A Best Possible Time Quantum for Advanced Round Robin With Shortest Job First Scheduling Algorithm

Prof. Dipali V. Patel[1]
[1]Department of Computer Engineering
[1]Vadodara Institute of Engineering, Vadodara, Gujarat, India

Abstract— CPU is the most important resource of computer. To control the sequence of invoking processes of CPU some scheduling algorithms are used. The aim of this algorithms are to reduce waiting time, turnaround time, context switch and CPU utilization. One of them is Round Robin which is mostly used in time shared system because each process is given a fix amount of time to execute processes. but effectiveness of this algorithm depends upon choice of Quantum time. It plays very important role for scheduling, if time is too small then context switch become very high and if it is very big then it will work as FCFS scheduling algorithm. In this paper I have discuss about the selection of quantum time, for that I have introduce a new formula which will help to reduce average waiting time, Turn around time and also context switch. As I have analyzed and tested, it give better performance than Round Robin(RR), IRR(Improved Round Robin), IRRSJF(Improved Round Robin with Shortest Job First),SARR(Self Adjust Round Robin).

Keywords: Scheduling algorithm, quantum time, round robin, SJFS

I. INTRODUCTION

Scheduling the processes is one of the main task of an operating system. Process scheduling is Determining which processes run when there are multiple runnable processes. It is very important Because it can can have a big effect on resource utilization and the overall performance of the system. The main aim of the scheduling is the maximization of CPU utilization, throughput and minimization of response time, waiting time and turnaround time.

A. Some CPU scheduling algorithms

SJF: Associate with each process the length of its next CPU burst. The CPU is assigned to the process with the smallest (next) CPU burst.
RR: Each process gets a small unit of CPU time.
FCFS: Jobs are executed on first come, first serve basis.
Priority Based Scheduling: Each process is assigned a priority. Process with highest priority is to be executed first and so on.

B. Performance metrics

- CPU Utilization - Percentage of time that the CPU is doing useful work (i.e. not idling). 100% is perfect.
- Wait time - Average time a process spends in the run queue.
- Throughput - Number of processes completed / time unit.
- Response Time - Average time elapsed from when process is submitted until useful output is obtained.
- Turnaround Time - Average time elapsed from when process is submitted to when it has completed.

C. Good CPU Scheduling Criteria are

Maximize the CPU throughput and utilization.
Minimize the waiting time, response time and turnaround time

II. RELATED WORK

There are various researches going around the world to improving the productiveness of round robin algorithm. Author has propose a median based approach to find the time quantum, combining the conventional shortest job first and Round Robin algorithms[4]. Introduced approach of best time quantum that is remained after one or more rounds[6]. A dynamic time quantum using median method introduced in SARR algorithm[1]

III. PROPOSED WORK

Here is a formula:

$$\text{Q.T.} = \begin{cases} \text{ciel}\left(Y_{\frac{n}{2}} + Y_f + Y_l\right)/n & \text{if odd} \\ \text{ciel}\left(Y_{\frac{n}{2}} + Y_{\frac{n}{2}+1} + Y_l\right)/n & \text{if even} \end{cases}$$

Where,

Y=number placed in group of number arranged in increasing manner.

n=total number of processes in ready queue.

f=first process' burst time from ready queue before arranging in increasing manner.

l=last process' burst time from ready queue before arranging in increasing manner.

A. Algorithm

1. Enter all processes in ready Queue.
2. Organize all processes in increasing order of their burst time.
3. While(ready queue != empty)
4. Repeat steps 4,5 and 6.
5. Pick first process from arranged process in ready queue and allocate CPU to it for 1 time quantum.
6. If (remaining CPU burst time <1 quantum time) then
7. allocate CPU again to currently running process. if process is finished then delete it from ready queue and go to step 3.
8. If new process enter in ready queue then go to step 2.
9. Delete currently running process and keep it at the end of ready queue.

CASE 1: for Even number of processes

Lets have four processes(P1,P2,P3,P4), burst time for each are (P2=32,P1=22,P3=52,P4=80) with arrival time 0.
calculating quantum time, Here, Number of process is even.
Q.T = (35+52+80)/4
= 41
Arrange all processes in increasing order of burst time.

Table 1

Process number	Arrival time	Burst time
P1	0	22
P2	0	35
P3	0	52
P4	0	80

Gantt Chart:

Table 2

P1	P2	P3	P3	P4	P4

0 22 57 98 109 150 189

Average waiting time = (0+22+57+109)/4
= 47
Turn aroundtime = (22+57+109+189)/4
= 94.25

Table 3

Algorithm	T.Q.	Avg. Waiting Time	Avg. Turnaround Time	Context switch
ARRSJF	41	47	94.25	6
IRRSJF[2]	20	52	99.25	11
RR	20	110.5	129.25	11
SARR[1]	43	57.75	105.5	7

CASE 2: for odd number of processes:
Lets have five processes(P1,P2,P3,P4,P5),
burst time for each are (P1=65,P2=35,P3=15,P4=55,P5=80) with arrival time 0.
Now, calculate quantum time,

T.Q = (15+65+80)/5=32
Now arrange all processes in increasing order of burst time.

Table 4

Process number	Arrival time	Burst time
P3	0	15
P2	0	35
P4	0	55
P1	0	65
P5	0	80

Gantt Chart:

Table 5

P3	P2	P2	P4	P4	P1	P5	P1	P1	P5	P5
0	15	47	50	82	105	137	169	201	234	250

Average waiting time = (0+137+15+50+170)/5
= 74.4
Turn aroundtime = (15+50+105+202+250)/5
= 124.4

Table 6

Algorithm	T.Q.	Avg. Waiting Time	Avg. Turnaround Time	Context switch
ARRSJF	32	74.4	124.4	11
IRRSJF	20	84	134	14
RR	20	124	174	14
SARR	55	79	129	7

IV. CONCLUSION

In this paper i have discussed about scheduling and then various types of scheduling. i generated a formula for time quantum and merge SJF and round robin scheduling algorithm. A comparison of various types of algorithms like, RR algorithm, ERR algorithm, IRR algorithm is also shown in table 3. It gives better performance compare to other algorithms in terms of Average Turnaround Time and Context Switch and Average Waiting Time.

REFERENCES

[1] Rami J. Matarneh" Self-Adjustment Time Quantum in Round robin Algorithm Depending on Burst time of now running process", Department of Management Information System,American Journal of Applied Science 6(10):1831-1837,2009,ISSN 1546-9239.
[2] RadheShyam,Parmod Kumar" Improved Round Robin with Shortest Job First Scheduling", Department of Computer Science,International Journal of advanced Research in Computer Science and software engineering,volume 5(Isuue 3),March 2015, ISSN 2277 128X.
[3] Debashree Nayak, Sanjeev Kumar Malla,DebashreeDebashini,"Improved Round Robin Scheduling using Dynamic Time Quantum", International Journal of Computer Applications volume 38-No. 5,January 2012,ISSN 0975 8887.
[4] LalitKishor, Dinesh Goyal "Time Quantum Based Improved Scheduling Algorithm", Department of Computer Science, International Journal of advanced Research in Computer Science and software engineering,volume 3(Isuue 4),April 2013, ISSN 2277 128X.
[5] aashnabisht, mohdabdulahad, sielviesharma "enhanced round robin algorithm for process Scheduling using varying quantum precision", Department of Computer Science, JamiaHamdard, New Delhi, IRAJ International Conference-Proceedings of ICRIEST-AICEEMCS, Dec 2013, ISBN: 978-93-82702-50-4.
[6] P. Surendra Varma, A Best possible time quantum for Improving Shortest Remaining Burst Round Robin (SRBRR) Algorithm, International Journal of Advanced Research in Computer Science and software Engineering, Volume 2, Issue 11, ISSN: 2277 128X , November 2012.

