An Improved Round Robin CPU Scheduling Algorithm

NEHA[1], ANKITA JIYANI[2]
[1]Student, Department of Computer Science & Technology, Poornima College of Engineering
[2]Assistant Professor, Department of Computer Science & Technology, Poornima College of Engineering

Abstract -- CPU is one of the most important component of the computer which is used for scheduling the process known as CPU Scheduling. It is the process of scheduling the process one after another while one is on hold and other is in ready state. Many types of scheduling algorithms are there like First-Come-First-Served (FCFS), Shortest Job First (SJF), Priority Scheduling and Round Robin which basically focus on maximizing throughput and CPU utilization and minimizing response time, turnaround time, number of context switching and waiting time. In this paper we proposed an improved round robin CPU scheduling algorithm. The result shows that improved round robin Scheduling is always giving better result than the conventional Round Robin Scheduling Algorithm.

Index Terms: Round Robin Scheduling, Response Time, Time Quantum, Burst Time, Waiting Time, Turnaround Time.

I. INTRODUCTION

The primary computer resource is CPU. There are basically four main function of CPU out of which scheduling is one of them. And Scheduling is central to the operating system design [1]. It is basic activity for multi programmed Operating system. Selecting an appropriate distribution of CPU and I/O bound programs is crucial for CPU scheduling [2]. CPU scheduler decides which of the processes in the ready queue is to be allocated the CPU. Ready queue can be maintained either as a FIFO queue, Priority queue, tree of any other unordered linked list.CPU scheduling can be preemptive or non-preemptive type.

According to Seltzer, M P. Chen and J Outer out 1990 [4], the last thirty years have seen an enormous amount of research in the area of disk scheduling algorithm. The core objective has been developed scheduling algorithms suited for certain goals sometimes with provable properties.

Round robin scheduling (RRS) is a job-scheduling algorithm that is considered to be very fair, as it uses time slices that are assigned to each process in the queue or line.[6] Each process is then allowed to use the CPU for a given amount of time, and if it does not finish within the allotted time[8], it is preempted and then moved at the back of the line so that the next process in line is able to use the CPU for the same amount of time.

II. TYPES OF CPU SCHEDULING ALGORITHMS

These algorithms are used to allocate jobs or resources to the CPU for their execution and to enhance the performance of CPU. Some of the popular CPU scheduling algorithms are First-Come-First-Served (FCFS), Priority Scheduling, Round Robin (RR) and Shortest Job First (SJF).FCFS is the most simple Scheduling Algorithm in which the process arrived in the ready queue are processed in firstly or we can say in the order process arrive they are executed in the same order that'swhy, it is named as First-Come-First-Served. In priority scheduling the processor is allocated to the process having the highest priority. In Round robin scheduling a small unit of time slice is allocated to every process present in the ready queue. Then each process is allowed to use the CPU for a given amount of time. In Shortest job first the job having the shortest burst time in the ready queue is executed first. If burst time and arrival time is same in any two jobs then the scheduling is done on the first come first served basis.

III. ROUND ROBIN CPU SCHEDULING ALGORITHM

Following is the proposed RR CPU scheduling algorithm:-

Step 1. START

Step 2. Make a ready queue of the Processes like READYQ.

Step 3. REPEAT steps 4, 5 and 6 TILL queue READYQ becomes empty.

Step 4. Select the first process from the ready queue and assign the CPU for the given time quantum.

Step 5. After complete execution of current process, removed it from the ready queue and go to step 3.

Step 6. Remove the currently running process from the ready queue READYQ and put it at the end of the ready queue.

Step 7. STOP

IV. IMPROVED ROUND ROBIN CPU SCHEDULING ALGORITHM

As mentioned, Round Robin Scheduling algorithm which is used for executing the jobs on processor with a small unit of time slice. Improved Round Robin Scheduling Algorithm which is almost similar to Round Robin scheduling algorithm using time slice but little different in the execution of jobs on processor. In improved round robin scheduling algorithm we take the first process in the ready queue and execute it till the allocated time quantum then we check the remaining burst time of the running process. If the remaining burst time is less than the allocated time quantum then the running process is executed again till it finishes. But, if the remaining burst time of the running process is more than the allocated time quantum then the next process in the ready queue is executed. And we remove the currently running process from the ready queue and put it at the tail of the ready queue. Hence, the improved round robin scheduling work.

Following is the proposed IRR CPU scheduling algorithm:-

Step 1. START

Step 2. Make a ready queue of the Processes like READYQ.

Step 3. Repeat steps 4, 5 and 6 TIL queue READYQ becomes empty.

Step 4. Select the first process from the ready queue and assign it to the CPU for the given time quantum.

Step 5. If the remaining CPU burst time of the currently running process is less than allocated time quantum then assign CPU again to the currently running process for remaining CPU burst time. After complete execution of current process, removed it from the ready queue and go to step 3.

Step 6. Remove the currently running process from the ready queue READYQ and put it at the end of the ready queue.

Step 7. STOP

V. EXPERIMENTAL ANALYSIS

Assumptions:

For fair evaluation of performance, we assumed that all the process have equal priority and the attribute like arrival time (a_i), burst time (b_i) , time slice and number of process (p_i) are known before assigning the processes to the processor. The context switching is 0.

Experimental Analysis:

For evaluating improved round robin algorithm we have taken two different cases. In first case the arrival time is 0 for all the process and the burst time is taken randomly. In second case the arrival time is non- zero. And burst time is taken randomly for all the process.

CASE 1- Zero arrival time and Random burst time

In this case arrival time has been considered zero and CPU burst time has been taken in random order. Time quantum is 10 milliseconds. Table 1 shows the processes and their arrival and burst time while table 2 shows the comparison result of RR and proposed IRR. The figure 1 shows the gantt chart representation of RR while figure 2 shows the gantt chart representation of proposed IRR and the graph shows the difference of waiting time and turnaround time between RR and IRR.

Table 1: processes and their burst time

Process (p_i)	Arrival Time (a_i)	Burst Time(b_i)
P1	0	15
P2	0	5
P3	0	25
P4	0	12
P5	0	30

P1	P2	P3	P4	P5	P1	P3	P4	P5	P3	P5

0 10 15 25 35 45 50 60
62 72 77 87

Figure 1: Gantt chart representation of RR

P1	P1	P2	P3	P4	P4	P5	P3	P3	P5	P5

0 10 15 20 30 40 42 52
62 67 77 87

Figure 2: Gantt chart representation of IRR

Waiting time = Completion time(c_i) - (arrival time(a_i) + burst time(b_i))

Average Waiting time = Waiting time of all process/ total number of process

Waiting time of processes for RR are :-

Process	Waiting time= c_i - (a_i – b_i)	Result
P1	50 – (0+15)	35
P2	15-(0+5)	10
P3	77-(0+25)	52
P4	62-(0+12)	50
P5	87-(0+30)	57

Average Waiting time of RR = 35+10+52+50+57/5=40.8

Average Turnaround = Turnaround time of all process/ total number of process

Turnaround time = (Completion time - Arrival time)

Turnaround time of processes for RR are :-

Process	Turnaround time= c_i - a_i	Result
P1	50 – 0	50
P2	15 - 0	15
P3	77 - 0	77
P4	62 - 0	62
P5	87 - 0	87

Average Turnaround time of RR= 50+15+77+62+87/5 = 58.2

Waiting time of processes for IRR are :-

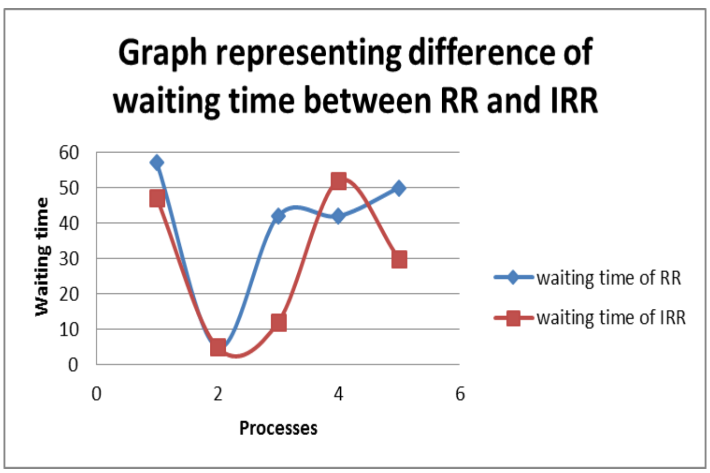
Process	Waiting time= c_i - (a_i – b_i)	Result
P1	15 – (0+15)	0
P2	20-(0+5)	15
P3	67-(0+25)	42
P4	42-(0+12)	30
P5	87-(0+30)	57

Average Waiting time of IRR = 0+15+42+30+57/5=28.8

Turnaround time of processes for IRR are :-

Process	Turnaround time= c_i - a_i	Result
P1	15 – 0	15
P2	20 – 0	20
P3	67 – 0	67
P4	42 – 0	42
P5	87 – 0	87

Average Turnaround time of IRR= 15+20+67+42+87/5 = 46.2



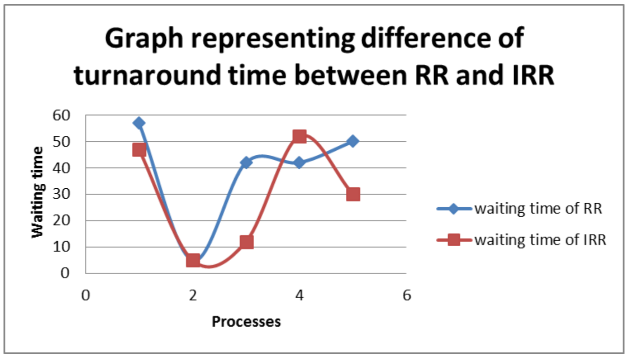


Table 2: Comparison of RR and IRR

Algorithm	Average waiting time(ms)	Average turnaround time(ms)
Round Robin	40.8	58.2
Improved Round Robin	28.8	46.2

CASE 2- Non- Zero arrival time and Random burst time

In this case arrival time has been considered non-zero and CPU burst time has been taken in random order. Time quantum is 10 milliseconds. Table 3 shows the processes and their arrival and burst time while table 4 shows the comparison result of RR and proposed IRR. The figure 3 shows the gantt chart representation of RR while figure 4 shows the gantt chart representation of proposed IRR and the graph shows the difference of waiting time and turnaround time between RR and IRR.Table 3: processes and their burst time

Process	Arrival Time	Burst Time
P1	0	25
P2	5	10
P3	8	12
P4	10	20
P5	12	15

P1	P2	P3	P4	P5	P1	P3	P4	P5	P1

0 10 20 30 40 50 60 62
72 77 82

Figure 3: Gantt chart representation of RR

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

0 10 20 30 32 42 52 57
67 72 82

Figure 4: Gantt chart representation of IRR

Waiting time of processes for RR are :-

Process	Waiting time= c_i - (a_i – b_i)	Result
P1	82 – (0+25)	57
P2	20-(5+10)	5
P3	62-(8+12)	42
P4	72-(10+20)	42
P5	77-(12+15)	50

Average Waiting time of RR = 57+5+42+42+50/5=39.2

Turnaround time of processes for RR are :-

Process	Turnaround time= c_i - a_i	Result
P1	82 – 0	82
P2	20 – 5	15
P3	62 – 8	54
P4	72 – 10	62
P5	77 – 12	65

Average Turnaround time of RR= 82+15+54+62+65/5 = 55.6

Waiting time of processes for IRR are :-

Process	Waiting time= c_i - (a_i – b_i)	Result
P1	72 - (0+25)	47
P2	20-(5+10)	5
P3	32-(8+12)	12
P4	82-(10+20)	52
P5	57-(12+15)	30

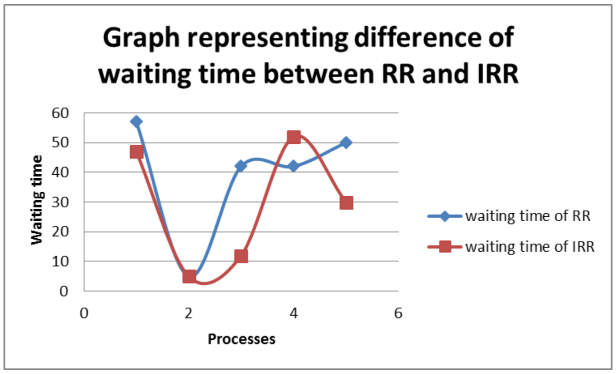
Average

Waiting time of IRR = 47+5+12+52+30/5= 29.2Turnaround time of processes for IRR are :-

Process	Turnaround time = c_i - a_i	Result
P1	72 – 0	72
P2	20 – 5	15
P3	32 – 8	24
P4	82 – 10	72

P5	57 - 12	45

Average Turnaround time of IRR= 72+15+24+72+45/5 = 45.6



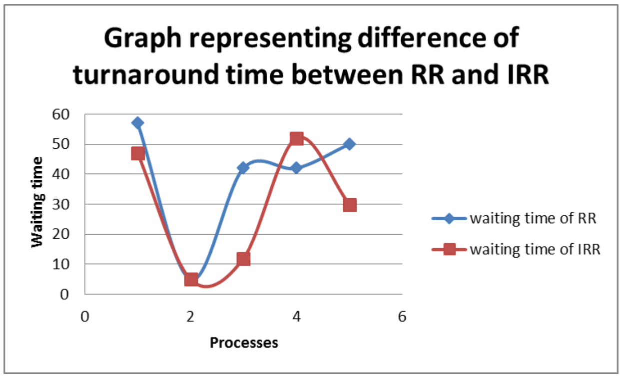


Table 4: Comparison of RR and IRR

Algorithm	Average waiting time(ms)	Average turnaround time(ms)
Round Robin	39.2	55.6
Improved Round Robin	29.2	45.6

VI. CONCLUSION

The allocation of CPU to the processes is one the most important task of the operating system. An improved round robin scheduling algorithm with varying parameters like burst time and time quantum proposed in this paper giving better performance than the conventional Round Robin algorithm. The waiting time and the turnaround time have been reduced in the proposed improved round robin algorithm. The proposed algorithm can be implemented to improve the performance of the system. Simulation results also prove the correctness of the theoretical results.

REFERENCES

[1] Rakesh Kumar Yadav, Abhishek K Mishra, Navin Prakash and Himanshu Sharma, "An Improved Round Robin Scheddling Algorithm for CPU Scheduling", International Journal on Computer Science and Engineering, Vol. 02, No. 04, 2010, pp. 1064-1066.

[2] Abbas Noon, Ali Kalakech, and Seifedine Kadry, "A New Round Robin based Scheduling Algorithm for Operating Systems: Dynamic Quantum Using the Mean Average", International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1, May 2011, pp. 224-229.

[3] H.S. Behera, Rakesh Mohanty, Sabyasachi Sahu, Sourav Kumar Bhoi, "Design and performance evaluation of multi cyclic round robin(MCRR) algorithm using dynamic time quantum" Journal of global research in computer science(ISSN-2229-371X), volume 2, No.2, February 2011.

[4] Seltzer, M P. Chen and J outerhout, 1990, Disk scheduling revisited in USENIX, Winter technical conference.

[5] A. Silberschatz, P. B. Galvin, and G. Gagne, "Operating System Concepts", 7th Edn., John Wiley and Sons Inc, 2005, ISBN 0-471-69466-5.

[6] M.K. Srivastav, Sanjay Pandey, Indresh Gahoi and Neelesh Kumar Namdev, "Fair Priority Round Robin with Dynamic Time Quantum", International Journal of Modern Engineering Research, Vol. 02, Issue. 03, May-June 2012, pp. 876-881.

[7] H.S. Behera, Rakesh Mohanty, Jajnaseni Panda, Dipanwita Thakur and Subasini Sahoo, "Experimental Analysis of a New Fare-Share Scheduling Algorithm with Waited Time Slice for Real Time Systems", Journal of Global Research in Computer Science, Vol. 2, No. 2, February 2011, pp. 54-60.

[8] Rami J. Matarneh,"Self-Adjustment Time Quantum in Round Robin Algorithm Depending on Burst Time of Now Running Processes", American J. of Applied Sciences 6(10):1831-1837,2009.

[9] Sunita Mohan,"Mixed Scheduling (A New Scheduling Policy)", Proceedings of Insight'09, 25-26 November 2009.