# PERFORMANCE IMPROVISATION IN SCHEDULING THROUGH OPTIMAL RESORCE UTILIZATIOON

Ms.Vaibhavi Pandya[1], Mr.Indr Jeet Rajput[2]
[1]M.E (Pursuing) Computer Engineering, [2]HOD of Computer Department
Hashmukh Gosawmi Engineering College, Ahmedabad ,Gujarat

***Abstract: For optimum use of the resources the CPU scheduling is one of the important key concepts of any operating system. There are many scheduling algorithms available for a multi-programmed operating system like FCFS, SJF, Priority, Round Robin etc.[11,15]Round Robin scheduling is one of the efficient technique. Round robin has a limitation that time quantum in it is static and must be taken care to choose the time quantum time fairly. Scheduling is heart of any computer system since it contains decision of giving resources between possible processes[11]. To increase the efficiency dynamic quantum time can be calculated and average waiting time, average turnaround time and context switching can be decreased. Both Round Robin and Dynamic Round Robin are starvation free. The most important issues with any OS is how the scheduler allocates CPU for the processes waiting in the ready queue for processing[23]***

***Keywords: round robin scheduling, ready queue, average waiting time, average turnaround time, context switching, time quantum, dynamic time quantum.***

## I. INTRODUCTION

Central Processing Unit (CPU) is the core part for every computer system and it should be utilized efficiently.So CPU scheduling is an important concept of any Operating System(OS). Sharing of computer resources between multiple processes is known as scheduling[1]. In a uni-processor environment,all process one by one are allow to execute at a time and every process must have to wait in the queue to get chance for using the CPU. The responsibility of scheduler is to select a process for execution from the waiting queue, Reduce average waiting time, average turnaround time, context switches and response time is the core objective of scheduling algorithm.Real time system can be classified as 1. Hard Real Time System 2. Soft Real time System as discussed in [1].Rround robin scheduling comes under the soft real time system[1,12 In Round Robin CPU scheduling, performance of the system depends on the choice of the optimal time quantum[8]. In this algorithm, a small amount of time called as time slice or time quantum is assigned to each process[1]. Based on scheduling activity, scheduling is broken three separate functions as shown in fig. 1. Long term scheduler, Medium term Scheduler, Short term scheduler.[1,6]. The main component of kernel that creates the "selection" is called the scheduler[9]. Long term scheduler is also called as job scheduler. It determines which programs are going into the systems for processing. Medium term scheduler . The responsibility of Medium term scheduler is to temporarily removes a low priority processes or remove a process from the main memory which have longer time inactivity. Short term scheduler is also known as CPU schedule. The responsibility of this scheduler is to invoked when an event occurs that may lead to the blocking of the current process and may opportunity to preempt the currently executing process in favour of another. The scheduling algorithms may either be preemptive or non-preemptive[10]. Preemptive algorithms are those which have nature to switch to the other process in between the execution. The Shortest Remaining Time First (SRTF), Round Robin (RR) scheduling policies are of this type. Non-preemptive algorithms don't switch to the other process and continue to execute till the process has finished. The FCFS and SJF policies are of this type[10].

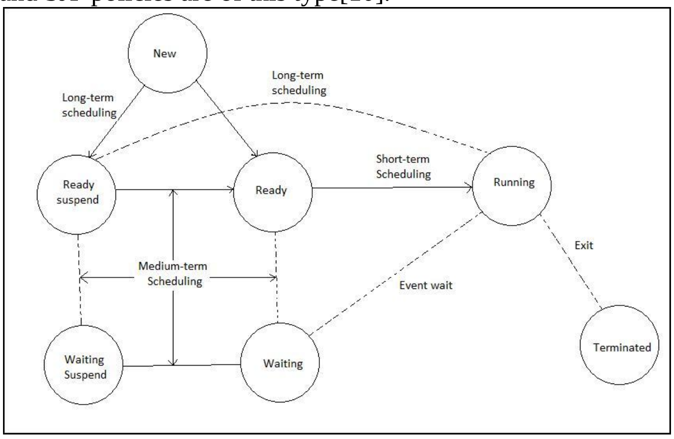


Fig. 1 Type of Schedulers[1]

SCHEDULING CRITERIA

Different CPU scheduling algorithms have different properties, and the choice of a particular algorithm may favor one class of processes over another [6]. The criteria include the following[13][18]:

Utilization/Efficiency: CPU should be best utilized by allocating the significant tasks; so that it should not be ideal.

Throughput: Throughput is defined as number of processes completed in a period of time. Throughput is less in round robin scheduling. Throughput and context switching are inversely proportional to each other[14].

Turnaround time: Total time taken from submission of the process till the completion. Turnaround time should minimize the time of users who wait for the output.

Waiting time: the time between the task becomes available until the first time of its execution[3].

Response Time: Is the duration after submission till the response. It should be minimal in case of interactive users.

Fairness: CPU should be unbiased and every process should get its fair time to execute.

  

## II. RELATED WORK

The RR architecture is a pre-emptive version of the first come first-served scheduling algorithm [7].In a traditional round robin scheduling algorithm a time quantum is assigned to each process in equal portion. So that all processes are executed fairly. For soft real time application CPU scheduling is very efficient as it is starvation free and easy to implement. Many researchers had tried to overcome these problems in round robin in real by giving their own methodologies. The recent studies made from references have shown that if dynamic time quantum is adapted, waiting time, turnaround time, context switches and throughput will be reduced to some larger extent instead of having fixed time quantum. Using dynamic time quantum as stated by Neeraj kumar Rajput and Ashwani kumar [1] proposed algorithm that is adaptive one that selects the time quantum value based on the task set. It will definitely improve the average waiting time, average turnaround time and context switches. In this proposed scheduling algorithm, time quantum is not predefined actually it is adapted on the basis of newly created task sets. Thus this approach does not use traditional methodology and hence gives the better throughput than static round robin.Malhar Thombare, Rajiv Sukhwani, Priyam Shah,Sheetal Chaudhari,and Pooja Raundale.Roy [2]have used shortest burst in [2] for calculation of time slice. Ahmed Alsheikhyl, Reda Ammar1 and Raafat Elfouly [3] has used a comparative analysis between several existing Round Robin algorithms based on the average time for waiting and turn-around and number of context switches.they propose an algorithm that they uses average of two highest burst times was computed and then the average of two lowest arrival times was taken from that estimated value for one time only; later, we subtract the average of arrival time for only lowest process.Rohaya Latip and Zulkhairi Idris[4] The scheduling algorithms of HRRN and FCFS implemented in this paper reorder jobs in the global queue based on the priority scheme variable. Based on the findings,it can be concluded that HRRN outperform FCFS scheduling algorithm by 5% on average in regards to total waiting time versus number of jobs submitted due to its fair aging priority scheme policy.Bin Nie, Jianqiang Du*, Guoliang Xu, Hongning Liu, Riyue Yu, and Quan Wen[5] proposed an include FCFS, SJ (P) FHRRN,and put forward a new operating system scheduling algorithm, named mediantime slice-Highest Response Ratio Next (MTSHRRN) ,the method was proved to be feasible and effective after tested the process sequence.Neetu Goel,Dr. R.B. Garg.[6]have studied many scheduling algorithm and states that to evaluate a scheduling algorithm to code it and has to put it in the operating system, only then a proper working capability of the algorithm can be measured in real time systems. The treatment of shortest process in SJF scheduling tends to result in increased waiting time for long processes.[6]. Mahesh Kumar M.R, Renuka Rajendra.B, Niranjan C.K Sreenatha .M[8] proposed a new algorithm named as “,” Prediction of Length of the Next CPU Burst in SJF Scheduling Algorithm using Dual Simplex Method “in which dual simplex optimization technique to overcome the problem of SJF scheduling by predicting / approximating the length of the next CPU burst. M.V. Panduranga Rao , K.C. Shet[9] states that using MFQ simulator developed by us during the process. To run the simulator, CPU burst of a set of processes are to be entered and while doing so the system ignores the nature (CPU or I/O Bound) of each process. The dynamic time quantum for each queue is generated automatically by the simulator[10]. Through a number of experiments performed, we observed that the performance of MFQ improves by applying SJF selection prior to RR algorithm from second queue onwards using dynamically generated time quantum as compared to other algorithms and for static quantum. Sukumar Babu Bandarupall, Neelima Priyanka Nutulapati,Prof. Dr. P.Suresh Varma[11] make a study about comparison between preeptive and non-preemptive scheduling algorithms.Performance Improvement Using CPU Scheduling Algorithm[24] developed a new algorithm with th help of round robin and SJF named as SRT algorithm and improvesthe functionality of round robin scheduling algorithm. Ayan Bhunia[21] developed an approach for jobs which starve in the lower priority queue for long time to get CPU cycle. As a result response time of those starved processes decreases eight to ten percent and over all turn around time of the whole scheduling process decreases around eight to ten percents.

## III. PROPOSED WORK

The adaptive round robin scheduling is an efficient scheduling algorithm in terms of execution and starvation. In the proposed method, focus is on making task set and then apply the adaptive round robin algorithm. According to the arrival time, task sets are being formed. One time quantum value is valid only for a task set. Every time while preparing task set, the algorithm will select the best suitable time quantum value with the help of greedy approach.

### *A. ALGORITHM*

Assuming algorithm considers that processes are arriving at different instances. The value number of processes will be taken by user itself. The steps of algorithm are showing below—

```
Begin
Step 1:Initialization
        at[n]           //Arrival time of the process
        bt[n]           //burst time of the process
        TQ              //time quantum for the process
        N               //number of process
Step 2: Generate the Queue for the processes
Step 3: Last inserted queue will selected first for the execution.
Step 4:If process is ready for execution
{
goto step5
}
Else
goto step 10.
Step 5:For first process set quantum time is equal to the burst time of the first process
        TQ=bt[Pn]
Step 6:Arrange all the processes in the ready queue.
```

  

Step 7: Execute the process
Step 8: Insert each process into the stack.
Step 9: If
Stack is overflow i.e. SP==10
{
Step 9.1 POP the processes
Step 9.2 Calculate the time quantum TQ from previous 10 process for next 10 process
Step 9.3 Update the value of TQ
Step 9.4: New TQ selected
goto step:6
}
else
wait process for execution
Step 10: Calculate AWT,ATT and CS //Average waiting time, average turnaround time, Context switches

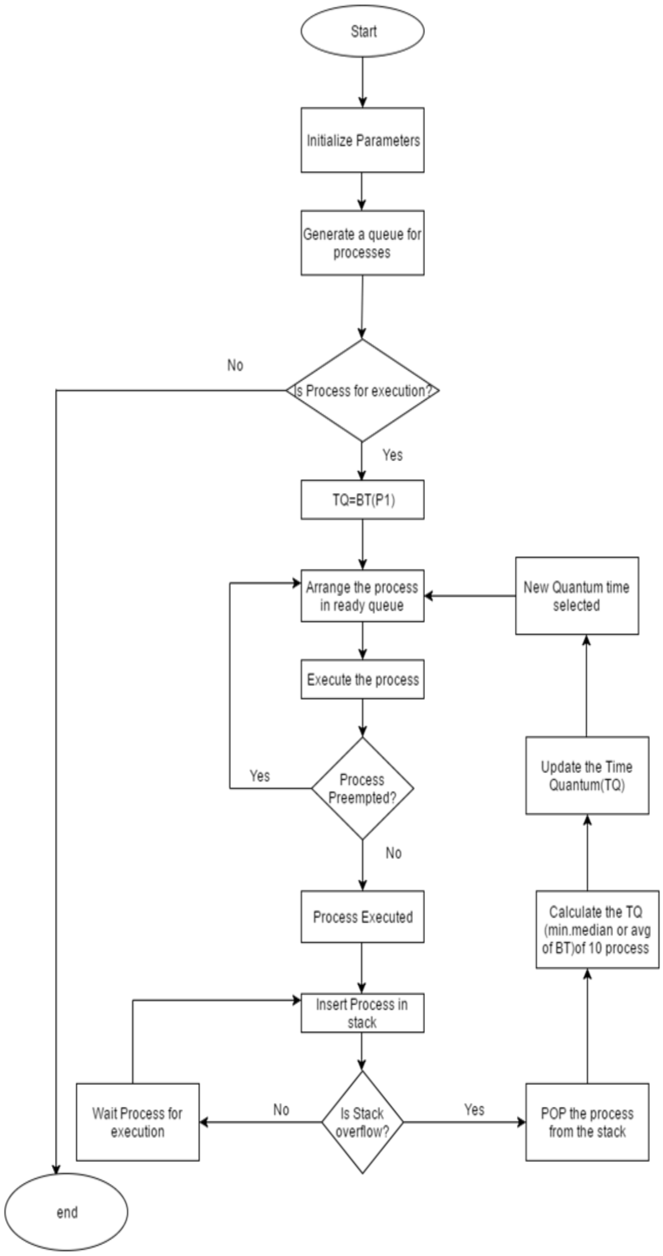


Fig. 2 Flow chart of PRR

*B. FLOW CHART*

Based on the above algorithm, flow chart can be prepared as shown in fig. 3. This shows necessary steps which are taken at different stages. This algorithm starts from the elliptical shape and also ends with shape. All variables are shown by using rectangular shapes and the decision making processes are shown inside the diamond shape having two branches labeling yes and no. Each decision making shape ensures that there is no infinite looping in the proposed algorithm. All rectangular shapes are showing some particular steps same as in stated algorithm. The queue is used for the purpose for the processes which are ready for execution. All processes are reside in the ready queue. The processes which are done with the execution will be kept in the stack. For first ten processes the TQ will be 2ms.this TQ is applicable only for this task set of ten processes. Then on the basis of burst times of every processes' burst time the mean or median burst time for processes will be considered as a new quantum value for next ten processes. This process keeps on continuing until all process finished burst.

## IV. SIMULATION

The proposed TARR algorithm has been simulated along with existing round robin scheduling algorithm for performance comparison. The simulator is implemented by using the C#.net and the performance graphs are drawn in ms- excel. The following matrices namely average waiting time, average turnaround time and number of context switching are measured in the simulated environment. The processes are arriving at different time interval. Tables1 and 2 gives the result for tasks arriving at different instances and different dynamic time quantum adapted according to the algorithm defined in fig.3. The proposed TARR algorithm is simulated and compared with fixed time quantum round robin for different time quantum i.e. fixed round robin time quantum values will vary accordingly but proposed TARR time quantum will adapt it according to algorithm. Traditional round robin scheduling result is calculated for any number of different time quantum values for comparison with proposed TARR algorithm. The result of simulation is shown in fig. 4 for average waiting time, fig. 5 for average turnaround time and fig. 6 for number of context switches. The different simulated can be find out by giving different values of number of processes, range of arrival time, range of burst time and time quantum value only for static round robin.

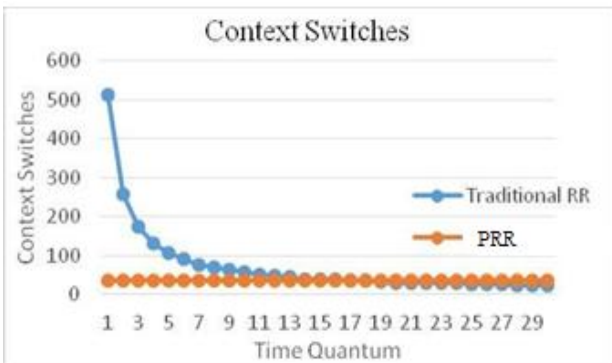


Fig. 3 Context Switches

  


Fig. 4 Average Waiting Time

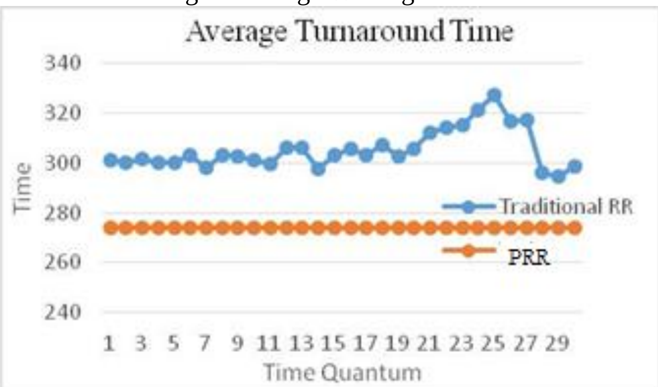


Fig. 5 Average Turnaround Time

## V. RESULT AND CONCLUSION

The proposed algorithm provides the efficient utilization of resources in terms of scheduling since the proposed algorithm is adaptive one that selects the time quantum value 2ms for first ten processes. It will definitely improve the average waiting time, average turnaround time and context switches. The static round robin scheduling is well known algorithm mainly used for fair scheduling and removal of starvation. If the time quantum value is too small, the number of context switches will be high. If time quantum value is too large, it behaves as FCFS that causes starvation. The proposed algorithm ensures that time quantum value should not be too small and too large. And time quantum will change based on the task set. In the static round robin scheduling, time quantum is predefined before the execution take place and applicable for all processes in the ready queue at a particular instance but in this proposed scheduling algorithm, time quantum is not predefined actually it is adapted on the basis of newly created task sets. Thus this approach does not use traditional methodology and hence gives the better throughput than static round robin.

| Process | Arrival Time | Burst Time |
|---|---|---|
| P1 | 2 | 4 |
| P2 | 3 | 2 |
| P3 | 7 | 3 |
| P4 | 11 | 4 |
| P5 | 12 | 1 |
| P6 | 15 | 2 |
| P7 | 17 | 5 |
| P8 | 18 | 4 |
| P9 | 21 | 3 |
| P10 | 22 | 2 |

Let's take processes arrived at different time, having different burst time. AWT stands for average waiting time, ATT stands for average turnaround time and CS stands for context switch. These results are taken from the simulation results by considering any value for time quantum i.e. TQ=2 then apply fixed round robin scheduling and then by applying proposed algorithm time quantum will be adopted as average of two processes execution time for proposed scheduling algorithm as in Tabl, TQ=03ms. In simulation, the result is directly depending on the processes which are generated by some random number generator function.

| Process No | AT | BT | CT | TAT | WT |
|---|---|---|---|---|---|
| 2 | 3 | 2 | 6 | 3 | 1 |
| 1 | 2 | 4 | 8 | 6 | 2 |
| 3 | 7 | 3 | 11 | 4 | 1 |
| 5 | 12 | 1 | 15 | 3 | 2 |
| 4 | 11 | 4 | 16 | 5 | 1 |
| 6 | 15 | 2 | 18 | 3 | 1 |
| 9 | 21 | 3 | 24 | 6 | 3 |
| 7 | 17 | 5 | 29 | 12 | 7 |
| 10 | 22 | 2 | 31 | 9 | 7 |
| 8 | 18 | 4 | 32 | 14 | 10 |

Algorithm can be work in real time environment Most of the RR scheduling algorithm can generate the dynamic time quantum but average of executing time of the processes is not considered. In improved task set based adaptive algorithm for the first ten process the queue behaves in FCFS manner. Separate stack and queue is maintained for the processes. Stack will be limited to the size 10 because of the generation of new quantum in every cycle but for the testing purpose here the size of the stack is 2 only. Two separate calculations of the average turnaround time and AWT is done for fix quantum time and mean quantum time. For the next cycle of the execution the quantum value will be average value of execution time of the processes which were stored in the stack. The mean and fix value can be considered and at the last comparison among all quantum value will be calculated. PUSH and POP operations are performed in stack for insertion and deletion of the processes. Better resource utilization is made because fair quantum time is selected and the process has no need to wait longer. Algorithm uses both the technique using fix quantum time and with the help of mean quantum time. It has been observed that the AWT and average turn around time in mean quantum time compare to fix quantum time algorithm. The algorithm is developed with the help of ASP.NET 3.5 with c# conclave application and the result can be also stored in the ms excel.

## REFERENCE

[1] Neeraj Kumar Rajput, Ashwani Kumar " A Task set Based Adaptive Round Robin (TARR) scheduling algorithm for improving performance", In IEEE 2015 1st International conference on futuristic trend in computational analysis and knowledge

  

management (ABLAZE 2015)

[2] Malhar Thombare, Rajiv Sukhwani, Priyam Shah,Sheetal Chaudhari,and Pooja Raundale." Efficient Implementation of Multilevel Feedback Queue Scheduling",In: *IEEE WiSPNET 2016 conference.*

[3] Ahmed Alsheikhyl, Reda Ammar1 and Raafat Elfouly2," An Improved Dynamic Round Robin Scheduling Algorithm Based on a Variant Quantum Time",In: *IEEE 2015, 978-1-5090-0275-7/15*

[4] Rohaya Latip and Zulkhairi Idris," Highest Response Ratio Next (HRRN) vs First Come First Served (FCFS) Scheduling Algorithm in Grid Environment",In: J.M. Zain et al. (Eds.): ICSECS 2011, Part II, CCIS 180, pp. 688–693, 2011.© Springer-Verlag Berlin Heidelberg 2011

[5] Bin Nie, Jianqiang Du*, Guoliang Xu, Hongning Liu, Riyue Yu, and Quan Wen, " A New Operating System Scheduling Algorithm",In ECWAC 2011, Part I, CCIS 143, pp. 92–96, 2011. © Springer-Verlag Berlin Heidelberg 2011

[6] Neetu Goel,Dr. R.B. Garg." A Comparative Study of CPU Scheduling Algorithms", In: *International Journal of Graphics & Image Processing |Vol 2|issue 4|November 2012*

[7] Maysoon A. Mohammed, Mazlina AbdulMajid and Rana Fareed Ghani, "Queueing Theory Study of Round Robin versus Priority Dynamic Quantum Time Round Robin Scheduling Algorithms", In: *IEEE 2015 4th International Conference on Software Engineering and Computer Systems (ICSECS), Kuantan, Pahang, Malaysia. August 19-21, 2015*

[8] Mahesh Kumar M.R, Renuka Rajendra.B, Niranjan C.K Sreenatha .M," Prediction of Length of the Next CPU Burst in SJF Scheduling Algorithm using Dual Simplex Method",In © IEEE 2014 IEEE Conference Number – 33344 July 8, 2014, Coimbatore, India.

[9] M.V. Panduranga Rao1 , K.C. Shet2," Assessment of Response Time for New Multi Level Feedback Queue Scheduler",In © International Journal of Computer Trends and Technology (IJCTT) – volume 13 number 3 – Jul 2014

[10] Dr. Sanjay. K. Dwivedi, Ritesh Gupta, "A simulator based performance analysis of Multilevel Feedback Queue Scheduling"In IEEE2014 5th International Conference on Computer and Communication Technology

[11] Sukumar Babu Bandarupalli1, Neelima Priyanka Nutulapati2,Prof. Dr. P.Suresh Varma3 "A Novel CPU Scheduling Algorithm–Preemptive & Non-Preemptive" International Journal of Modern Engineering Research (IJMER) ,Vol.2, Issue.6, Nov-Dec. 2012 pp-4484-4490

[12] Christopher McGuire and Jeonghwa Lee"The Adaptive80 Round Robin Scheduling Algorithm" © Springer Science+Business Media Dordrecht 2015 H.K. Kim et al. (eds.), Transactions on Engineering Technologies,DOI 10.1007/978-94-017-7236-5_17

[13] Neetu Goel ,Dr. R. B. Garg, "Performance Analysis of CPU Scheduling Algorithms with Novel OMDRRS Algorithm",In (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 1, 2016

[14] Raman ,Dr.Pardeep, Kumar Mittal "An Efficient Dynamic Round Robin CPU Scheduling Algorithm"© 2014, IJARCSSE All Rights Reserved Volume 4, Issue 5, May 2014

[15] Vaishali Chahar,Supriya raheja "A Review of Multilevel Queue and Multilevel Feedback Queue Scheduling Techniques"in © 2013, IJARCSSE

[16] Supriya raheja,Reena Dadhich,Smita rajpal" Designing of vogue logic based multilevel feedback queue scheduler"© 2015 production and hosting by Elsevier B.V. on behalf computer and information, Cairo university.

[17] M.Ramakrishna, G.Pattabhi Rama Rao" efficient round robin cpu scheduling algorithm for operating systems", m. ramakrishna* et al. / (ijitr) international journal of innovative technology and research Volume No. 1, Issue No. 1, December-January 2013, 103-109

[18] Anju Muraleedharan1*, Neenu Antony2 and R. Nandakumar3" Dynamic Time Slice Round Robin Scheduling Algorithm with Unknown Burst Time" @Indian Journal of Science and Technology, Vol 9(8), DOI: 10.17485/ijst/2016/v9i8/76368,February 2016

[19] Mohd Shoaib, Mohd Zeeshan Farooqui" A Comparative Review of CPU Scheduling Algorithms',@ Proceedings of National Conference on Recent Trends in Parallel Computing (RTPC - 2014),November 1-2, 2014, Mangalayatan University, Aligarh, India.

[20] Lalit Kishor , Rajendra Singh, Praveen Sharma" Optimized Scheduling Algorithm", Proceedings published by International Journal of Computer Applications® (IJCA) International Conference on Computer Communication and Networks CSI-COMNET-2011

[21] Ayan Bhunia" Enhancing the Performance of Feedback Scheduling"@ International Journal of Computer Applications (0975 – 8887) Volume 18– No.4,March 2011

[22] Charu Rani1, Mrs. Manju Bala2" Comparison of Round Robin Based CPU Scheduling Algorithms"@ International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 9, September 2015

[23] Maysoon A. Mohammed*,Mazlina AbdulMajid and Balsam A. Mustafa" A proposed priority dynamic quantum time algorithm to enhance varying time quantum round robin algorithm"@ Int. J. Computer Applications in Technology, Vol. 54, No. 3, 2016

[24] 1Neeraj Kumar, 2Nirvikar" Performance Improvement Using CPU Scheduling Algorithm-SRT" Volume 2, Issue 2, March – April 2013

  