

# An Improved and Performance Evaluation of a New Shortest Round First (SRF) based Scheduling Algorithm Considering with Quantum Time

Md. Alomgir Hossain

Lecturer of Computer Science and Engineering  
IUBAT–International University of Business Agriculture  
&  
Technology, Uttara, Dhaka, Bangladesh.

Md. Shariful Islam

Student of Computer Science and Engineering  
IUBAT–International University of Business Agriculture  
&  
Technology, Uttara, Dhaka, Bangladesh

**Abstract:** CPU scheduling is the most primary and basic function of computer's operating system. This is very essential for execution of every program. CPU scheduling is a mechanism by which the CPU of a computer is kept busy. According to CPU scheduling our paper focuses about the shortest round first based scheduling algorithm with quantum time which compare to First come first serve(FCFS), Shortest job first,(SJF), Round Robin(RR). According to our new scheduling algorithm gives us better result than the others scheduling.

**Keywords:** *Operating system, Context switch, Scheduling, turnaround time, waiting time, Average time and burst time.*

## INTRODUCTION:

The operating system is the most important program that runs on a computer to run other programs and applications. For the management of handling multiple applications of a PC. The CPU must have a useful way of using CPU scheduling. Mainly Operating system is to provide as a good environment where user can execute program with efficient manner.

**CPU scheduling:** CPU scheduling operates this multiprogramming task by switching one after another. There must be a way for the operating system and application processes to share the CPU.

**CPU scheduler:** It means amount of the time the CPU is takes for executing instruction of processes. Scheduler can select a process among the processes that are ready to execute and allocates CPU to one of them.

**Dispatcher:** Dispatcher function is essential for CPU scheduling because it controls the process by short term scheduler. The dispatcher should be as fast as possible so that it can switch programs.

**Scheduling criteria:** The criteria include with the following topics.

**CPU utilization:** Generally the CPU should keep busy 100% of time so that CPU cycle does not get wasted. But actually CPU usage range is 40 percent for light load and 90 percent for heavy load.

**Throughput:** Throughput is known for number of process that takes to finish of execution per unit time. May range from 10/second to 1/hour depending on the special task.

1. Turnaround time: The time which is needed for a specific process to finish, from submission time to completion. Turnaround time is the sum of waiting time and burst time.
2. Waiting time: Waiting time is time that tells how much time processes spend in the ready queue waiting their turn in order to get on the CPU.
3. Response time: The time needed in an active program from the issuance of a statement to commence of a response to that statement.
4. Proposed methodology: Operating system provides CPU scheduling to give a scheduling for every process in ready queue. According to the CPU scheduling algorithm its allows minimizing of waiting time, turnaround time and response time and optimum consentient high throughput. There are different type of scheduling algorithms are available in operating system such as First come first served(FCFS), Shortest job first(SJF), priority scheduling, Round robin(RR). The FCFS is the simplest CPU algorithm and it is easy to implement. It does not provide the best service. SJF (Shortest job first) scheduling provides the shortest process priority. RR (Round Robin) is one of the most popular scheduling results in average turnaround time and average waiting time. Our paper gives batter result then others scheduling algorithms.

Proposed SRF CPU scheduling algorithm:

Step 1: Taking processes.

Step 2: Enter the arrival time and burst time of processes in ready queue.

Step 3: If any process arrival time comes then enter the process in the ready queue and do step 4.

Step 4: Present all the processes in the ready queue in ascending order by using their burst time.

Step 5: If the process available in the ready queue then taking first ascending order shortest process from the ready queue and do step 6 otherwise do step 7.

Step 6: Process burst time > quantum time if the condition is true then allocate the CPU to it for a time interval of up to quantum and set => process burst time = remaining burst time and do step 3, 4 and 5. If the condition is false then allocates the CPU to the currently running process and removed it from the ready queue and do step 3, 4, 5.

Step 7: If the ready queue is empty then finish.

Examples:

Process	Arival time	Burst time
P1	0.0	7
P2	2.0	4
P3	4.0	1
P4	5.0	4

Suppose Quantum time = 2

Gantt chart:

P1	P2	P3	P2	P4	P4	P1	P1	P1	
0	2	4	5	7	9	11	13	15	16

Waiting time:

$$P1 = (0-0) + (11-2) + (13-13) + (15-15) = 9$$

$$P2 = (2-2) + (5-4) = 1$$

$$P3 = (4-4) = 0$$

$$P4 = (7-5) + (9-9) = 2$$

$$\text{Average waiting time} = (9+1+0+2)/4 = 3$$

$$\text{Average turnaround time} = (16+5+1+6)/4 = 7$$

Others CPU algorithms:

First Come First Served (FCFS) scheduling: FCFS is very just like a FIFO queue. However, FCFS can take very long average waiting time depending on the first process to get there takes a long time. For example, consider the following 4 processes.

FCFS:

Process	Arival time	Burst time
P1	0.0	7
P2	2.0	4
P3	4.0	1
P4	5.0	4

Gantt chart:

P1	P2	P3	P4	
0	7	11	12	16

Waiting time:

$$P1 = 0-0 = 0$$

$$P2 = 7-2 = 5$$

$$P3 = 11-4 = 7$$

$$P4 = 12-5 = 7$$

$$\text{Average waiting time} = (0+5+7+7)/4 = 4.75$$

$$\text{Average turnaround time} = (7+9+8+11)/4 = 8.75$$

Shortest job first (SJF) Scheduling: The SJF algorithm is essential to pick the fastest small job that needs to be done as quickly as possible and then pick the next smallest fastest job to do next. This algorithm picks a process depending on the next shortest CPU burst.

SJF:

Process	Arival time	Burst time
P1	0.0	7
P2	2.0	4
P3	4.0	1
P4	5.0	4

Gantt chart:

P1	P3	P2	P4	
0	7	8	12	16

Waiting time:

$$P1 = 0-0 = 0$$

$$P2 = 8-2 = 6$$

$$P3 = 7-4 = 3$$

$$P4 = 12-5 = 7$$

$$\text{Average waiting time} = (0+6+3+7)/4 = 4$$

$$\text{Average turnaround time} = (7+10+4+11)/4 = 8$$

Priority Scheduling (PS): Priority Scheduling is a case of SJF, where each task is assigned a priority and the job with the highest priority which will gets scheduled first.

PS:

Process	Arival time	Burst time	Priority
P1	0.0	7	3
P2	2.0	4	2
P3	4.0	1	1
P4	5.0	4	4

Gantt chart:

P1	P3	P2	P4	
0	7	8	12	16

Waiting time:

$$P1 = 0-0 = 0$$

$$P2 = 8-2 = 6$$

$$P3 = 7-4 = 3$$

$$P4 = 12-5 = 7$$

$$\text{Average waiting time} = (0+6+3+7)/4 = 4$$

$$\text{Average turnaround time} = (7+10+4+11)/4 = 8$$

Round Robin (RR) scheduling: Round robin scheduling is a process which is quite similar to FCFS scheduling except that CPU burst are assigned with limits called time quantum.

Process	Arival time	Burst time
P1	0.0	7
P2	2.0	4
P3	4.0	1
P4	5.0	4

Suppose Quantum time = 2

Gantt Chat:

P1	P2	P3	P4	P1	P2	P4	P1	P1	
0	2	4	5	7	9	11	13	15	16

Waiting time:

$$P1 = (0-0) + (7-2) + (13-9) + (15-15) = 9$$

$$P2 = (2-2) + (9-4) = 5$$

$$P3 = (4-4) = 0$$

$$P4 = (5-5) + (11-7) = 4$$

$$\text{Average waiting time} = (9 + 5 + 0 + 4) / 4 = 4.5$$

$$\text{Average turnaround time} = (16 + 9 + 1 + 8) / 4 = 8.5$$

Paper Analysis: If we see the following chart than we can understand the compare of our scheduling with others scheduling algorithm.

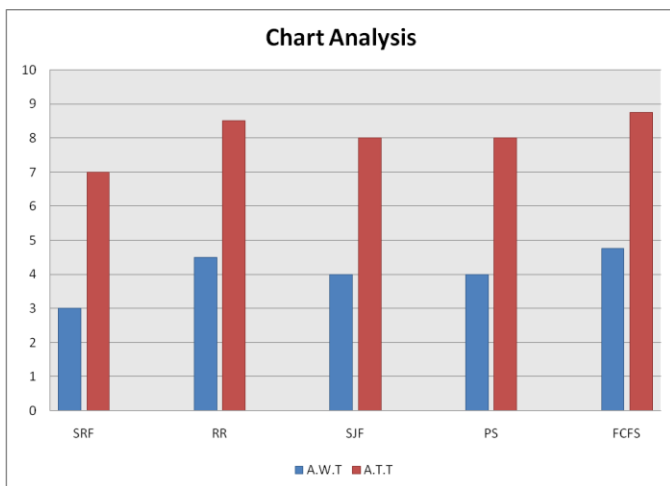


Fig.1

Here, the (Fig.1) shown that average waiting time of SRF is 3 and average turnaround time is 7. The average waiting time of RR is 4.5 and average turnaround time is 8.5. The average waiting time of SJF is 4 and average turnaround time is 8. The average waiting time of PS is 4 and average turnaround time is 8. The average waiting time of FCFS is 4.75 and average turnaround time is 8.75.

So, our scheduling algorithm gives better result than others scheduling algorithms.

### CONCLUSION

In this study we proposed an improved and performance evaluation of a new Shortest Round First (SRF) based scheduling algorithm considering with quantum time for operating system. SRF scheduling algorithm shows that better performance and effectiveness compare with another

scheduling. According to Shortest Round First gives better Average Waiting Time and Average Turnaround Time.

### REFERENCES

- [1] SILBERSCHATZ, ABRAHAM, PETER GALVIN, and GREG GAGNE. *Operating System Concepts*. 9<sup>th</sup> Edit. New Jersey: John Wiley and Sons, INC, 2012. Print
- [2] U. Saleem, and Javed M. Y. *Simulation of CPU Scheduling Algorithms*. Working paper no. 6893967. Vol. 2. Kuala Lumpur: Dept. of Comput. Eng., Nat.Univ. of Scis. & Technol., Rawalpindi, Pakistan, 2000.
- [3] Bashir Alam, "Fuzzy Round Robin CPU Scheduling Algorithm", *Journal of Computer Science*, pp. 1079-1085, 2013.
- [4] 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.
- [5] A. Noon, A. Kalakech, and S. Kadry, "A new Round Robin Based scheduling algorithm for Operating Systems: Dynamic Quantum time Mean Average", *International Journal of Computer Science Issues*. 8(3), 224-229, 2011.
- [6] *Operating Systems\_CPU Scheduling*. [http://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/5\\_CPU\\_Scheduling.html](http://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/5_CPU_Scheduling.html), accessed 8th October 2013.
- [7] 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
- [8] William Stallng, 2006, "Operating system", 5th Edition Person Education, ISBN :81311703045.
- [9] Mehdi Neshat, Mehdi Sargolzaei, Adel Najaran & Ali Adeli, (2012) "The New method of Adaptive CPU Scheduling using Fonseca and Fleming's Genetic Algorithm", *Journal of Theoretical and Applied Information Technology*, Vol. 37, No. 1, pp 1-16.
- [10] H.S. Behera, Rakesh Mohanty, Jainaseni Panda, Dipanwita Thakur & Subasini Sahoo, (2011) "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, pp 54-60.
- [11] H.S. Behera, Simpi Patel & Bijaylaxmi Panda, (2011) "A new dynamic Round-robinand SRTN algorithm using variable original time slice and dynamic intelligent time slice for soft real time system". *International Journal of Computer Applications*, Vol. 2, No.1, pp 54-60.
- [12] H.S. Behera, Jainaseni Panda, Dipanwita Thakur & Subasini Sahoo, (2011) "A New Proposed Two Processor Based CPU Scheduling Algorithm with Varying Time quantum for Real Time Systems", *Journal of Global Research in Computer Science*, Vol. 2, No. 4, pp 81-87.
- [13] Abbas Noon, Ali Kalakech, and Seifedine Kadry, (2011) "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, pp 224-229.
- [14] M.K. Srivastav, Sanjay Pandey, Indresh Gahoi & Neelesh Kumar Namdev, (2012) "Fair Priority Round Robin with Dynamic Time Quantum", *International Journal of Modern Engineering Research*, Vol. 2, Issue 3, pp 876-881.
- [15] Manish Kumar Mishra & Abdul Kadir Khan, (2012) "An Improved Round Robin CPU Scheduling Algorithm", *Journal of Global Research in Computer Science*, Vol. 3, No. 6, pp 64-69.
- [16] Abdulrazak Abdulrahim, Salisu Aliyu, Ahmad M Mustapha & Saleh E. Abdullahi, (2014) "An Additional Improvement in Round Robin (AAIRR) CPU Scheduling Algorithm", *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 4, Issue 2, pp 601-610.
- [17] Abdulrazak Abdulrahim, Saleh E. Abdullahi & Junaidu B. Sahalu, (2014) "A New Improved Round Robin (NIRR) CPU Scheduling Algorithm", *International Journal of Computer Applications*, Vol. 90, No. 4, pp