Analysis of Adaptive Round Robin Algorithm and Proposed Round Robin Remaining Time Algorithm

Arpita Sharma¹, Mr. Gaurav Kakhani²
¹M. Tech. Scholar, Department of Computer Science & Engineering, Mewar University, India
²Asst. Professor, Department of Computer Science & Engineering, Mewar University, India
¹arpita27.sharma@gmail.com;  gauravkakhani@gmail.com ²

Abstract—The Round Robin Scheduling Algorithm is designed especially for time sharing system. Each process is assigned a time interval, called its time quantum, during which it is allowed to run. Each process is provided a fixed time to execute called time quantum. The Round Robin Scheduling algorithm is a fair scheduling algorithm that gives equal time quantum to all processes. The choice of the time quantum is critical as it affects the algorithm’s performance. This paper is all about the study of Adaptive Round Robin Algorithm and Proposing a new algorithm Round Robin Remaining Time Algorithm which will improve the performance of Adaptive Round Robin Algorithm in terms of Average Waiting Time (AWT) and Average Turnaround Time (ATT).

Keywords—“Operating System, Scheduling Algorithm, Round Robin, Average Waiting Time, Average Turnaround Time”

I. INTRODUCTION

An operating system interacts between the user and the computer hardware. The purpose of an operating system is to provide a platform in which a user can execute programs in well-located and efficient manner. CPU scheduling deals with the problem of deciding which of the processes in the ready queue is to be allocated the CPU Scheduling algorithm are used to allocate the CPU to the processes waiting in the ready queue. As a process executes, it changes state. The state of a process is defined in part by the current activity of that process. The process scheduler selects an available process for program execution on the CPU. As processes enter the system, they are put into a job queue. This queue consists of all processes in the system. The processes that are residing in main memory and are ready and waiting to execute are kept on a list called the ready queue [1].

© 2015, IJCSMC All Rights Reserved
The Standard Round Robin Scheduling Algorithm is designed especially for time sharing system. Each process is assigned a time interval, called its time quantum, during which it is allowed to run. Each process is provided a fixed time to execute called time quantum. A time quantum is generally from 10 to 100 milliseconds. The ready queue is treated as a circular queue. The CPU scheduler goes around the ready queue, allocating the CPU to each process for a time interval of up to 1 time quantum [2].

Adaptive Round Robin Scheduling Algorithm using Shortest Burst Approach Based on Smart Time Slice, It is a Priority Driven Scheduling Algorithm based on burst time of processes. First of all 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 next idea of this approach is to choose the smart time slice (STS) is mainly depends on number of processes [3]. Round Robin Remaining Time Algorithm processes are assigned to the ready queue on the basis of burst time then rearrange all the process in increasing order of their burst time. Time quantum can be calculated by $\sum pi / 2n$. If the remaining CPU burst time of the currently running process is less than the time quantum, the CPU is again allocated to the currently running process for remaining CPU burst time. Otherwise if the remaining CPU burst time of the currently running process is longer than the time quantum, the process will be put at the tail of the ready queue.

II. LITERATURE REVIEW

Soraj and Roy, 2011 [3] proposed a new algorithm that arranges the processes in ascending order of burst time, and then it chooses the smart time slice (STS), which is mainly dependent on the number of processes. “Adaptive Round Robin Scheduling using Shortest Burst Approach Based on Smart Time Slice”. It is a Priority Driven Scheduling algorithm based on burst time of processes. First of all, it arranges the processes according to the execution time/burst time in increasing order that is the smaller the burst time, the higher the priority of the running process. The next idea of this approach is to choose the smart time slice (STS), which is mainly dependent on the number of processes. The smart time slice is equal to the burst time of the mid process, when the number of processes is odd. If the number of processes is even, then the smart time slice (STS) will be the average of the CPU burst of all running processes. Based on the experiments and calculations, the algorithm radically solves the fixed time quantum problem which is considered a challenge for Round Robin Scheduling Algorithm. This algorithm assumes that all processes arrive at the same time in the ready queue.

Ishwari and Deepa, 2012 [4] proposed algorithm which improves all the drawbacks of round robin CPU scheduling algorithm. The paper also presents the comparative analysis of proposed algorithm with existing round robin scheduling algorithm on the basis of varying time quantum, average waiting time, average turnaround time and number of context switches [8].

Saeidi and Hakimeh, 2012 [5] proposed an algorithm which determined the Optimum Time Quantum Value in Round Robin process scheduling method, the operation and performance of the algorithm were analysed over some algorithms in the literature of the work and also over the Standard Round Robin Scheduling Algorithm. It does this by using a new non-linear mathematical model which calculates the optimum value of the time quantum in Round Robin scheduling algorithm, in order to minimize the average waiting time of the processes.

Manish Kumar Mishra, 2013 [6] focused on improved Round Robin Scheduling Algorithm coined enhancing CPU performance using the features of Shortest Job First and Round Robin scheduling with varying time quantum. The proposed algorithm is experimentally proven better than Standard Round Robin Algorithm. The simulation results show that the waiting time and turnaround time have been reduced in the proposed algorithm compared to traditional Round Robin.
III. OBJECTIVE

The aim of this paper work as the following:

- Comparison of Round Robin Remaining Time Algorithm with Standard Round Robin and Adaptive Round Robin in terms of Average Waiting Time and Average Turnaround time.

IV. PROPOSED ALGORITHM

Following is the proposed Round Robin Remaining Time Algorithm:

Step 1: Assign Process to ready queue.
Step 2: Rearrange all the process in increasing order of their burst time
Step 3: While (ready queue! =NULL)
Step 4: Calculate time quantum =∑pi / 2^n
Step 5: If (remaining burst time < time quantum )
    Allocate CPU again to the current running process for remaining burst time
    Else
    Remove the current running process from the ready queue and put it at the tail of the ready queue.
Step 6: If no of process > 0
    Go to step 5
Step 7: End while
Step 8: Calculate average waiting time, average turnaround time.

V. RESULTS

Case I: Input component for the processes in increasing order

With five processes and their increasing burst time (P1 = 14, P2 =34, P3 = 45, P4 = 62, P5= 77) shown in Table 1. Figure 1, Figure 2 and Figure 3 shows Gantt chart for three algorithms Standard Round Robin, Adaptive Round Robin and Round Robin Remaining Time Algorithm respectively.

Table 1

<table>
<thead>
<tr>
<th>Process Name</th>
<th>CPU Burst Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>14</td>
</tr>
<tr>
<td>P2</td>
<td>34</td>
</tr>
<tr>
<td>P3</td>
<td>45</td>
</tr>
<tr>
<td>P4</td>
<td>62</td>
</tr>
<tr>
<td>P5</td>
<td>77</td>
</tr>
</tbody>
</table>

Standard Round Robin Algorithm

Enter processes in ready queue according their given burst time in order that is P1=14, P2=34, P3=45, P4=62 and P5=77 with time quantum= 25.
**Adaptive Round Robin algorithm**

First of all, arrange the processes in ready queue according their given burst time in increasing order that is P1=14, P2=34, P3=45, P4=62 and P5=77 and after that choose the time quantum according Adaptive RR algorithm, the time quantum is the mid process burst time if the given processes are odd, that is 45.

**Round Robin Remaining Time Algorithm**

First of all, arrange the processes in ready queue according their given burst time in increasing order that is P1=14, P2=34, P3=45, P4=62 and P5=77 and after that calculate the time quantum according Round Robin Remaining Time Algorithm. The time quantum is 23..

**Simulation Result of Case I**
Case II: Input component for the processes in decreasing order

With five processes and their decreasing burst time (P1 = 83, P2 = 54, P3 = 30, P4 = 19, P5 = 8) as shown in Table 2. Figure 5, Figure 6 and Figure 7 shows Gantt chart for Standard Round Robin, Adaptive Round Robin and Round Robin Remaining Time Algorithm respectively.

<table>
<thead>
<tr>
<th>Process</th>
<th>CPU Burst Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>83</td>
</tr>
<tr>
<td>P2</td>
<td>54</td>
</tr>
<tr>
<td>P3</td>
<td>30</td>
</tr>
<tr>
<td>P4</td>
<td>19</td>
</tr>
<tr>
<td>P5</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 2

<table>
<thead>
<tr>
<th>Process</th>
<th>CPU Burst Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>83</td>
</tr>
<tr>
<td>P2</td>
<td>54</td>
</tr>
<tr>
<td>P3</td>
<td>30</td>
</tr>
<tr>
<td>P4</td>
<td>19</td>
</tr>
<tr>
<td>P5</td>
<td>8</td>
</tr>
</tbody>
</table>

Standard Round Robin Algorithm

Enter processes in ready queue according to their given burst time in order that is P1=83, P2=54, P3=30, P4=19 and P5=8 with time quantum= 26.

Gantt Chart

Figure 5: Gantt Chart for Standard Round Robin Algorithm (Case II)

Average Waiting Time: 110.4ms
Average Turnaround Time: 149.2ms

Adaptive Round Robin Algorithm

First of all, arrange the processes in ready queue according to their given burst time in increasing order that is P5=8, P4=19, P3=30, P2=54 and P5=83 and after that choose the time quantum according to Adaptive Round Robin Algorithm, the time quantum is the mid process burst time if the given processes are odd, that is 30.

Gantt Chart

Figure 6: Gantt Chart for Adaptive Round Robin Algorithm (Case II)

Average Waiting Time: 46.6ms
Average Turnaround Time: 85.4ms

Round Robin Remaining Time Algorithm

First of all, arrange the processes in ready queue according to their given burst time in increasing order that is P5=8, P4=19, P3=30, P2=54 and P5=83 and after that calculate the time quantum according to Round Robin Remaining Time Algorithm. The time quantum is 19.

Gantt Chart

Figure 7: Gantt Chart for Round Robin Remaining Time Algorithm (Case II)
Average Waiting Time: 44.4ms
Average Turnaround Time: 83.2ms

**Simulation Result of Case II**

The analysis is based on the comparison of Standard Round Robin Algorithm, Adaptive Round Robin Algorithm and Round Robin Remaining Time Algorithm in terms of Average Waiting Time and Average Turnaround Time.

**Case-I**

Table of Comparison of Standard Round Robin, Adaptive Round Robin and Round Robin Remaining Time Algorithm is given below:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time Quantum</th>
<th>Average Waiting Time</th>
<th>Average Turn Around Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Round Robin</td>
<td>25</td>
<td>97</td>
<td>143.4</td>
</tr>
<tr>
<td>Adaptive Round Robin</td>
<td>45</td>
<td>71</td>
<td>117.4</td>
</tr>
<tr>
<td>RRRT Algorithm</td>
<td>23</td>
<td>66.6</td>
<td>113</td>
</tr>
</tbody>
</table>

**Graphical Representation of Case-I**

The figure 9 shows the comparison of average waiting time between Standard Round Robin Algorithm, Adaptive Round Robin Algorithm and Round Robin Remaining Time Algorithm.
The figure 10 shows the comparison of average turnaround time between Standard Round Robin Algorithm, Adaptive Round Robin Algorithm and Round Robin Remaining Time Algorithm.

Figure 10: Comparative Graph for Average Turnaround Time (Case I)

**Case-II**

Table of Comparison of Standard Round Robin, Adaptive Round Robin and Round Robin Remaining Time Algorithm is given below:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time Quantum</th>
<th>Average Waiting Time</th>
<th>Average Turn Around Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Round Robin</td>
<td>26</td>
<td>110.4</td>
<td>149.2</td>
</tr>
<tr>
<td>Adaptive Round Robin</td>
<td>30</td>
<td>46.6</td>
<td>85.4</td>
</tr>
<tr>
<td>RRRT Algorithm</td>
<td>19</td>
<td>44.4</td>
<td>83.2</td>
</tr>
</tbody>
</table>
Graphical Representation of Case-II

The figure 11 shows the comparison of average waiting time between Standard Round Robin Algorithm, Adaptive Round Robin Algorithm and Round Robin Remaining Time Algorithm.

![Average Waiting Time Graph](image1)

The figure 12 shows the comparison of average turnaround time between Standard Round Robin Algorithm, Adaptive Round Robin Algorithm and Round Robin Remaining Time Algorithm.

![Average Turnaround Time Graph](image2)

After studying all graphs regarding case I and case II it can be said that out of these three algorithms Round Robin Remaining Time Algorithm is best in terms of average waiting time and average turnaround.

CONCLUSION

This comparative study of three CPU Scheduling Algorithm Standard Round Robin Algorithm, Adaptive Round Robin Algorithm and Round Robin Remaining Time Algorithm, shows that the Round Robin Remaining Time Algorithm reduces the Average Waiting Time (AWT) and Average Turnaround Time (ATT) in both cases of increasing order and decreasing order of burst time of processes, So it can be said that Round Robin Remaining Time Algorithm improves performs better than Standard Round Robin Algorithm, Adaptive Round Robin Algorithm in terms of Average Waiting Time (AWT) and Average Turnaround time (ATT).
REFERENCES


