : 0975-3397 ,Vol. 4 No. 01, January 2012.

[11] Tarek Helmy, Abdelkader Dekdouk " Burst Round Robin: As a Proportional-Share Scheduling Algorithm", IEEE Proceedings of the fourth IEEE-GCC Conference on towards Techno-Industrial Innovations, pp. 424-428, 11-14 November,2007.

[12] Yaashuwanth .C & R. Ramesh "Inteligent time slice for round robin in real time operating system", IJRRAS 2 (2), February 2010.

[13] R. J. Matarneh, " Seif-Adjustment Time Quantum in Round Robin Algorithm Depending on Burst Time of the Now Running Proceses ", American Journal of Applied Sciences 6 (10),pp. 1831-1837, 2009.

[14] H. S. Behera, Rakesh Mohanty, Sabyasachi Sahu and Sourav Kumar Bhoi. "Comparative performance analysis of multi-dynamic time quantum round robin (mdtqrr) algorithm with arrival time", ISSN : 0976-5166, Vol. 2, No. 2,Apr-May2011,pp.262-271.

[15]Pallab banerjee ,Prof Dr L.N.Padhy. "Comparative analysis of Maximum performance round robin(MPRR) by Dynamic Time Quantum with static time quantum",ISSN:2277-128X,pg.372-377Vol-4,Issue-11,Nov-2014

[16] Pallab banerjee,Talat Zabin,Shweta Kumari,Pushpa Kumari "Comparative performance analysis of best performance of round robin scheduling algorithm(BPRR) using Dynamic Time quantum with priority based round robin(PBRR) CPU Scheduling algorithm in Real Time System".ISSN:2277-1956,Vol-4,Number-2, pg. 151-159,May 2015.

[17] R.Nallakumar ,Dr.N.Sengottaiyan ,S.Nithya "A Servey of Task Scheduling Methods in Cloud Computing"IJCSE,ISSN:2347-2693,Volume-2,Issue-10 ,2014

[18] Abbas Noon,Ali Kalakech,Seifedine Kadry"A New Round Robin Scheduling Algorithm for Operating Systms: Dynamic Quantum Using Mean Average"IJCSI,ISSN:1694-0814,Vol-8,Issue-3,No-1,May-2011

