Volume 5, Issue 3, March 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Improved Round Robin with Shortest Job First Scheduling

[1]Radhe Shyam, [2]Parmod Kumar

[1]Extension Lecturer (Computer Science) Govt. College for Women, Hissar, Haryana- India

[2]MCA, GJUS&T, Hissar, Haryana- India

Abstract: One of the most important components of the computer resource is the CPU. CPU scheduling is the basis of operating systems. Scheduling is the technique used for controlling the order of process which is to be performed by a CPU of a computer. Some of the popular CPU schedulingare First Come First Served (FCFS), Shortest Job First (SJF), Priority Scheduling and Round Robin (RR). Most CPU scheduling algorithms concentrate on maximizing CPU utilization and throughput and minimizing turnaround time, response time, waiting time, and number of context switching for a set of requests. In this paper, a new Improved Round Robin Scheduling is designed. This Scheduling algorithms gives better result compare to Round Robin (RR), Improved Round Robin (IRR), Enhanced Round Robin (ERR), Self Adjustment Round Robin (SARR), FCFS and some other scheduling algorithm.

Keywords: Operating System, Scheduling,Context switch, Round Robin,Turnaround time,Waiting time,FCFS.

I. INTRODUCTION

An operating system is a software act as an intermediary between the user of a computer and computer hardware. The purpose of an O.S is to provide an environment in which a user can execute programs in efficient manner. An operating system provided functionality is device management, memory management, file management, CPU Scheduling,process management, protection and security.

A. Scheduling:

Scheduling is also a fundamental function of operating system. Almost all Computer devices and System are scheduled before use. As CPU is one of the primary computer resources. Thus its scheduling is central to O.S designs. Scheduling is the strategy by which the system decides which task should be executed at any given time. Whenever the CPU became ideal the O.S must select one of the processes in the ready queue to be executed.

B. Scheduling Criteria:

CPU Utilization – This is a measure of how much busy the CPU is. As we want to proper use of CPU thus the scientific developed technique is used. CPU utilizations can range from 0 to 100 percent.

Throughput- Throughput is the no. of processes completed per time period. For long processes this rate may be two process per hour. For short process throughput might be 15 processes per second.

Waiting time **-**Waiting time is the sum of periods spends waiting in the ready queue. CPU executes only one job at a time. The rest of process wait for the CPU.

Turnaround time **-** It is the total time taken to execute a job. The interval from the time of submission of a process to the time of completion is the turnaround time.

Response time**-** Response time is the time from the submission of a request until the first response is produes.

Context Switch**–** A Context switch is process of storing and restoring context of a preempted process, so that execution can be resumed from same point at a later time.

C. Better CPU Scheduling criteria are following.

- MaximizeThroughput andCPU utilization.
- Minimize Turnaround Time, Response time,Waiting time and context Switch.

II. RELATED WORKS

In this Review of literature the work done by the earlier authors in direction of the process scheduling and the process management is described. The work done by these authors is explained here under.

In Year 2012, Manish kumar Mishra [4] describes an improvement in RR. IRR picks the first process from the ready queue and allocate the CPU to it for a time interval of up to 1 time quantum. After completion of process's time quantum, it checks the remaining CPU burst time of the currently running process. If the remaining CPU burst time of the currently running process is less than 1 time quantum, the CPU again allocated to the currently running process for remaining CPU burst time. In year 2009, Rami J. Matarneh [5] performed a work Self-Adjustment-Round-Robin (SARR) based on a new approach called dynamic time quantum, in this approach the time quantum repeatedly adjusted according to the burst time of the now-running processes. In year 2011, SarojHiranwal [6] in which first of all we arrange the processes according to the execution time/burst time in increasing order that is smallest the burst time higher the priority of the running process. The smart time slice is equal to the mid process burst time of all CPU burst time. In year 2011**,** H.S. Behera [7] proposes a newly improved process scheduling algorithm by using dynamic time quantum along with weighted mean. In year 2013**,**LalitKishor& Dinesh Goyal [8] proposed a median based time quantum scheduling algorithm which is combination of SJF & RR. In year 2012, H.S. Behera [9] proposed the summation of mean and standard deviation based time quantum scheduling algorithms which is combination of SJF & RR. In year 2013, AashnaBisht [10] performed a work Enhanced Round Robin(ERR), in which modify the time quantum of only those processes which require a slightly greater time than the allotted time quantum cycle. The remaining process will be executed in the conventional Round robin manner. In year 2011, RakashMohanty&Manas Das [11] performed a work in which a new variant of Round Robin scheduling algorithms by executing the processes according to the new calculated Fit factor 'f' and using the concept of dynamic time quantum. In year 2012, DebashreeNayak[12] performed a work in which a median plus some other value are added in time quantum. This scheduling which is combination of SJF &RR.In year 2012**,**Ishwari Singh Rajput, Deepa Gupta[14] proposed priority based Round-robin CPU scheduling algorithms is based on the integration of round robin and priority scheduling. It retains the advantage of round robin in reducing starvation and also integrates the advantage of priority scheduling. The Proposed new algorithm also implement the concepts of aging by assigning new priorities to the processes. In year 2012, P.SurendraVarma[15] performed a work, In which the improved version of SRBRR (Shortest Remaining Burst Round Robin) by assigning the processor to processes with shortest remaining burst in round robin manner using the best possible time quantum. In this paper the time quantum is computed with the help of median and highest burst time. In year 2012, H.S. Behera&Brajendra Kumar Swain[16] performed a work it gives precedence to all processes according to their priority and burst time, then applies the Round Robin algorithm on it. This Proposed algorithm is developed by taking dynamic mean time quantum in to account. In year 2012, Brajendra Kumar Swain, H.S Behera and Anmol Kumar Parida[17] in this paper the new algorithm used Round Robin with Highest Response Ratio next scheduling, which uses Highest Response Ratio criteria for selecting processes from ready queue. In year 2010, RakeshkumarYadav,NavinPrakash and Himanshu Sharma [19] performed a work. In this paper the new algorithm are used Round Robin with shortest Job first scheduling. In which allocate all processes to the CPU only one time as like present RR Scheduling algorithm. After it we select shortest job from the waiting queue and it shortest job assign to the CPU. After that next shortest job are selected. In year 2012, Ali JbaeerDawood [20] performed a work. The processes were ascending with shortest remaining burst time and calculate the TQ from multiply the average summation of minimum and maximum BT by (80) percentage.

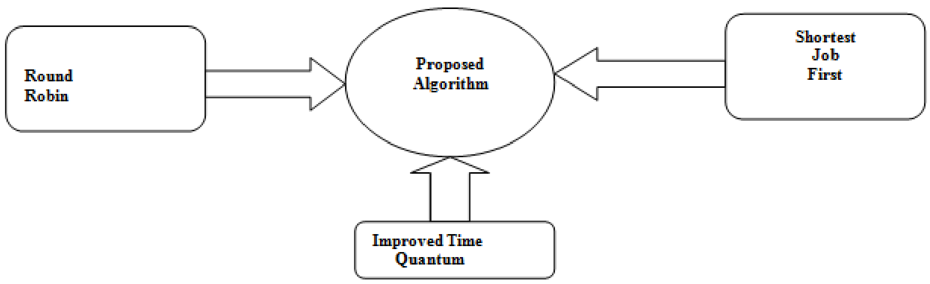
III. PROPOSED ALGORITHM

CPU schedulingis the basis of multi programming O.S.Most CPU scheduling algorithm sconcentrateonmaximizingthroughputand CPU utilizationandminimizingwaiting time, response time, turnaround time and number of context switchingforasetofrequests. Some of the CPU scheduling algorithms are First-Come-First-Served(FCFS), PriorityScheduling,ShortestJob First(SJF)andRoundRobin(RR). FCFS is the simple stformof CPU scheduling algorithm. These algorithms are easyto implement, butitgenerallydoesnotprovidethebestservice. Round Robinbeingthemostpopularintimeshared operatingsystem, butitmaynotbe suitable forreal time operating systemsbecauseoflarge turnaround time, waitingtimeandhighnumberofcontextswitches. This paper describesan improvementinRR.

The proposed IRRSJF CPU scheduling algorithm

1. Enter the burst time of process in Ready queue.
2. Arrange all the processes present in ready queue in ascending order of their burst time.
3. Do steps 4, 5 and 6 WHILE Ready Queue becomes empty and If new process are enter in the Ready Queue then control go to step 2.
4. Pick the first ascending order sorted process from the ready queue and allocate the CPU to it for a time interval of up to 1 time quantum.
5. If the remaining CPU burst time of the currently running process is less than 1 time quantum then allocate CPU again to the currently running process for remaining CPU burst time. After completion of execution, removed it from the ready queue and go to step 3.
6. Remove the currently running process from the ready queue and put it at the tail of the Ready Queue.

The presented algorithmic approach is a hybrid model that collected three main approaches shown in figure 1.



In this paper the proposed algorithm are used Round Robin with shortest Job first scheduling. The TQ studied to improve the efficiency of RR and performs degrades with respect to context switching, Average Wait Time and Average turned around time. The processes were ascending with shortest remaining burst time and then TQ are given to that ascending process to CPU. Then the proposed algorithm performs better than Round Robin (RR), Improved Round Robin (IRR), FCFS and some other scheduling algorithm in terms of reducing the number of context switches, average waiting time and average turnaround time.

IV. ILLUSTRATION

Suppose five processes arriving time=0, and CPU burst time is (P1=65, P2=35, P3=15, P4=55, P5=80). Then the processes are sorted in ascending order which results in sequence P3=15, P2=35, P4=55, P1=65, P5=80. Time quantum is 20 assumed. After first iteration the remaining CPU burst time sequence is P3=0, P2=0,P4=35,P1=45,P5=60. In this case, processes P3 and P2 are deleted from the ready queue. If that time new process arrived suppose that are P6=25 and arrival time is 110 thenthe processes are sorted again in ascending order which results in sequence P6=25, P4=35, P1=45, P5=60. After second iteration the remaining CPU burst time sequence is P6=0, P4=0,P1=25,P5=40. In this case, processes P6, P4 are deleted from the ready queue. If third iteration no new process arrived then IRR are worked. Since, now there is no process in ready queue. Then calculates the ATT and AWT. In this example, AWT=85.8, ATT=146.6. Gantt chart of this example are following.

P3(1	P2(20	P2(15	P4(20	P1(20	P5(20	P6(20	P6(5)	P4(20	P4(15	P1(20	P5(20	P1(20	P1(5)	P5(20	P5(20

V. EXPERIMAENTAL ANALYSIS

In every case we will compare the result of the proposed IRRSJF method with Round Robin, Improved RR, Enhanced RR and Self adjustment RR (SARR). Here we taken 20 as the static time quantum (TQ) for IRRSJF, RR, IRR and ERR algorithm. In SARR algo.mean time quantum is used.

Case1:CPU burst time in increasing order-Let's consider four processes(P1,P2,P3,P4) with arrival time=0 and burst time(22,35,52,80) as shown in table 1. Table 2 shows the output using 5 algo. and Figure 3 shows Gantt chart.

TABLE 1

PROCESS NO.	ARRIVAL TIME	BURST TIME
P1	0	22
P2	0	35
P3	0	52
P4	0	80

TABLE 2

ALGORITHM	TIME QUANTUM	AVG. WAITING TIME	AVG. TURANAROUND
IRRSJF	20	52	99.25
RR	20	57	129.25
IRR	20	52	99.25
ERR	20	67	114.25
SARR	43	57.75	105

TABLE 3

IRRSJF

P1(20)	P1(2)	P2(20)	P2(15)	P3(20)	P4(20)	P3(20)	P3(12)	P4(20)	P4(20)	P4(20)

0 20 22 42 57 77 97 117 129 149 169 189

RR

P1(20)	P2(20)	P3(20)	P4(20)	P1(2)	P2(15)	P3(20)	P4(20)	P3(12)	P4(20)	P4(20)

0 20 40 60 80 82 97 117 137 149 169 189

IRR

P1(20)	P1(2)	P2(20)	P2(15)	P3(20)	P4(20)	P3(20)	P3(12)	P4(20)	P4(20)	P4(20)

0 20 22 42 57 77 97 117 129 149 169 189

ERR

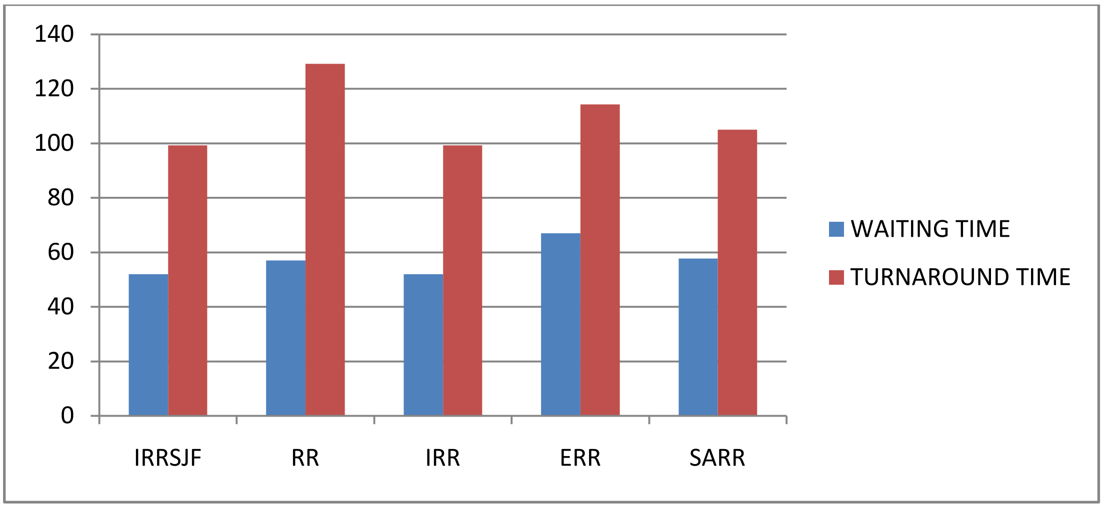
P1(20)	P1(2)	P2(20)	P3(20)	P4(20)	P2(15)	P3(20)	P4(20)	P3(12)	P4(20)	P4(20)

0 20 22 42 62 82 97 117 137 149 169 189

SARR

P1(22)	P2(35)	P3(43)	P4(43)	P3(9)	P4(37)

0 22 57 100 143 152
189



COMPARITIVE BARCHART OF WAITING TIME AND TURANAROUND TIME IN CASE 1

Case2:CPU burst time in decreasing order-Let's consider five processes(P1,P2,P3,P4) with arrival time=0 and burst time(80,52,35,22) as shown in table 4. Table 5 shows the output using 5 algo. and Figure 6 shows Gantt chart.

TABLE 4

PROCESS NO.	ARRIVAL TIME	BURST TIME
P1	0	80
P2	0	52
P3	0	35
P4	0	22

TABLE 5

ALGORITHM	TIME QUANTUM	AVG. WAITING TIME	AVG. TURANAROUND
IRRSJF	20	52	99.25
RR	20	110.25	158.5
IRR	20	80.25	127.75
ERR	20	97	144.25
SARR	43	111	158.25

TABLE 6

IRRSJF

P4(20)	P4(2)	P3(20)	P3(15)	P2(20)	P1(20)	P2(20)	P2(12)	P1(20)	P1(20)	P1(20)

0 20 22 42 57 77 97 117 129 149 169
189

RR

P1(20)	P2(20)	P3(20)	P4(20)	P1(20)	P2(20)	P3(15)	P4(2)	P1(20)	P2(12)	P1(20)

0 20 40 60 80 100 120 135 137 159 169
189

IRR

P1(20)	P2(20)	P3(20)	P3(15)	P4(20)	P4(2)	P1(20)	P2(20)	P2(12)	P1(20)	P1(20)

0 20 40 60 75 95 97 117 137 149 169
189

ERR

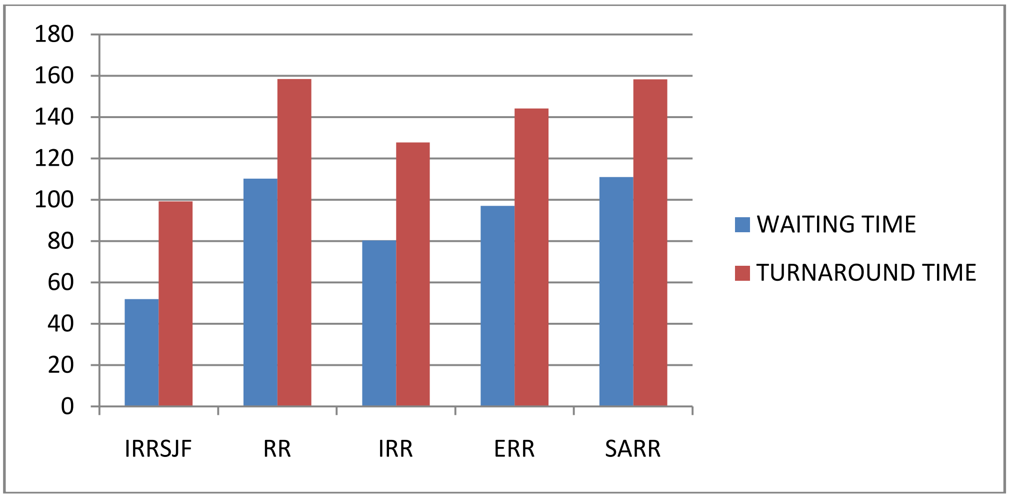
P1(20)	P2(20)	P3(20)	P4(20)	P4(2)	P1(20)	P2(20)	P3(15)	P1(20)	P2(12)	P1(20)

0 20 40 60 80 82 102 122 137 157 169
189

SARR

P1(43)	P2(43)	P3(35)	P4(22)	P1(37)	P2(9)

0 43 86 121 143 180
189



COMPARITIVE BARCHART OF WAITING TIME AND TURANAROUND TIME IN CASE 2

Case3:CPU burst time in random order-Let's consider five processes(P1,P2,P3,P4) with arrival time=0 and burst time(52,22,35,80) as shown in table 7. Table 8 shows the output using 5 algo. and Figure 9 shows Gantt chart.

TABLE 7

PROCESS NO.	ARRIVAL TIME	BURST TIME
P1	0	52
P2	0	22
P3	0	35
P4	0	80

TABLE 8

ALGORITHM	TIME QUANTUM	AVG. WAITING TIME	AVG. TURANAROUND
IRRSJF	20	52	99.25
RR	20	92	139.25
IRR	20	62	109.25
ERR	20	102	124.25
SARR	28	79.25	126.5

TABLE 9

IRRSJF

P2(20)	P2(2)	P3(20)	P3(15)	P1(20)	P4(20)	P1(20)	P1(12)	P4(20)	P4(20)	P4(20)

0 20 22 42 57 77 97 117 129 149 169
189

RR

P1(20)	P2(20)	P3(20)	P4(20)	P1(20)	P2(2)	P3(15)	P4(20)	P1(12)	P4(20)	P4(20)

0 20 40 60 80 100 102 117 137 149 169
189

IRR

P1(20)	P2(20)	P2(2)	P3(20)	P3(15)	P4(20)	P1(20)	P1(12)	P4(20)	P4(20)	P4(20)

0 20 40 42 62 77 97 117 129 149 169
189

ERR

P1(20)	P2(20)	P2(2)	P3(20)	P4(20)	P1(20)	P3(15)	P4(20)	P1(12)	P4(20)	P4(20)

0 20 40 42 62 82 102 117 137 149 169
189

SARR

P1(28)	P2(22)	P3(28)	P4(28)	P1(24)	P3(7)	P4(28)	P4(24)

0 28 50 78 106 130 137 165 189

COMPARITIVE BARCHART OF WAITING TIME AND TURANAROUND TIME IN CASE 3

VI. CONCLUSION

From the above experiments, IRRSJF algorithm shows better results than RR algorithm, IRR algorithm, ERR algorithm and SARR algorithm in enhancing the CPU performance and its efficiency. By using our algorithm we are getting better Average Waiting Time, Average Turnaround Time and Context Switch. As we have taken the ideal cases in calculating the TAT and WT. In future we can implement this algorithm in real time operating systems.

REFERENCES

[1] Silberschatz, A., P.B. Galyin and G.Gange, 2004, " Operating System Concepts". 7th Edn, John Wiley and Sons, USA, ISBN: 13:978-0471694663, PP: 944.
[2] William Stalling, 2006, "Operating system", 5th Edition Person Education , ISBN :81311703045,
[3] "A.S. Tanenbaum, 2008" Modern Operating Systems, Third Edition, Printice Hall ,ISBN : 13:9780136006633,PP: 1104,
[4] Manish kumar Mishra, "Improved Round Robin CPU Scheduling Algorithm", Journal of Global Research in computer science, ISSN - 2229-371X, vol. 3,No. 6, June 2012
[5] Rami J Matarneh, "Self- Adjustment Time Quantum in Round Robin Algorithms Depending on Burst Time of the Now Running Processes", American Journal of Applied Sciences, ISSN 1546-92396, (10):1831-1837, 2009
[6] SarojHiranwal, " Adaptive Round Robin Scheduling using shortest Burst Approach Based on smart time slice ", International Journal of Data Engineering(IJDE), volume2, Issue 3,2 011
[7] H.S. Behera, "Weighted Mean Priority Based scheduling for Interactive systems", Journal of global Research in computer science, ISSN-2229-371X, volume 2, No.5, May 2011
[8] LalitKishor& Dinesh Goyal, "Time Quantum Based Improved Scheduling Algorithms", International Journal of Advanced Research in Computer science and Software Engineering, ISSN: 2277-128X, Volume 3, Issue 4, April 2013
[9] H.S.Behera, " Enhancing the CPU performance using a modified mean-deviation round robin scheduling algorithm for real time systems", Journal of global Research in computer science, ISSN-2229-371X, volume 3,No. 3, March 2012,
[10] AashnaBisht, " Enhanced Round Robin Algorithm for process scheduling using varying quantum precision", IRAJ International Conference-proceedings of ICRIEST- AICEEMCS,29 th Dec 2013,pune India. ISBN: 978-93-82702-50-4
[11] RakashMohanty&Manas Das, " Design and performance Evaluation of A new proposed fittest Job First Dynamic Round Robin Scheduling Algorithms", International journal of computer information systems, ISSN: 2229-5208, vol. 2, No. 2, Feb 2011.
[12] DebashreeNayak&Sanjeev Kumar Malla, " Improved Round Robin Scheduling using Dynamic time quantum", International Journal of computer Applications, ", International Journal of computer Applications, volume 38, No. 5, January 2012
[13] Lalitkishor,Rajendra Singh, "Optimized Scheduling Algorithms", International journal of computer applications, International conference on computer communication and networks CSI-COMNET-2011

[14] Ishwari Singh Rajput,Deepa Gupta, "A Priority based round robin CPU Scheduling Algorithms for real time systems", International journal of Innovations in Engineering and Technology, ISSN:2319-1058, Vol. 1 ISSUE 3 ,Oct 2012.
[15] P.SurendraVarma, "A Best possible Time quantum for Improving Shortest Remaining Burst Round Robin (SRBRR) algorithms", International Journal of advanced Research in computer science and software Engineering, ISSN: 2277 128X, Vol. 2, ISSUE 11, November 2012.
[16] H.S. Behera&Brajendra Kumar Swain, " A New proposed precedence based Round Robin with dynamic time quantum Scheduling algorithm for soft real time systems", International Journal of advanced Research in computer science and software Engineering, ISSN:2277- 128X, Vol. 2, ISSUE 6, June 2012.
[17] Brajendra Kumar Swain,H.SBehera and Anmol Kumar Parida, "A new proposed round robin with highest response ratio next scheduling algorithm for soft real time system". International Journal of Engineering and Advanced Technology, ISSN: 2249-8958, Vol. - 1,ISSUE-3, Feb 2012
[18] Abbas Noon, Ali Kalakech and SeifedineKadry, " A new Round Robin based scheduling algorithm for operating system: Dynamic quantum using the mean average", International Journal of computer science, ISSN: 1694-0814, Vol. 8,ISSUE 3, No. 1, May 2011
[19] RakeshkumarYadav, Abhishck K Mishra ,NavinPrakash and Himanshu Sharma, " An Improved round robin scheduling algorithm for CPU scheduling", International Journal of computer science and Engineering, ISSN: 0975-3397, Vol. 02, No. 04, 2010 , 1064-1066
[20] Ali JbaeerDawood, "Improved Efficiency of Round Robin Scheduling Using Ascending Quantum and Minumium-Maxumum Burst Time", Journal of university of anbar for pure science, ISSN: 1991-8941, Vol. 6, No. 2, 2012

