MULTI LEVEL QUEUE ROUND ROBIN CPU SCHEDULING ALGORITHM (MQRR)

[1]G.SWETHA, [2]G.RAMYA

[1,2]ComputerScience Engineering Department, CVRCE, India
E-mail: [1]shreyasnaani@gmail.com, [2]ramya.gangala @gmail.com

Abstract - The main objective of CPU scheduling is maximum utilization of CPU, minimizing the average waiting time(AWT),minimizing the average turnaround time(ATT) and also minimizing the average response time. Various operating systems follow various scheduling algorithms[1][2].The classic scheduling algorithms include First Come First Serve(FCFS),Shortest Job First(SJF),Priority, Round Robin, Multi level queue(MLQ),Multi Level Feedback Queue(MLFQ).The criteria to compare these algorithms include CPU utilization, Throughput, Turnaround time, waiting time and response time. In this paper we have proposed a new CPU scheduling algorithm called multilevel Queue Round Robin(MQRR) CPU scheduling algorithm, which considers the ready queue to be partitioned into two separate queues. We have compared this algorithm with Round-Robin algorithm in terms of Average Waiting Times and Average Turnaround Times and also compared these two algorithms by considering different process arrival times and equal process arrival times. MQRR is giving less AWT and less ATT when compared with Round Robin algorithm in all the cases. Comparison has been done by considering burst times in three different cases, that is ascending order, descending order ,and random order burst times.

Keywords - Average Waiting Time, Average Turnaround Time, Burst time, Gantt chart.

I. INTRODUCTION

One of the important components of computer is Central Processing Unit(CPU).Because it does the processing. Processes need CPU to execute their tasks. In multiprogramming environment where multiple processes are under execution simultaneously, sharing the CPU among many processes is a difficult task. Of course, CPU executes only one task at a time but when a process goes from running state to waiting state, then CPU can be given to any other process which is ready. As there may be many processes in the ready queue, one process need to be selected from many processes which are ready. This is called process scheduling. This is the duty of operating system. Operating system is a program or collection of programs which acts as intermediary between user processes and hardware. Because user can not interact with the hardware directly, operating system does this job. Operating System is a resource manager. It manages many resources of the system like CPU, files, Disk, Printer etc. It schedules processes for utilizing those resources. As CPU is one of the most important parts of the computer, we are concentrating on CPU scheduling in this paper. There are two types of scheduling algorithms. They are preemptive and nonpreemptive. Preemptive scheduling algorithms: When a process is running, CPU can be given to other process by interrupting currently running process. Nonpreemptive scheduling algorithms: When a process is running it should not be interrupted until it completes its CPU burst time.
In classic CPU scheduling algorithms some are preemptive and some are nonpreemptive and also some algorithms are both preemptive and non preemptive.

II. EXISTING ALGORITHMS

There are various algorithms for CPU scheduling [4][6][7][8].

First Come, First Served scheduling: It is a nonpreemptive scheduling algorithm.CPU is assigned to processes in the order of their arrivals. Whichever process comes first into the ready queue, that process will be scheduled first.

Shortest Job First: It is a nonpreemptive scheduling algorithm CPU is assigned to the process that has the smallest burst time. If two processes have same burst times, the FCFS is used to solve the problem.

Shortest Remaining Time First scheduling algorithm: It is a preemptive scheduling algorithm.CPU is assigned to the process which has the smallest burst time. But if a process arrives later with the burst time requirement of less than the remaining time of the currently running process, then current process is interrupted and CPU will be assigned to the process with the minimum burst time requirement.

Priority scheduling algorithm: CPU is assigned to the processes according to the priority given by the user. It is both preemptive and nonpreemptive.

Round-Robin Scheduling algorithm: It is a preemptive scheduling algorithm .This is designed especially for time sharing systems. Here equal priority will be given to all the processes. A small unit of time called time quantum or time slice is defined.CPU will be assigned to each process in a circular queue fashion one time quantum time to

each process. After completion of one time quantum time for a process, CPU will be given to other process. If a process has more burst time than the given time quantum, then it will be added to the tail of the ready queue. In this way all the processes will be treated equally in this Round-Robin scheduling algorithm.
In this paper we are modifying existing Round-Robin algorithm and proposing a new CPU scheduling algorithm called MQRR algorithm ,to minimize average waiting time and average turnaround time.

III. PROPOSED ALGORITHM:MQRR ALGORITHM

1. START
2. If all processes arrived at the same time then continue step3.Otherwise,assign 10ms(milli seconds)burst time to the first process in the queue. Because if no process is arrived by the time it enters ,it has to be assigned CPU for some amount of time.
3. Find average burst time (say ABT)of all the processes which are arrived later.
4. Ready queue is partitioned into two queues. First queue contains the processes whose burst times are less than or equal to ABT in ascending order of their burst times. Second queue contains the processes whose burst times are greater than ABT in ascending order of their burst times. If two processes need same burst time, then follow FCFS to break the tie.
5. Say burst time of the first process in the first queue is FBT. Assign CPU alternatively one process in the first queue with its complete burst time, and one process in the second queue with FBT time until all the processes in the second queue are being assigned CPU one time to each process.
6. If first queue has more processes than the second queue, then after assigning CPU once to each of the processes in the second queue ,then assign CPU to the remaining processes of the first queue with their complete burst time requirement, then schedule second queue.
7. If second queue has more processes than first queue, then assign CPU to the remaining processes with one FBT each.
8. After assigning CPU one time to all the processes in both the queues, now schedule second queue.
9. Find average burst time of the remaining processes in the second queue. Say AVG. Now assign AVG burst time to each process in the second queue, after assigning it, if the remaining burst time of the process is still less than AVG then assign CPU to that process only, otherwise send it to the tail of the second queue.
10. Continue this procedure till all processes in the second queue finishes their task.
11. END

IV. RESULT ANALYSIS

Comparison with Round-Robin Algorithm using three examples by considering burst times in ascending order, descending order and random order burst times and different arrival times.

1.Consider Burst times in ascending order.

Process	Arrival time	Burst time
P1	0	10
P2	1	20
P3	2	30
P4	3	40
P5	4	50
P6	5	60

Table 1 :Processes with different arrival times and burst times in Ascending order.

By using Round-Robin scheduling algorithm: Say Time Quantum is 10ms.

Gantt chart:

P1	P2	P3	P4	P5	P6	P2	P3	P4	P5	P6	P3	P4	P5	P6	P4	P5	P6	P5	P6	P6

0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 18 190 200210

Average Waiting Time:89.1ms
Average Turnaround Time:124.1ms

By using MQRR Algorithm:
1. Assign 10ms to first process P1.Here P1 needs only 10ms ,so it has finished its task. So it has been terminated. Now by considering remaining processes which have arrived into the ready queue, Average Burst time of all the processes P2,P3,P4,P5,P6 is 40.Now First Queue contains all the processes whose burst times are less than or equal to 40ms in ascending order of their burst times. That is First queue contains P2,P3,P4. And second queue contains all the processes whose burst times are greater than 40ms in ascending order of their burst times. That is P5,P6.

2. Now P2 is the first process in the first queue, and it needs 20ms burst time, so according to the algorithm FBT is 20ms.P2 will get 20ms and finishes its task. Then P5 in the second queue will get selected and 20ms will be given to P5.Then P3 in the first queue will get selected and it will get 30ms(that is its complete burst time),now in the second queue P6 wil get selected and it will get 20ms burst time(that is FBT time).Now P4 in the first queue will get selected and it will get 40ms (that is its complete burst time).Here all the processes in the first queue have completed their tasks.

3. Now P5,P6 are in the second queue.And the remaining times for them are 30ms,40ms.Now their AVG is 35ms so each process will get 35ms burst time.P5 will finish its task and after giving 35ms to P6,the remaining burst time requirement is 5ms ,which is less than AVG.So P6 will get complete 40ms burst time and it finishes its task.

Gantt Chart :

P1	P2	P5	P3	P6	P4	P5	P6

0 10 30 50 80 100 140 170 210

Average Waiting Time:69.1ms
Average Turnaround time: 104.1ms

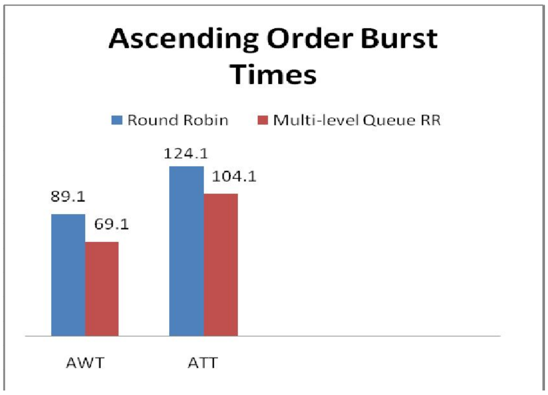

Fig 1:Comparision between Round-Robin and MQRR when burst times are considered in the ascending order and process arrival times are different.

2.Consider Burst times in descending order.

Process	Arrival time	Burst time
P1	0	60
P2	1	50
P3	2	40
P4	3	30
P5	4	20
P6	5	10

Table 2:Processes with different arrival times and burst times in Descending order.

By using Round-Robin scheduling algorithm: Say Time Quantum is 10ms.

Gantt chart:

P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
1	2	3	4	5	6	1	2	3	4	5	1	2	3	4	1	2	3	1	2	1

0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210

Average Waiting Time:114.1ms
Average Turnaround Time:149.1ms

By using MQRR Algorithm:

Assign 10ms to first process P1.Here P1 needs 60ms.so,after assigning 10ms to it, it still requires 50 ms.

Now consider all the processes in the ready queue. Average burst time of P1,P2,P3,P4,P5,P6 is 33.33.According to the algorithm First queue contains the processes in their ascending order ,that is,P6,P5,P4. And P6 is the first process in the first queue, so FBT is 10ms.And second queue contains P3,P1,P2(As P1,P2 needs 50ms each, consider P1 first as it arrived first).Now P6 gets 10ms burst time and completes its task. Then P3 in the second queue will be assigned CPU for 10ms(FBT) burst time, next P5 gets 20ms,next P1 gets FBT time(10ms),next P4 gets 30ms burst time and P2 gets 10ms burst time. Now all the processes in the first queue got finished their tasks. Now remaining burst times of P3,P1,P2 are 30ms,40ms,40ms.So AVG for these processes is 36.66,P3 finishes its task first, then P1 gets 40ms CPU time because after assigning 36.66ms to it, remaining time is less than AVG. So it will get 40ms and also P2 finishes its task as it requires 40ms.

Gantt chart:

P1	P6	P3	P5	P1	P4	P2	P3	P1	P2

0 10 20 30 50 60 90 100 130 170 210

Average Waiting Time:74.1ms
Average Turnaround time: 109.1ms

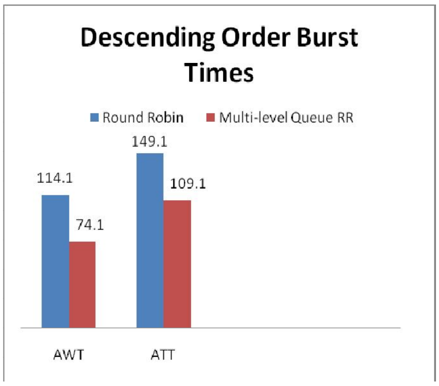

Fig 2:Comparision between Round-Robin and MQRR when burst times are considered in the descending order and process arrival times are different.

3.Consider Random order Burst times .

Process	Arrival time	Burst time
P1	0	40
P2	1	20
P3	2	60
P4	3	10
P5	4	30
P6	5	50

Table 3:Processes with different arrival times and burst times in Random order.

By using Round-Robin scheduling algorithm: Say Time Quantum is 10ms.

Gantt Chart:

P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
1	2	3	4	5	6	1	2	3	5	6	1	3	5	6	1	3	6	3	6	3

0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210

Average Waiting Time:100.8ms
Average Turnaround time: 135.8ms

By using MQRR Algorithm:

Assign 10ms to first process P1.Here P1 needs 40ms.so,after assigning 10ms to it,it still requires 30 ms.

Now consider all the processes in the ready queue. Average burst time of P1,P2,P3,P4,P5,P6 is 33.33.According to the algorithm First queue contains the processes in their ascending order ,that is,P4,P2,P1,P5. And second queue contains P6,P3.And First process in the first queue is P4,so FBT is 10ms.First P4 gets 10ms and finishes its task, after that P6 gets FBT that is 10ms,thenP2 gets 20ms ,after that P3 gets 10ms.By now all the processes in the second queue got assigned CPU once but still First queue processes P1,P5 did not get assigned CPU. So according to the algorithm CPU will be assigned to P1,P5 with their complete burst time requirements 30ms,30ms.Now second queue need to be scheduled. Remaining times of P6,P3 are 40ms,50ms.AVG is 45ms.So P6,P3 will finish their tasks one after the other.

Gantt Chart:

P1	P4	P6	P2	P3	P1	P5	P6	P3

0 10 20 30 50 60 90 120 160 210

Average Waiting Time:70.8ms
Average Turnaround time: 105.8ms

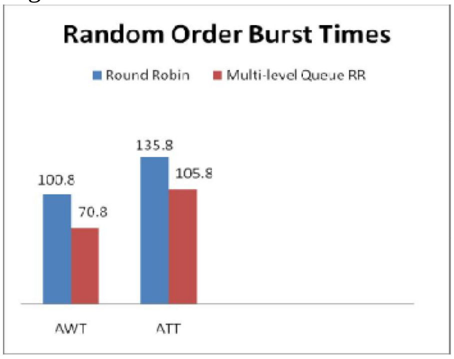

Fig 3:Comparision between Round-Robin and MQRR when burst times are considered in the Random order and process arrival times are different.

Now, we consider all the processes arrived at time 0. Consider the following example where processes P1, P2 ,P3, P4, P5, P6 have arrived into the ready queue at time 0:

1.Consider Burst times in ascending order.

Process	Burst time
P1	10
P2	20
P3	30
P4	40
P5	50
P6	60

Table 4:Processes with equal arrival times and burst times in Ascending order.

By using Round-Robin scheduling algorithm: Say Time Quantum is 10ms.

Gantt chart:

P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
1	2	3	4	5	6	2	3	4	5	6	3	4	5	6	4	5	6	5	6	6

0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180190200210

Average Waiting Time:91.6ms
Average Turnaround time: 126.6ms

By using MQRR Algorithm:

According to the algorithm when all processes arrives at the same time,first compute average burst time of all the processes.Here AVG is 35ms.Now send the processes which have burst time greater than AVG into second queue.First queue contains P1,P2,P3 and second queue contains P4,P5,P6.Here P1 is the first process, so FBT is 10ms.First P1 will get 10ms and finishes its task,next P4 will get FBT time,that is 10ms then P2 will get 20ms,and finishes its task.Then P5 will get FBT time,that is 10ms,then P3 will get 30ms and finishes its task.Now remaining process burst times in second queue are,P4 need 30ms,P5 needs 40ms and P6 needs 50ms.Now AVG of these processes is 40ms.So P4,P5,P6 will finish tasks accordingly.

Gantt chart:

P1	P4	P2	P5	P3	P6	P4	P5	P6

0 10 20 40 50 80 90 120 160 210

Average Waiting Time:68.3ms
Average Turnaround time: 103.3ms

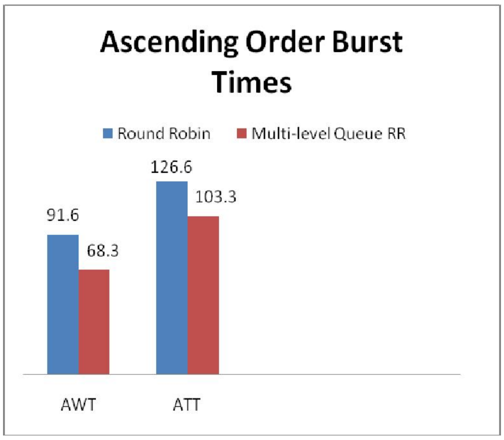

Fig 4:Comparision between Round-Robin and MQRR when burst times are considered in the ascending order and process arrival times are equal.

1.Consider Burst times in ascending order.

Process	Burst time
P1	60
P2	50
P3	40
P4	30
P5	20
P6	10

Table 5:Processes with equal arrival times and burst times in Descending order.

By using Round-Robin scheduling algorithm: Say Time Quantum is 10ms.

Gantt chart:

P1	P2	P3	P4	P5	P6	P1	P2	P3	P4	P5	P1	P2	P3	P4	P1	P2	P3	P1	P2	P1

0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210

Average Waiting Time:116.6ms

Average Turnaround Time:151.6ms

By using MQRR Algorithm:

According to the algorithm when all processes arrives at the same time,first compute average burst time of all the processes.Here AVG is 35ms.Now send the processes which have burst time greater than AVG into second queue.First queue contains P6,P5,P4 and Second queue contains P3,P2,P1.Here P6 is the first process,so FBT is 10ms.First P6will get 10ms and finishes its task,next P3 will get FBT time,that is 10ms then P5 will get 20ms,and finishes its task.Then P2 will get FBT time,that is 10ms,then P4 will get 30ms and finishes its task.Now remaining process burst times in second queue are,P3 need 30ms,P2 needs 40ms and P1 needs 50ms.Now AVG of these processes is 40ms.So P3,P2,P1 will finish tasks accordingly.

Gantt chart:

P6	P3	P5	P2	P4	P1	P3	P2	P1

0 10 20 40 50 80 90 1 160 210

Average Waiting Time:68.3ms

Average Turnaround Time:103.3ms

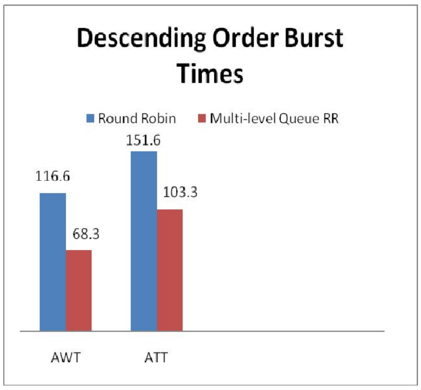

Fig 5:Comparision between Round-Robin and MQRR when burst times are considered in the descending order and process arrival times are equal.

3.Consider Random Burst times .

Process	Burst time
P1	40
P2	20
P3	60
P4	10
P5	30
P6	50

Table 6:Processes with equal arrival times and burst times in Random order.

By using Round-Robin scheduling algorithm:Say Time Quantum is 10ms.

Average Waiting Time:103.3ms

Average Turnaround time: 138.3ms

By using MQRR:

According to the algorithm when all processes arrives at the same time,first compute average burst time of all the processes.Here AVG is 35ms.Now send the processes which have burst time greater than AVG into second queue.First queue contains P4,P2,P5and Second queue contains P1,P6,P3.Here P4 is the first process,so FBT is 10ms.First P4will get 10ms and finishes its task,next P1 will get FBT time,that is 10ms then P2 will get 20ms,and finishes its task.Then P6 will get FBT time,that is 10ms,then P5 will get 30ms and finishes its task.Now remaining process burst times in second queue are,P1 need 30ms,P6 needs 40ms and P3 needs 50ms.Now AVG of these processes is 40ms.So P1,P6,P3 will finish their tasks accordingly.

Gantt chart:

P4	P1	P2	P6	P5	P3	P1	P6	P3

0 10 20 40 50 80 90 120 160 210

Average Waiting Time:68.3ms

Average Turnaround time: 103.3ms

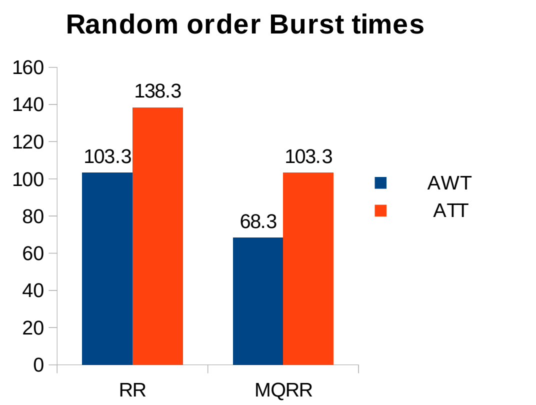

Fig 6:Comparision between Round-Robin and MQRR when burst times are considered in the Random order and process arrival times are equal.

CONCLUSION

We have many other modified Round-Robin Algorithms[3][5].If we compare this MQRR with Smart Optimized Round Robin (SORR) CPU Scheduling Algorithm, in SORR ,there is no procedure to calculate average waiting time and average turnaround time when the arrival times of the processes are different. When we compared MQRR with Round-Robin Algorithm, we got less average waiting times and less average turnaround times in all the three different cases of burst times. We considered different arrival times of the processes and equal arrival times of the processes, in both cases MQRR is giving better results than Round Robin in terms of AWT and ATT. By using MQRR, when we considered equal arrival times, we have observed that ,in all the 3 different cases(Ascending,descending,random order burst times),we got same average waiting times and same average turnaround times.In time sharing systems,where we need good response time for all the processes,MQRR is a good choice than Round Robin.Because we are using two queues one for short processes and one for long processes,and giving priority to both the queues.One more advantage of MQRR is selection of Time quantum is done dynamically.

REFERENCES

[1] Shweta Jain1, Dr. Saurabh Jain2 ," A Review Study on the CPU Scheduling Algorithms",International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Vol. 5, Issue 8, August 2016 Copyright to IJARCCE DOI 10.17148/IJARCCE.2016.5805 22

[2] Imran Qureshi,"CPU Scheduling Algorithms: A Survey"Advanced Networking and Applications Volume: 05, Issue: 04, Pages:1968-1973 (2014) ISSN : 0975-0290

[3] Rahul Joshi*, 2Shashi Bhushan Tyagi ,"Smart Optimized Round Robin (SORR) CPU Scheduling Algorithm " Volume 5, Issue 7, July 2015 ISSN: 2277 128X ,International Journal of Advanced Research in Computer Science and Software Engineering

[4] A., Silberschatz, J. L., Peterson, and P.B., Galvin, "Operating System Concepts," Addison Wesley, 7ED 2006

[5] Abdulraza, Abdulrahim,Salisu Aliyu, Ahmad M Mustapha, Saleh E Abdullahi, " An Additional Improvement in Round Robin (AAIRR) CPU Scheduling Algorithm," International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 2, February 2014 ISSN: 2277 128X.

[6] Silberschatz, A., Galvin, P. and Gagne, G. Operating System Concepts, International Student Version, Ed.8, India, John Wiley and Sons, Inc. 2010. Stalling, W. Operating System, Ed.5, New Delhi, Pearson Education, Singapore, Indian Edition, 2004.

[7] Tanenbaum, A. and Woodhull, A., S. Operating system, Ed. 8, New Delhi ,Prentice Hall of India, 2000.

[8] Dhamdhere, D., M. System Programming and Operating Systems, Revised Ed. 8, New Delhi, Tata McGraw-Hill Education Pvt. Ltd., 2009.

[9] Dhakad, V., K. and Sharma, L. Performance analysis of Round Robin scheduling using adaptive approach based on smart time slice and comparison with SRR. International Journal of Advances in Engineering & Technology ©IJAET, May 2012, Vol. 3, No. 2, pp. 333-339.

★★★