ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

Linear Data Model Based Study of Improved Round Robin CPU Scheduling Algorithm

Saurabh Jain[1], Diwakar Shukla[2], Ratnesh Jain[3]

Associate Professor, Department of Computer Applications, Shri Vaishnav Institute of Tech. & Sc. Indore (M.P), India[1]

Professor, Department of Computer Science & Applications, Dr. H.S. Gour Univ, Sagar (M.P.), India[2]

TGT, Kendriya Vidyalaya, Arah, Distt. Bhojpur, (Bihar), India[3]

Abstract: One of the main functions of operating system is process management. CPU scheduling plays an important role in getting good performance from the computer system. The main aim of the CPU scheduling algorithms is to minimize waiting time, turnaround time, response time and context switching and maximizing CPU utilization. Round Robin CPU scheduling is a choice for the time sharing systems, where the performance of the system depends on the choice of the optimal time quantum. This paper presents a study of improved Round Robin CPU scheduling algorithm which is a combo of the features of Shortest Job First and Round Robin scheduling with varying time quantum. The data model approach has been adapted in analysis with simulation. It is proven that linear increment of the burst time of processes also creates corresponding linear increment in efficiency parameters. The origin shift of the burst time is also directly proportional to the efficiency parameters.

Keywords: CPU Scheduling, Round Robin (RR) Scheduling, Improved Round Robin Scheduling, Shortest Job First (SJF) Scheduling, Burst Time (BT), Waiting Time (WT), Turnaround Time (TT).

I. INTRODUCTION

In the multiprogramming environment, multiple processes are kept in main memory for maximum CPU utilization. The improper use of CPU can reduce the efficiency of computer system. The CPU can be maximally utilized by running processes all the time and switching the processes between CPU and main memory. One of the focal objectives of CPU scheduling algorithms is to minimize waiting time and turnaround time of processes to improve CPU utilization. In the execution processes, all the processes that are waiting for the processor are kept in a ready queue. Whenever CPU becomes idle, a waiting process from the ready queue is selected by the CPU scheduler and CPU is allocated to that process depending on the scheduling criteria [3]. Some of measures of ideal CPU scheduling algorithm are:

a. **Burst Time (BT):** the maximum time, a process needs CPU for its execution.
b. **Waiting Time (WT):** the total time spent by the process waiting in the ready queue.
c. **Turnaround Time (TT):** the total time taken by a process from the time of submission to the time of completion of the process.

An optimized CPU scheduling algorithms focus on reducing the waiting time and turnaround time and also focuses to minimize the context switching overheads by scheduling the processes from the ready in an effective manner. First-Come-First-Served (FCFS) Round Robin (RR), Shortest Job First (SJF), Multilevel Queue, Multilevel Feedback Queue, and Priority scheduling are some popular CPU scheduling algorithms.

In SJF CPU scheduling algorithm, the process with shortest CPU burst time executes first from the ready queue. In RR scheduling algorithm, each process from the ready queue is selected allotted a fixed time quantum.

In this paper, we have proposed a data model based study of combo algorithm of Round Robin and Shortest Job Scheduling suggested due to [1] that uses the features of SJF and RR and studied the impact of varying time quantum on the waiting time and turnaround time with a linear data model approach with simulation.

II. REVIEW OF WORK DONE

For the Time sharing systems, RR scheduling algorithm is the key algorithm since it requires a sharing of CPU time allotment between different processes being residing in the computer system. A fixed time quantum is allocated to the process waiting in the ready queue, in first execution cycle and then from the next cycle, the Shortest Job First algorithm is used to select next process [2]. The time quantum is continuously adjusted according to the remaining burst time of the processes in each cycle in IRRVQ algorithm [1,4]. The performance analysis of Re-adjusted Round Robin scheduling algorithm is done with Dynamic Quantum [5]. The CPU is allocated to the first process from the ready queue for a time interval of up to one time quantum in an Improved Round Robin (IRR) CPU scheduling algorithm [7]. Same performance evaluation is done by [6] using dynamic quantum. Markov Chain model is used to analyse the performance of Round Robin scheduling scheme [8] and also used with the different classes of Round Robin scheduling scheme [10]. A study on the performance of Deficit Round Robin Alternated algorithm under Markov Chain model has also made some contribution [9].

III. IRRVQ CPU SCHEDULING ALGORITHM (DUE TO [1])

The improved Round Robin CPU scheduling algorithm (Due to [1]) with varying time quantum (IRRVQ) combines the features of SJF and RR scheduling

 DOI10.17148/IJARCCE.2015.46121

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

algorithms with varying time quantum. Initially the processes in the ready queue are arranged in the ascending order of their remaining burst time. CPU is allocated to the processes using RR scheduling with time quantum value equal to the burst time of first process in the ready queue. After each cycle processes in the ready queue are arranged in the ascending order of their remaining burst time and CPU is allocated to the processes using RR scheduling with time quantum value equal to the burst time of first process in the ready queue.

Following is the proposed IRRVQ CPU scheduling algorithm (Due to [1])

1. Make a ready queue RQUEUE of the Processes submitted for execution.
2. DO steps 3 to 9 WHILE queue RQUEUE becomes empty.
3. Arrange the processes in the ready queue RQUEUE in the ascending order of their remaining burst time.
4. Set the time quantum value equal to the burst time of first process in the ready queue RQUEUE.
5. Pick the first process from the ready queue RQUEUE and allocate CPU to this process for a time interval of up to 1 time quantum.
6. Remove the currently running process from the ready queue RQUEUE, since it has finished execution and the remaining burst time is zero.
7. REPEAT steps 8 and 9 UNTIL all processes in the ready queue gets the CPU time interval up to 1 time quantum.
8. Pick the next process from the ready queue RQUEUE, and allocate CPU for a time interval of up to 1 time quantum.
9. IF the currently running process has finished execution and the remaining CPU burst time of the currently running process is zero, remove it from the ready queue ELSE remove the currently running process from the ready queue RQUEUE and put it at the tail of the ready queue.

IV. PROPOSED DATA MODEL

Let Pi be the i^{th} process (i=1,2,3,4,5) and a, α are two model parameters. We suggest the burst time of process in the form of linear data model as bi = a+iα.

where bi : burst time of i^{th} process
a, α: model parameters

V. SIMULATION STUDY

As compared to the study done by Mishra and Rashid [1] in which they have merged the Round Robin scheduling with SJF scheduling, we have used a data model bi = a+iα , where bi will be calculated as the burst time calculated gradually in the increasing order.

The following two iterations used to see the pattern of variation between average waiting time and average turnaround time.

Case I: For a=10, i = 1,2,3,4,5 and α = 1,2,3,4,5,6.

Case II: For a=20, i = 1,2,3,4,5 and α = 1,2,3,4,5,6.

The graphs and tables generated by the above two iterations are as follows:

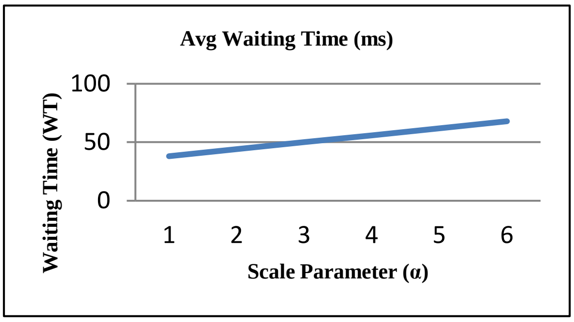

FIG. 4.1(A): AVG. WAITING TIME WHEN A=10

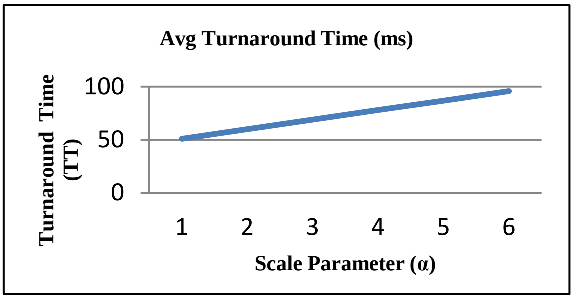

FIG. 4.1(B): AVG. TURNAROUND TIME WHEN A=10

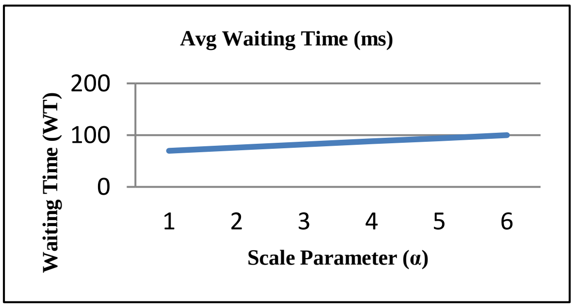

FIG. 4.2(A): AVG. WAITING TIME WHEN A=20

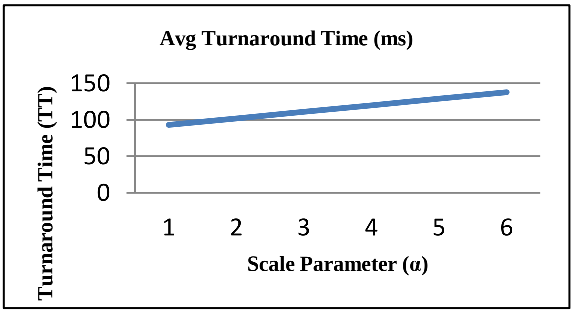

FIG. 4.2(B): AVG. TURNAROUND TIME WHEN A=20

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

i	α = 1			α = 2			α = 3			α = 4			α = 5			α = 6		
	$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$		
	BT	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**
P_1	11	0	11	12	0	12	13	0	13	14	0	11	15	0	15	16	0	16
P_2	12	44	56	14	48	62	16	52	68	18	56	74	20	60	80	22	64	86
P_3	13	47	60	16	54	70	19	61	80	22	68	90	25	75	100	28	82	110
P_4	14	49	63	18	58	76	22	67	89	26	76	102	30	85	115	34	94	120
P_5	15	50	65	20	60	80	25	70	95	30	80	110	35	90	125	40	100	140
Avg. WT & TT		**38**	**51**		**44**	**60**		**50**	**69**		**56**	**78**		**62**	**87**		**68**	**96**

TABLE 4.1: CALCULATION OF WAITING TIME AND TURNAROUND TIME WHEN A=10 (CASE I)

i	α = 1			α = 2			α = 3			α = 4			α = 5			α = 6		
	$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$			$b_i=a+\alpha i$		
	BT	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**	**BT**	**WT**	**TT**
P_1	21	0	21	22	0	22	23	0	23	24	0	24	25	0	25	26	0	26
P_2	22	84	106	24	88	112	26	92	118	28	96	124	30	100	130	32	104	136
P_3	23	87	110	26	94	120	29	101	130	32	108	140	35	115	150	38	122	160
P_4	24	89	114	28	98	126	32	107	139	36	116	152	40	125	165	44	134	178
P_5	25	90	115	30	100	130	35	110	145	40	120	160	45	130	175	50	140	190
Avg. WT & TT		**70**	**93**		**76**	**102**		**82**	**111**		**88**	**120**		**94**	**129**		**100**	**138**

TABLE 4.2: CALCULATION OF WAITING TIME AND TURNAROUND TIME WHEN A=20 (CASE II)

VI. CONCLUSION

The IRRVQ suggested due to [1] depends on linear model parameters. Whenever the equal length interval between burst time values are achieved then corresponding waiting time and turnaround time both are also having equal interval increments. This relationship is linear and increments in burst time will posses gradual increments in α values. We conclude that under data model $b_i=a+i\alpha$, the turnaround time and waiting time both are directly proportionally to the length between the two successive burst times of processes under IRRVQ scheduling scheme. Moreover, it is also observed that processes having higher duration of burst time also follow these proportionality criteria. The origin shift (from a=10 to a=20) provides similar proportionality pattern in linear increments.

REFERENCES

[1] Mishra, M. and Rashid, F. (2014): An Improved Round Robin Scheduling Algorithm with varying quantum, International Journal of Computer Science, Engineering and Applications, Vol.4, No.4, pp. 1-8.

[2] Yadav, R., Mishra, A., Prakash, N. & Sharma, H. (2010): An Improved Round Robin Scheduling Algorithm for CPU Scheduling, International Journal on Computer Science and Engineering, Vol. 2, No. 4, pp. 1064-1066.

[3] Silberschatz, A., Galvin, P. & Gagne, G. (2005): Operating System Concepts, John Wiley and Sons Inc, pp. 157-167.

[4] Matarneh, R. (2009): Self-Adjustment Time Quantum in Round Robin Algorithm Depending on Burst Time of Now Running Processes, American Journal of Applied Sciences, Vol. 6, No. 10, pp.1831- 1837.

[5] Behera, H., Mohanty, R. & Nayak, D. (2010): A New Proposed Dynamic Quantum with Re-Adjusted Round Robin Scheduling Algorithm and Its Performance Analysis, International Journal. of Computer Applications, Vol. 5, No. 5, pp. 10-15.

[6] Nayak, D., Malla, S. and Debadarshini, D. (2012): Improved Round Robin Scheduling using Dynamic Time Quantum, International Journal of Computer Applications, Vol. 38, No. 5, pp. 34-38.

[7] Mishra, M. & Khan, A. (2012): An Improved Round Robin CPU Scheduling Algorithm, Journal of Global Research in Computer Science, Vol. 3, No. 6, pp. 64-69.

[8] Shukla, D., Jain, S. and Singhai, R. (2009): A Markov Chain Model for the analysis of Round Robin scheduling scheme, International Journal of Advanced Networking and Applications, Vol.1, No.1, pp. 1-7.

[9] Shukla, D., Jain, S. and Singh, U. (2008 a): A Markov Chain model for Deficit Round Robin Alternated (DRRA) scheduling algorithm, Proceedings of International Conference on Mathematics and Computer Science, pp. 52-61.

[10] Shukla, D., Jain, S. and Singh, U. (2008 b): A general class of Round Robin scheduling schemes under Markov Chain model, Proceedings of International Conference on Mathematics and Computer Science, pp. 62-70

DOI10.17148/IJARCCE.2015.46121