**PERFORMANCE ANALYSIS AND MODIFICATION OF IMPROVED MEAN ROUND ROBIN SHORTEST JOB FIRST SCHEDULING ALGORITHM**

**Enteria, Jervin Rae G.**

Computer Science Student

University of the Cordilleras

jervin.galano01@gmail.com

**Mitabtab, Sharrie Ann Marie G.**

Computer Science Student

University of the Cordilleras

yukimuraryuuji23@gmail.com

**Orbino, Michael L.**

Computer Science Student

University of the Cordilleras

fayegolles@gmail.com

**Orsolino, Abigail A.**

Computer Science Student

University of the Cordilleras  
orsolino0508@gmail.com

**Abstract** – [1]In recent times, some researchers are still comparing their proposed algorithms to the famous algorithms such as FCFS, SJF RR when the research domain has already been modified.

CPU scheduling is one of the most primary and essential parts of any operating system. It prioritizes processes to efficiently execute user requests and help in choosing the appropriate process for execution. It is an integral part of any operating system and defines the basic functionality of an operating system. An existing algorithm Improved Mean Round Robin Scheduling Algorithm is used in this paper for the analysis, in order for the scheduling algorithm to be utilized fully. The algorithm is responsible for analyzing the processes, choosing and dispatching the most appropriate process for execution. Two of the most commonly used scheduling algorithms are the round-robin (RR) algorithm and the Priority Scheduling algorithm. However, they both have their own pros and cons with respect to the quality of service provided by a good scheduling algorithm. Thus our paper seeks to add a certain feature of Adaptive Round Robin with the algorithm IMRRSJF to further improve its functionality and also a performance analysis is made on the basis of data gathered with the help of Mathematical Analysis using Non-recursive Algorithms.

**Introduction**

[2]In a single-processor system, only the process can run at a time; any others must wait until the CPU is free and can be rescheduled. The objective of multiprogramming is to have some process running at all times, to maximize CPU utilization.

[3]Scheduling is a fundamental operating-system function. Almost all computer resources are scheduled before use. The CPU is, of course, one of the primary computer resources. Thus, its scheduling is central to the operating-system design. CPU scheduling determines which process runs when there are multiple run-able processes. CPU scheduling is also important because it can have a big effect on resource utilization and the overall performance of the system.

[4] CPU scheduling is similar to other types of scheduling, which have been studied over the years. CPU scheduling refers to the decision of allocating a single   
resource among multiple clients. It also tracks the order of allocation and duration. The primary objective of scheduling is to optimize system performance. The optimization system is considered the most demanding criteria by the system designers.

[5]This concept was implemented via algorithm called scheduling algorithm, thus the following examples: First Come First Serve (FCFS), Shortest Job First (SJF), and Round Robin (RR). These algorithms were designed based on suitability of implementations compatible with the behavior of the system that achieved its objectives.

It is in this light that in addition, listed algorithms with their suitable respective areas of applications as: FCFS and SJF are suitable for batch system and non-interactive systems. Each of the algorithms is expected to exhibit a good quality of maintaining a minimum turnaround time, waiting time, response time, context switch and maximum throughput and CPU utilization irrespective of its system of the application.

[6] Round Robin being the most popular in time-shared an operating system, but it may not be suitable for real-time operating systems because of the high turnaround time, waiting time and a large number of context switches.

Improved Round Robin Shortest Job First (IMRRSJF) is a proposed algorithm by Shyam and Nandal in the year 2014, that combines Round Robin with Shortest Job First Scheduling. The Time Quantum studied to improve the efficiency of Round Robin and performs; degrades with respect to Context Switching, Average Waiting Time and Average Turnaround Time. This approach was proposed to calculate the TQ, known as the square root of mean and highest burst time multiplied to the number of processes. The processes were ascending with the shortest remaining time and then time quantum is given to the ascending process to CPU.

**\*Note:**

The process is the state when a program is executed. When the computer is running, there are many processes running simultaneously. A process may create a derivative process that is carried out by a larger process. The derivation process is also able to create a new process so that all these processes will be able to form a process tree. When a process is made then the process can obtain these resources such as CPU time I/O devices etc. This is to make the computer work more productive. A process generally consists of two cycles of I/O and CPU Burst performed alternately until the processes are complete.

**RELATED WORKS**

An Operating System gives a significant job in overseeing processes in the form of multiple queues. The arrival of a process is random alongside their various classifications and types. All these require scheduling algorithms to work over the real time environment in reference to task, control and efficiency. Different Researchers have presented different CPU scheduling algorithm. A few of those researches are:

[8] Andysah et al. (2016) introduced the Comparison Analysis of CPU Scheduling: FCFS, SJF and Round Robin, compared three algorithms, FCFS, SJF, and Round Robin. Which target is to know what algorithm is more suitable for a certain process.

[9] Neetu Goel et al. (2012) presented a diagram that depicts the comparative study of various scheduling algorithms for a single CPU and shows which algorithm is best for the particular situation. Using this representation, it becomes much easier to understand what is going on inside the system and why a different set of processes is a candidate for the allocation of the CPU at different times. The objective of the study is to analyze the high efficient CPU scheduler on design of the high quality scheduling algorithms which suits the scheduling goals.

[10] Verma, et al (2017) compared the performance of several job scheduling algorithms in terms of their parameters such as Execution Time, Response Time, and Cost, Scalability, Trust, Reliability, Resource utilization, Energy consumption and load balancing. The aim of the study is to show the comparison of the parameters used when it comes to their proficiency and efficiency in scheduling.

[11] Ahamad, Mohd (2017) proposed two approaches to modify the existing Round Robin Algorithm. In the first approach, we have combined the Round Robin Algorithm with Shortest Job First Algorithm and assigned the time quantum as the lowest burst time of first process in sorted (ascending) ready queue. By doing this we are able to reduce Average Turnaround Time but our aim is to reduce the Average Waiting Time also. Therefore, we propose a second approach, in this we have implemented the combination of Round Robin Algorithm with Shortest Job First Algorithm and assign the time quantum with the average burst time of all the processes of the ready queue.

[12] Shweta Jain, et al. (2016) evaluates the different scheduling algorithms that performed with different parameters such turnaround time, burst time, response time,

waiting time, throughput, fairness and CPU utilization. The review proposed different techniques which is helpful in designing the Operating System and new researches in Scheduling.

**Proposed Algorithm:**

The proposed algorithm is to improve the functionality of Improved Mean Round Robin Shortest Job First (IMRRSJF) is to incorporate a feature from Adaptive Round Robin in lessening the context switches.

[7] The algorithm that arranges the processes in ascending order of burst time, and then it chooses the smart time slices (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 the burst time of processes.

**OBJECTIVES:**

* Analysis of Improved Mean Round Robin Shortest Job First Scheduling Algorithm and modify the Improved Mean Round Robin Shortest Job First with Adaptive Round Robin features
* Comparison of the proposed algorithm with standard Round Robin and Proposed algorithm in terms of Average Waiting Time, Average Turnaround time and Context Switches

**Methodology**

The method we are going to be using is the Mathematical Analysis of Non-recursive Algorithm in order to determine the performance of Improved Mean Round Robin Shortest Job First.

Also the Mathematical Analysis of Non-recursive Algorithm has five steps:

Step 1: Identify the parameter/s indicating the input’s size

Step 2: Determine the basic operation of the algorithm

Step 3: Determine worst, average, and best cases for input of size n

Step 4: Get the sum of the number of times the algorithm’s basic operation is executed

Step 5: Establish the order of growth of the algorithm

**Pseudocode**

1. All the processes present in the ready queue are sorted in ascending order of their burst time.
2. While (Ready queue != NULL)

TQ = √((1/n) (A1+ A2 + …. + An)\*An) //TQ = Time Quantum = Square root of (mean \* Highest Burst) // A = Set of processes, where (A1,A2.....An) ϵ A and n = Total no. of processes //if TQ> maximum burst time, then max( BT) → TQ

1. Assign TQ to (1 to n) process for j = 1 to n { Pj → TQ , } //Pj = Process j
2. Calculate the remaining burst time of the processes.
3. if (new process is arrived and BT != 0 ) //BT = burst time go to step 1, else if ( new process is not arrived and BT !=0) go to step 2, else go to step 6, end if, end while
4. Calculate ATT, AWT and CS. //ATT = Average Turnaround Time , AWT = Average Waiting Time , CS = No. of Context Switches
5. End

Referring to the pseudocode, the while loop will be executed as long as the ready queue is not NULL // n + 1. This tells us that the worst case of the algorithm will be (n+1) \* n. While the best-case scenario would be when there are no processes to run, it means that the program will end eventually. Also as shown in the pseudocode that the computation for the Time Quantum will only be executed as long as all the processes are all done sorted.

**Comparison**

The table below shows the comparison of performance between the Improved Mean Round Robin Shortest Job First and the proposed algorithm.

Comparison in aspects such as:

Time Quantum - TQ

Average Waiting Time - AWT

Average Turnaround Time - ATAT

Context Switches - CS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | TQ | AWT | ATAT | CS |
| **IMRRSJF** | 63 | 78.2 | 128 | 6 |
| **Modified IMRRSJF** | 63 | 65.6 | 115.4 | 4 |

|  |  |  |
| --- | --- | --- |
|  | **IMRRSJF** | **Proposed** |
| Time Complexity | Best Case:  Worst Case: | Best Case:  Worst Case: |

**Discussion**

As seen in the comparison of the IMRRSJF and the modified algorithm, the difference between both algorithms can be seen. With the analysis of their performance, we can see that the proposed algorithm excels in different aspects although it only shows one test case we can

**Conclusion**

**Future works**

**References**

[1] Musa, K. I., Lasisi, K. E., & Gokir , J. A. (n.d.). COMPARATIVE PERFORMANCE AND ANALYSIS OF SOME IMPROVED ROUND ROBIN CPU SCHEDULING, *8*(8), 1494–1502. Retrieved from http://www.ijser.org/

[2] Silberschatz, A. P.B Galvin and G. Gagne (2012), Operating System Concepts, 8th Edition, Wiley India

[3] Sabrian, F., C.D. Nguyen, S. Jha, D.Platt and F. Safaei, (2005). Processing resources scheduling in programmable networks. Computer communication, 28:676-687

[4] A.S. Tanenbaun, Modern Operating Systems.3rd Edn, Prentice Hall, ISBN:13: 9780136006633, pp: 1104, 2008.

[5] A. Bhardwaj and R. Singh, “Comparative Study of Scheduling Algorithm in Operating Systems”, International Journal of Computers and Distributed Systems, vol. 03, no. 01, April-May 2013, ISSN 2278-5183.

[6] Shyam , R., & Nandal, S. K. (2014). Improved Mean Round Robin with Shortest Job First Scheduling, *4*(July), 170–179. Retrieved from <http://www.ijarcsse.com>

[7] Saroj Hiranwal, Dr. K.C. Roy “Adaptive Round Robin Scheduling using Shortest Burst Approach Based on Smart Time Slice”, International Journal of Computer Science and Communication Vol. 2, No.2, July-December 2011, pp 319-323.

[8] Siahaan, Andysah Putera Utama. (2016). Comparison Analysis of CPU Scheduling : FCFS, SJF and Round Robin. International Journal of Engineering Development and Research. 4. 124-132.

[9] Goel, Neetu & Garg, R.. (2013). A Comparative Study of CPU Scheduling Algorithms.

[10] Monika Verma, Er. Krishan Kumar and Dr. Himanshu Monga (2017) Comparative analysis of Job Scheduling algorithms, A Review. Vol. 5, Issue 3 ISSN: 2321 - 9939

[11] Ahamad, Mohd. (2017). Performance Analysis of Modified Round Robin CPU Scheduling Algorithm. International Journal of Advanced Computer Research. 8.

[12] Shweta Jain, Dr. Saurabh Jain (2016) A Review Study on the CPU Scheduling Algorithms. Vol 5. Issue 8.