Volume 3, Issue 5, May. 2024, No. 48, pp. 525-541
Submitted 2/2/2024; Final peer review 21/3/2024
Online Publication 4/5/2024
Available Online at http://www.ijortacs.com

AN ENHANCED DYNAMIC TIME QUANTUM MEAN ROUND ROBIN ALGORITHM FOR EFFICIENT RESOURCE ALLOCATION IN CLOUD COMPUTING

Bamaiyi Sule*[1], Dr. Etemi Joshua Garba[2] , Dr. Yusuf Musa Malgwi[3]
[1,2,3]Department of Computer Sciences, Modibbo Adama University Yola, Adamawa State, Nigeria.
[1*]Bams4ann@gmail.com, [2]e.j.garba@mau.edu.ng, [3]yumalgwi@mau.edu.ng

Abstract

This study presents an impartial scheduling algorithm that allocates the same time quantum to all processes. The selection of time quantum and preemptive strategies significantly impacts the algorithm's performance. This research introduces an enhanced version of the Revamped Mean Round Robin (RMRR) CPU scheduling algorithm, titled 'An Enhanced Dynamic Time Quantum Mean Round Robin Algorithm for Efficient Resource Allocation in Cloud Computing.' Through empirical evaluation, the new algorithm demonstrated improved performance metrics compared to existing scheduling approaches. Benchmarking against other algorithms in the literature, other new algorithm yields minimal average waiting time (AWT), average turnaround time (ATAT), and number of context switches (NCS). These findings underscore the efficacy of the newly developed algorithm, suggesting its preference for systems employing Round Robin (RR) CPU scheduling.

Keywords

Round Robin, Revamped mean Round Robin (RMRR), Scheduling Algorithm, Cloud Computing,Waitin Time, Turnaround Time, Response Time, average waiting time (AWT), average turnaround time (ATAT), number of context switches (NCS).

1. Introduction

In recent time, the concept of resource allocation is paramount to address in the cloud computing environment. Cloud computing is an emerging and trending technology (Pradhan, Prafulla. Behera, Ray, 2016). It aims to provide reliable, customized and Quality of Service (QoS) guaranteed computing dynamic environments for end-users. It is platform independent, hence users have no need to install any piece of software on their local PC and the resources provided by the cloud developers are used on a rented basis in a pay as you go manner (Gokilavani, Selvi, and Udhayakumar, 2013; Bhavani and Guruprasad, 2014;

Currently Cloud Computing is an emerging computing technology which is the big step in development and deployment of an increasing number of distributed applications (Pradhan, et al., 2016). Cloud Computing is defined as the computing model that operates based on Clouds. In turn, the Cloud is defined as a conceptual layer11 that operates above an infrastructure to provide services in a timely manner (Prakash et al., 2014). The emergence of cloud Computing is aim at providing reliable, customized and Quality of Service (QoS) guaranteed computing dynamic environments for end users (Pradhan et al., 2016). Distributed processing, parallel processing and grid computing together emerged as cloud computing. The basic principle of cloud computing is that user data is not stored locally but is stored in the data center of internet (Gokilavani et al., 2013). According to the National Institute of Standards and Technology (NIST) definition, Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction (Mell and Grance, 2011; Biwas, Samsuddoha, Asif, and Ahmed, 2023).

There are numerous advantages of cloud computing the most basic ones being lower costs, re-provisioning of resources and remote accessibility. Cloud computing lowers cost by avoiding the capital expenditure by the company in renting the physical infrastructure from a third party provider (Pradhan, et al., 2016; Biwas et al., 2023). Due to the flexible nature of cloud computing, we can quickly access more resources from cloud providers when we need to expand our business. The remote accessibility enables us to access the cloud services from anywhere at any time. To gain the maximum degree of the above mentioned benefits, the services offered in terms of resources should be allocated optimally to the applications running in the cloud. Cloud computing, at its simplest, is a collection of computing software and services available from a decentralized network of servers (Yaashuwanth, and Ramesh, 2015; Igbal, Ullah, Khan, Aslam, Shaheer, Humayon, Salahudin, and Adeel, 2023).

2. CPU SCHEDULING ALGORITHM

The fundamental CPU scheduling algorithms are First Come First Serve (FCFS), Shortest Job First (SJF), Round Robin (RR), Revamped Mean Round Robin (RMRR), New Improved Robin Robin (NIRR), Improved Round robin with Varying Time Quantum (IRRVQ) and Improved Dynamic Round Robin (IDRR). The FCFS scheduling is the simplest CPU scheduling algorithm, which assigns CPU to the processes on the basis of their arrival time to the ready queue. The Arrived process is inserted into the rear of the ready queue and the process after execution is removed from the front of the ready queue. A longer CPU-bound process will dominate the CPU time and may force shorter CPU-bound processes to wait for long periods. In SJF, the scheduler sort processes according to minimum burst time in the ready queue, so that the process having least burst time is scheduled first. If two processes have same burst times, then FCFS is followed. Processes which require large time may wait for prolonged periods, because the CPU will execute short processes first. It has been proven as the fastest scheduling algorithm. Round Robin (RR) is particularly designed for time-sharing systems; each process have a small unit of CPU time (time quantum). This algorithm allows a process in the ready queue to run until its time quantum finishes, and then executes the next process in the ready queue. While in the Revamped Mean Round Robin, New Improved Round Robin, Improved Round Robin with Varying Time Quantum and Improved Dynamic Round Robin, queues are created to hold incoming processes, and the time quantum are determine both statically and dynamically.

2.1 SCHEDULING CRITERIA

Different CPU scheduling algorithms have its own properties as mentioned above. The choice of a specific algorithm may favour one class of processes over another. For choice of an algorithm for a particular situation, the characteristics of various algorithms must be taken in consider (Kathuria, Singh, Tiwari, & Prashant, 2016). Many criteria have been proposed for comparing CPU scheduling algorithms. These characteristics are used to compare and to make a significant difference in which algorithm is judged to be the best one.

The scheduling criteria include the following:

i. Context switch: A context switch is a procedure that a computer's CPU (central processing unit) follows to change from one task (or process) to another while ensuring that the tasks do not conflict.
ii. CPU utilization: The percentage of time that the CPU is busy.

iii. Throughput: The number of processes completing in a unit of time.
iv. Turnaround time: is the time interval from the time of submission of a process to the time of the completion of the process.
v. Waiting time: The total amount of time that a process is in the ready queue.
vi. Response time: The time between when a process is ready to run and its next I/O request.

A good scheduling algorithm should possess the following characteristics as stated by (Khatri, 2016; Omotehinwa*et al.*, 2019).

i. Minimum context switches.
ii. Maximum CPU utilization.
iii. Maximum throughput.
iv. Minimum turnaround time.
v. Minimum waiting time.
vi. Minimum response time

The performance of Round Robin CPU scheduling is depends on time quantum selection, because if time quantum is large then RR will works the same as the FCFS. If the time quantum is very small then context switches will be very high. Different values of time quantum will lead to different performances and will affect the algorithm's efficiency by affecting the turnaround time, waiting time, response time and number of context switches.

The Developed algorithm calculates time quantum dynamically, by taking the time quantum in the pre-ready queue to be equal to min burst time of process available in the pre-ready queue multiply by two that is **TQ= min burst time * 2**. While for ready queue time quantum is dynamically calculated as root-mean-square of the available burst time of processes Multiply by 1.7, **that is TQ = root-mean-square *1.7**.

This algorithm together with FCFS, SJF, RR are implemented and their results are compared based on average turnaround time, average waiting time, average response time and number of context switches. Results of the analysis shows that the developed algorithm is very promising as it out performed other algorithms in term of average turnaround time, average waiting time, average response time and number of context switches.

3. REVIEW OF RELATED LITERATURE

This section gives a review of related work done by some researchers on the variant scheduling Algorithm, considering both statically determine time quantum and dynamically determine time quantum.

The author (Abdulrahim, Abdullahi, and Sahalu, 2014), proposed to improved the work of (Manish and Abdulkadir, 2012), titled New Improved Round Robin CPU (NIRR) scheduling algorithm. The proposed algorithm was implemented and benchmarked against five other algorithms. From their experiment, the proposed algorithm produces minimal average waiting time, turnaround time and number of context switches more than the other algorithms in their literature, but has a major drawback on response time.

Yaashuwanth and Ramesh, (2015) and Abubakar*et al.*, (2023), propose a Modified Round Robin Algorithm (MRR) where Time Slice (TS) is calculated and allocates based on (range× total no of process (N)) divided by (priority (pr) × total no of process (p)). Range is calculated by (maximum burst time + minimum burst time) divided by the scheduling process. From their results MRR outperform the Simple RR for all parameters. (Samal*et al*., 2013) also proposed another algorithm called Time Slice Priority Based Round Robin (TSPBRR)**.** TSPBRR focuses on priority and is compared to other variants. From their experimental result, TSPBRR focuses on priority and as compared to other variants. It is also observed that the ATAT and AWT in TSPBRR are superior to that in MRR algorithm. Let 'TQ_i' is the time quantum in round i. The number of rounds i varies from 1 to n, where value of i increments by 1 after every round till ready queue is not equal to NULL.

The author (pradhan *et al*. 2016), proposes an algorithm that modifies the RR algorithm with a dynamic time quantum. The algorithm calculates time quantum by maintain two different queues namely PRQ and RQ. The time quantum is obtained by taking the mean of the burst time of processes in the RQ. When it allocates CPU to first process present in RQ if the remaining CPU burst time of the currently running process is less than time quantum then the CPU will still allocate the currently running process for remaining CPU burst time and after the process complete its execution, it will be removed from the ready queue and allocate CPU to next process. This mechanism helps in reducing the number of contexts switching and waiting time, but has a major drawback on response time and turnaround time.

In this work (Gupta, Priya, Carlos, garg, and Dinesh, 2022). Gupta Elaborate an optimized his earlier work of Priority Round Robin for better CPU Scheduling. The time-specific applications are assigned to Central Processing Unit (CPU) of the system and one of the most promising functions of the time-sharing operating systems is to schedule the process in such a way that it gets executed in minimal time. At present, the Round Robin Scheduling Algorithm (RRSA) is the most widely used technique in a timesharing operating system because it gives better performance than other scheduling techniques, namely, First Come First Serve (FCFS), Shortest Job First (SJF), and Priority scheduling. The major challenge in RRSA is the static value of Time Quantum (TQ) which have plays a pivotal to decrease or increase the performance of the system. In existing literature, many statistical techniques are used for identifying efficient time quantum for RRSA. However, there is limited exposure in existing literature on generating a learning model for identifying optimized TQ. In this research work, a new research direction is given for identifying Optimized TQ by training a learning model and predicting optimum TQ value.

The author (Khatri, 2016), proposes an Improved Dynamic Round Robin model in which the median of the set of processes in the ready queue is considered as the optimal time quantum and if the median is less than 25 then its value must be considered as 25 to avoid the overhead of context switch. The first process in the ready queue is allocated to the CPU for a time interval of up to 1 time quantum. If the remaining burst time of the currently running process is less than or equal to 1 time quantum, the processor again allocated to the same process. Else the process is preempted and place at the rear of the ready queue otherwise the process will be halted. Experimental result shows that the proposed algorithm IDRR, when compared with various variants of Round Robin algorithm it produces minimal average waiting time and average turnaround time but has a major drawback on response time.

The Authors (Nayak, *et al.*, 2012), propose an algorithm which employs the median method to find out optimal time quantum. All processes in ready queue are arranged in ascending order based on their burst time. Median and the highest burst time is calculated and set as the time quantum. Their experimentally results performs better than the RR algorithm, by reducing context switching, average waiting and average turnaround time

The Researchers (Manish and Faizur, *2014)*, propose an improve Round Robin with varying time quantum (IRRVQ) employ the features of Shortest Job First and Round Robin scheduling with varying time quantum to improve on Round Robin CPU scheduling. IRRVQ arrange all processes found in the ready queue in ascending order according to their burst time and set time quantum equal to burst time of the first process in the ready queue. Their simulation results show that the waiting time and turnaround time have been reduced in the proposed algorithm compared to traditional RR.

The author (Pradhan, 2016) proposed a Modified Round Robin Algorithm for Resource Allocation in Cloud Computing. In their work they modify the traditional round robin that has the drawback of static time quantum and high waiting and turnaround time. They address this issue by employing the use of dynamic time quantum to minimize average waiting and turnaround time. Their experiment was performed using matlab. Although their result drastically reduced the high average waiting and turnaround time found in the traditional round robin. The problem with their work is that, jobs with smaller remaining burst time are always kept in the queue which lead to higher response time and increase in context switches.

The work of (Dash *et al.*, 2015*)*, presented a Dynamic Average Burst Round Robin (DABRR) uses dynamic time quantum. The processes are arranged in an ascending order in the ready queue base on their burst time. The average of the active processes in the ready queue are calculated and set as the time quantum after each iteration. Their results improved performance in terms of average waiting time and average turnaround time.

4. THE DEVELOPED ALGORITHM

The Developed algorithm worked the following steps:

Step 1: Start

Step 2: Create two queues PREREADYQUEUE (PRQ) and READYQUEUE (RQ)

Step 3: new processes will put into PREREADYQUEUE

Step 4: while (PRQ is not empty) Perform Round Robin with time quantum equals to n unit time.

Step 5: Move all processed process to RQ from PRQ,**i**n READYQUEUE, apply following steps.

Step 6: TQ = root mean time square of burst time of processes present *1.65 in RQ.

Step 7: Allocate CPU to first process that arrive in READYQUEUE.

Step 8: If the remaining CPU burst time of the currently running process is less than or equal to the time quantum then allocate the CPU again to the currently running process for remaining CPU burst time.

Step 9: After completion of execution, remove the process from the ready queue and allocate CPU to next process in READYQUEUE.

Step 10: If a new process arrives in the system during this time with Burst time less than Current Time Quantum, put it into READYQUEUE.

Step 11: Else put it into PREREADYQUEUE.

Step 12: Wait till current process get executed completely in READYQUEUE. Now go to step 4.

Step 13: WHILE READYQUEUE is not empty.

Step 14: Calculate AWT, ATAT,ART, NCS.

Step 15: END

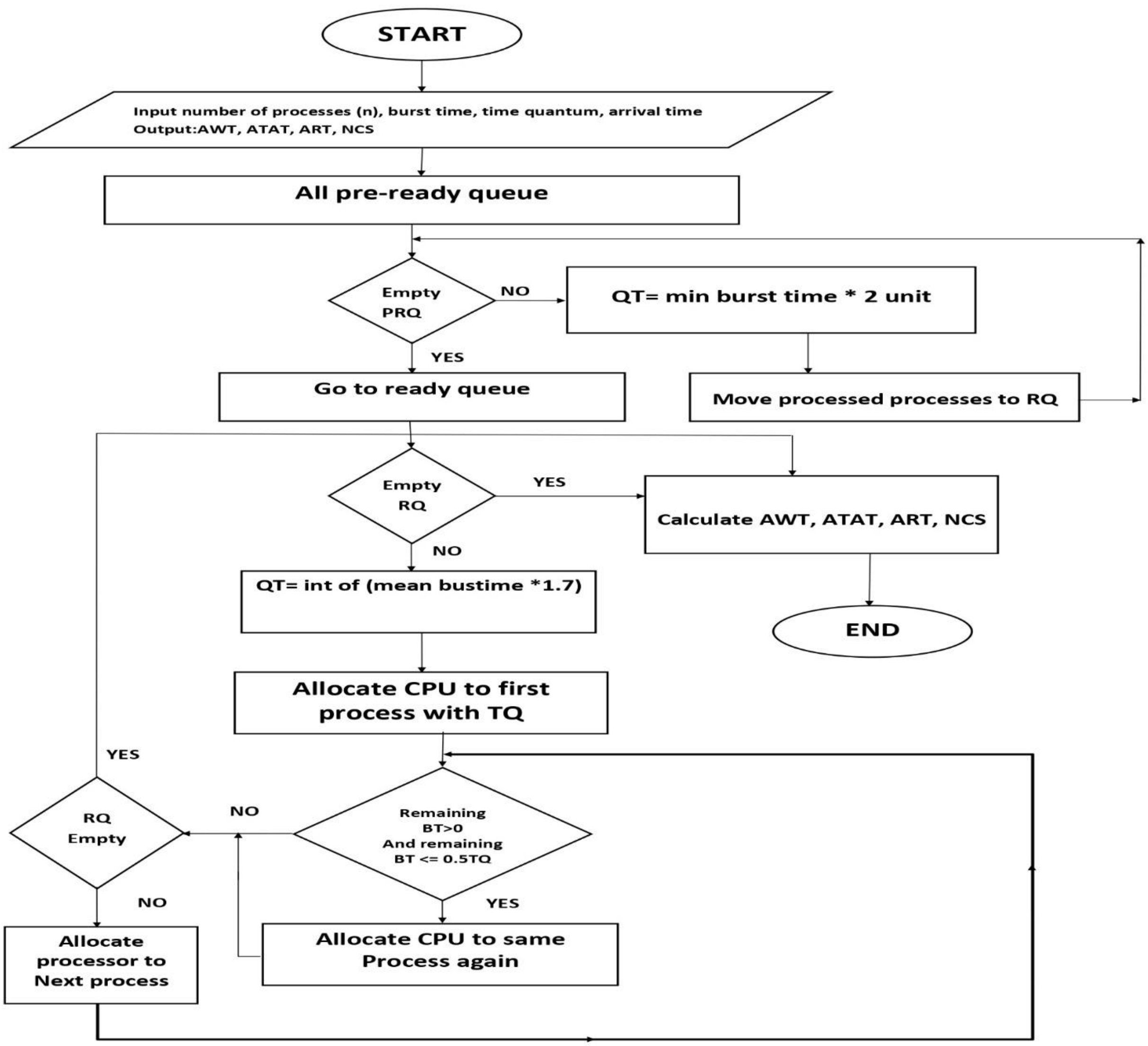

Figure 1: Flowchart of the Developed Algorithm

5. Experimental Data

We consider the experimental data generated using a random scheduler, with queue of five process P0, P1, P2, P3, P4, arriving at time 0, 1, 17, 2, 4 with burst time 99, 94, 48, 23, 27 respectively, is used in carrying out the findings thus, the system considers five different processes with randomly generated burst time for each processes and their corresponding arrival time randomly generated by the random scheduler. As displayed below in table 1 with their corresponding burst time.

Table 1: Random Generated Processes, Arrival Time and Bust Time

Process	Arrival Time	Bust Time
P_0	0	99
P_1	1	94
P_2	17	48
P_3	2	23
P_4	4	27

5.2 The Various Algorithms Considered

The following algorithm will be run and analyzed and the result will be compared with each other and the performance will be evaluated against the standard evaluation metrics of average waiting time, Average Turn-Around Time and Number of context switches, in other to determine the best performing algorithm, the algorithm are:

a. The Revamped Mean Round Robin (RMRR)
b. New Improved Round Robin (NIRR)
c. Improve Round Robin with Varying Quantum-Time (IRRVQ)
d. Improve Dynamic Round Robin (IDRR)
e. The Developed Algorithm

Table 2: Comparative Result Obtain from Various Algorithm Considered

Performance metrics	RMRR	NIRR	IRRVQ	IDRR	Developed Algorithm
Average Waiting Time (s)	151	145	145	145	142

Average Turn-Aaround Time (s)	204	198	198	198	195
Number of Context Switches (s)	10	5	5	5	5

Average Waiting Time

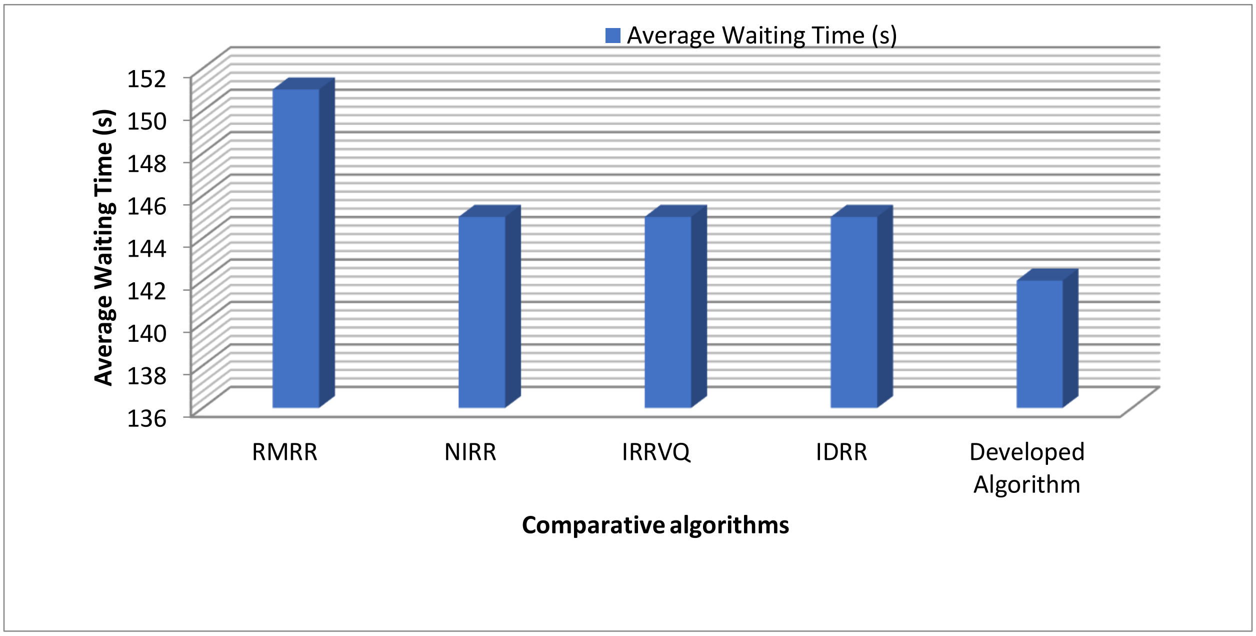

Figure 2: Comparison of Average Waiting Time

Average Turn-Around Time

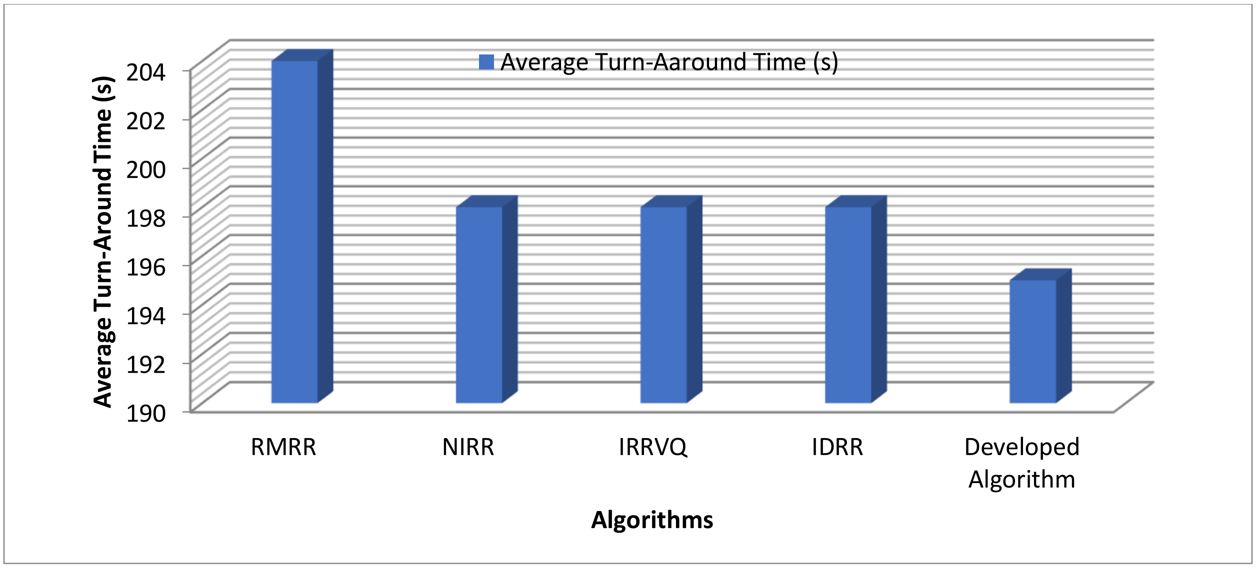

Figure 3: Comparison of Average Turn-Around Time

Number of Context Switches

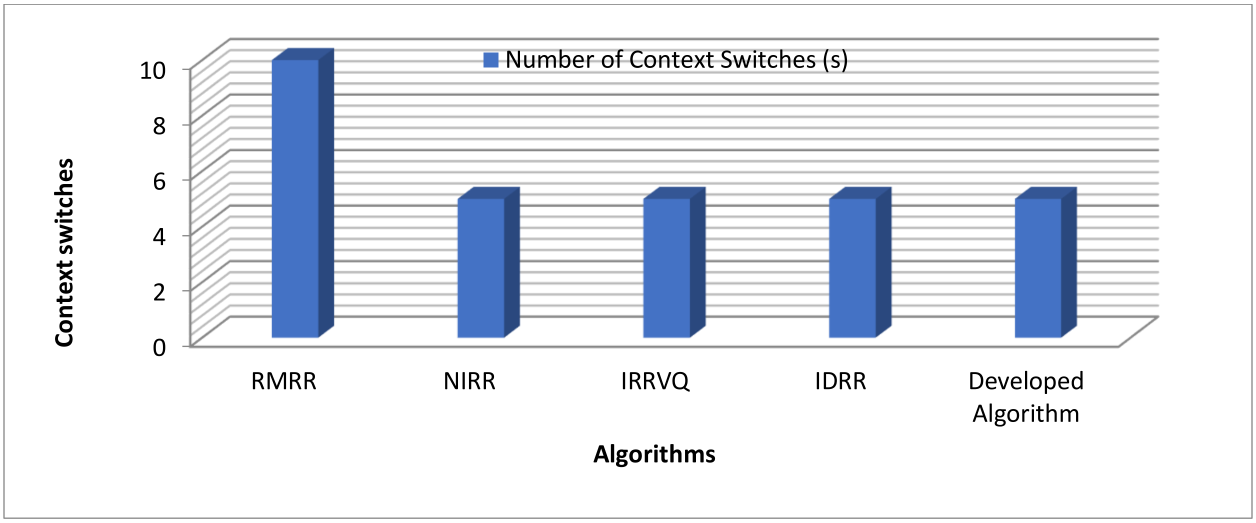

Figure 4: Comparison of Number of Context Switches

Graphical Comparison of Various Algorithms Considered

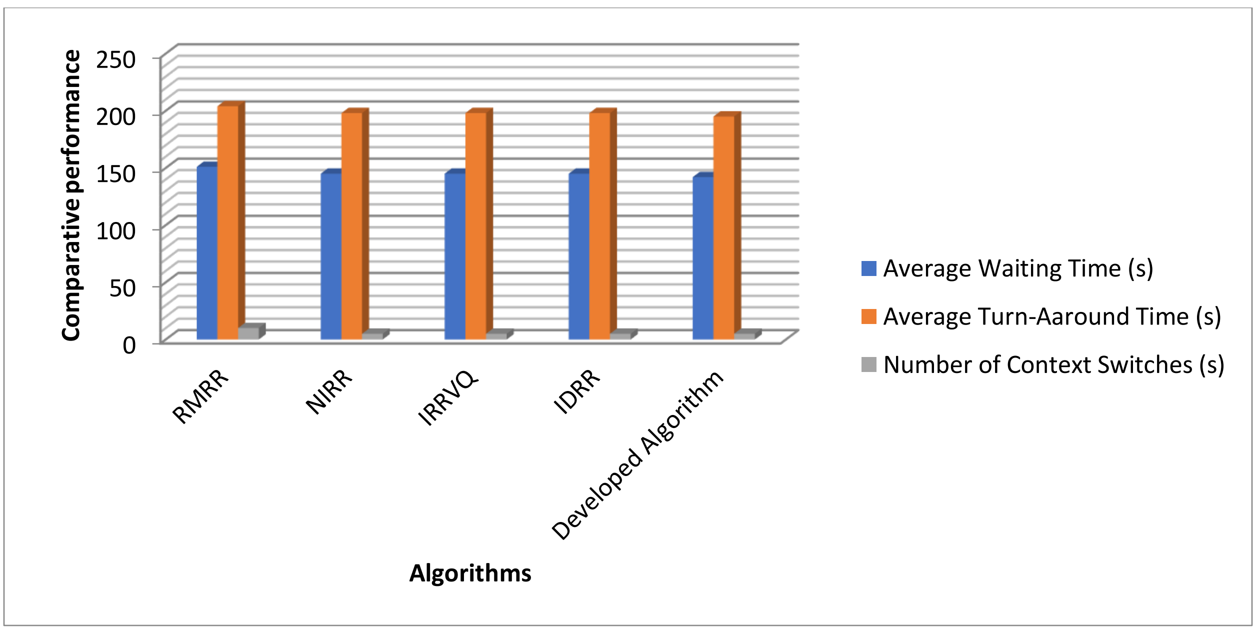

Figure 5: Comparison of Average Waiting Time, Average Turn Around Time and number of Context Switches

In this paper we proposed new algorithm known as Revamped Mean Round Robin (RMRR) CPU Scheduling algorithm .This algorithm is based on improvement and revised on the Simple Round Robin (RR) CPU Scheduling algorithm. Time Quantum is always an important factor for the Round Robin algorithm and always a Question arise – What is the optimal Time Quantum to be used in Round Robin Algorithm? For the effectiveness and better performance of the algorithm, result of this work provides an answer to this question by using dynamic time quantum. This developed algorithm together with other algorithm mentioned above were implemented in C# Programming Editor: visual studio 2017, and result were compared based on the following scheduling criteria namely, Average Waiting Time, Average Turn-Around Time, Number Context Switches. Based on the results obtained, it is observed that the developed algorithm performed better than the other algorithm because it produces minimal Average Waiting Time, Average Turn-Around Time, Number Context Switches compared.

REFERENCES

Abdulrahim, A., Abdullahi, E. S. and Sahalu, B. J. (2014). A New Improved Round Robin (NIRR) CPU Scheduling Algorithm.*International Journal of Computer Applications*, (0975 – 8887).

Abu, T. and Zamani, N. (2014). The Impact of the Cloud Based M-Learning in Higher Education,*International Journal of Advanced Research in Computer Science* Engineering 4,pp 569-574.

Abubakar, S.E., Yusuf, S.A., Obiniyi A. and Mohammed A. (2023) Modified Round Robin with Highest Response Ratio Next CPU Scheduling Algorithm Using Dynamic Time Quantum. *SuleLamido University Journal of Science & Technology* Vol. 6 No. 1&2 (March 2023), pp. 87-99 *https://doi,org/10.56471/slujst.v6i.363*

Ajit, S., Priyanka, G. and Sahil, B. (2010). An Optimized Round Robin Scheduling Algorithm for CPU Scheduling, *International Journal on Computer Science and Engineering (IJCSE)*, Vol. 02 No. 07, 2383-2385, pp 2382-2385.

Al-Tamimi, A.K., Jain R. and So-In, C. (2010). Dynamic resource allocation based on online traffic prediction for video streams, *Internet Multimedia Services Architecture and Applications (IMSAA), IEEE 4th International Conference on*, pp. 1-6 2010.

Amar, R. D., Sandipta, K. S. and Sanjay, K. S. (2015). An Optimised Round Robin CPU Scheduling Algorithm with dynamic time quantum, *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol. 5*,No.1, February 2015.

Amit, N., Sanjay C. and Gaurav S. (2011). Policy based resource allocation in IaaS Cloud in ELSEVIER - *Future Generation Computer Systems* 28,94–103.

Arya, G.P., Nilay, K. and Prasad D. (2018) An Improved Round Robin CPU Scheduling Algorithm Based on Priority of Process. *International Journal of Engineering &Technology*, 7 (4.5) (2018)238-241.

Arjun, G. (2017). Balanced Path Selection in Cloud Computing using round robin Algorithm, |*JSRCSEIT|Volume 2*|Issue 4|ISSN2456-3307

Atsuo, I., Taiki, M., Minoru I. and MizanurRahman S.K. (2010). *Proposal and Evaluation of Dynamin Resource Allocation Method Based on the Load Of VMs on IaaS*, *IEEE*,,978-1-4244-8704-2/11.

Baomin, X., Chunyan, Z., Enzhao, H. and Bin H. (2011). Job scheduling algorithm based on Berger model in cloud environment in *ELSEVIER* - *Advances in Engineering Software 42* 419–425.

Behera, H.S. Rakesh, M., Jajnaseni, P., Dipanwita, T. and Subasini, S. (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.

Bhavani, B. H., and Guruprasad, H. S. (2014). A Comparative Study on Resource Allocation Policies in Cloud Computing Environment. *An international journal of advanced computer technology.*

Biwas, D., Samsuddoha, A.L., Asif, R. and Ahmed M.(2023)Optimized Round Robin Scheduling Algorithm using Dynamic Time Quantum Approach in Cloud Computing Environment. *I.J Intelligent Systems and Applications*, 2023,1, 22-34 Published Online on February 8, 2023 by MECS Press (http://www.mecs-press.org/) DOI:10.5815/ijisa.2023.01.03.

Botterman, and Maarten. Policy Paper on IoT Future Technologies: Opening towards a New Reality..http://www.smart-action.eu/fileadmin/smart action/publications/PolicyPaperonIoTFutureTechnologies.pdf Issue brief no. D5.2. 39

Chanjuan, L. (2013). Analysis on the Design Patterns of Cloud Computing in Mobile Learning Systems. *Cloud Times, Mobile Cloud* [Online] *http://cloudtimes.org/mobile-cloud/*.

Chenn-Jung, H., TaiGuan, C., Chen, H. M., Wang, Y.H., Chang, S.C., Ching-Yu, L. andWeng, C.H. (2013). An Adaptive Resource Management Scheme in Cloud Computing in ELSEVIER - *Engineering Applications of Artificial Intelligence 26* 382-389.

Daniel, O. and Eduardo, O. (2010). Article: Is Cloud Computing the Solution for Brazilian Researchers? *International Journal of Computer Applications 6*(8):19–23.

Dash, R. A., Sahu, k. S., and Samantra, K. S. (2015). An Optimized Round Robin Cpu Scheduling Algorithm With Dynamic Time Quantum. *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT).*

Dogan, A. andOzguner, F. (2002). Matching and scheduling algorithms for minimizing execution time and failure probability of applications in heterogeneous computing, *IEEE Transactions on Parallel and Distributed Systems, pp*. 308–323.

Dorian, M. and Bernd, F. (2011*)*. Utility– based Resource Allocations for virtual machines in cloud computing, *IEEE*.

Fiad, A., Maaza, M. Z. and Bendoukha, H. (2020). Improved Version of Round Robin Scheduling Algorithm Based on Analytic Model, *International Journal of Networked and Distributed Computing*, Vol. **8**(**4**); December (2020), pp. 195–202. DOI: https://doi.org/10.2991/ijndc.k.200804.001; ISSN 2211-7938; eISSN 2211-7946 https://www.atlantis-press.com/journals/ijndc.

Gokilavani, M., Selvi S., and Udhayakumar, C. (2013). "*A Survey on Resource Allocation and Task Scheduling Algorithms in Cloud Environment".*

Guiyi, W., Athanasios, V., Zheng, Y. and Xiong, N. (2010). A game-theoretic method of resource allocation for cloud computing services*, in Springer, J Supercomput* 54: 252-269.

Gurdev, S., ShanuSood, and Amit, S. (2011). ―CM- Measurement Facets for Cloud Performance‖, *IJCA*, Lecturer, Computer science & Engineering, Eternal University, Baru Sahib (India), Volume 23 No.3.

Goudarzi, H. and Massoud, P. (2011). Maximizing Profit in Cloud Computing System Via Resource Allocation *IEEE 31st International Conference on Distributed Computing Systems Workshops*: pp,1- 6.

Hindman, B., Konwinski, A., Zaharia, M., Ghodsi, A. Joseph, D. Katz, R. Shenker, S. and Stoica.Mesos I, (2011). A platform for fine-grained resource sharing in the data center. *In NSDI*.

Igbal, M., Ullah, Z., Khan I.A., Aslam, S., Shaheer, H., Humayon, M., Salahudin, M. A. and Adeel, M. (2023). Optimizing Task Execution: The Impact of Dynamic Time Quantum and Priorities on Round Robin Scheduling. *Future Internet* 2023, 15, 104. https//doi.org/10.3390/fi15030104

Ishwari, S. R. and Deepa, G. (2012). A Priority based Round Robin CPU Scheduling Algorithm for Real Time Systems, *International Journal of Innovations in Engineering and Technology (IJIET), Vol. 1* Issue 3, pp 1-11.

Jiyin, L., Meikang, Q., Niu, J.W., Chen, Y. and Ming, C. (2010). Adaptive Resource Allocation for Premptable Jobs in Cloud Systems, *IEEE International Conference on Intelligent Systems Design and Applications*, pp. 31-36

Justin, Y. S., Moussa, T., and Abdallah K. (2011) Resource Planning for Parallel Processing in the Cloud, *in IEEE International Conference on High Performance Computing and Communications*, 978-0-7659-4538-7/11.

Kathuria, S., Singh, P. P., Tiwari, P. and Prashant, N. (2016). A Revamped Mean Round Robin (RMRR) CPU Scheduling Algorithm. *International Journal of Innovative Research in Computer and Communication Engineering.*

Khatri, J. (2016). An Improved Dynamic Round Robin CPU Scheduling Algorithm Based on Variant Time Quantum,*IOSR Journal of Computer Engineering (IOSR-JCE)* , PP 35-40.

Kochut, T., York, M. and Miller, G. (2010). Desktop Workload Study with Implications for Desktop Cloud Resource Optimization, *IEEE* 978-1-4244-6534-7/10 57

Kuo-Chan, H., and Kuan-Po, L. (2010). Processor Allocation policies for Reducing Resource fragmentation in Multi cluster Grid and Cloud Environments,*IEEE*, pp.971-976.

Lalit, K. Rajendra, S. and Praveen, S. (2011). Optimized Scheduling Algorithm, *International Journal of Computer Applications*, pp 106-109.

Manish, K. M. and Faizur, R. (2014). An Improved Round Robin CPU Scheduling Algorithm with Varying Time Quantum, *International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.4*, No.4.

Manish, K. M. and Abdulkadir, K. (2012). An Improved Round Robin CPU Scheduling Algorithm, *Journal of Global Research in Computer Science*, Volume 3, No. 6, pp 64-69. ISSN: 2229-371X

Mell, P. and Grance, T. (2011). The NIST definition of Cloud Computing. *NIST, Special Publication* 800–145, Gaithersburg, MD.

Mythili N.P., Korde P. and Dey P. (2017) An Advanced Approach to Traditional Round Robin CPU Scheduling Algorithm to Prioritize Processes with Residual Burst Time Nearest to the Specified Time Quantum. *IOP Conference Series: Materials Science and Engineering,* VIT University, Vellore-632014, Tamil Nadu, India.

Nayak, D., Malla, K. S. and Debadarshini, D. (2012). Improved Round Robin Scheduling using Dynamic Time Quantum, *International Journal of Computer Applications*, 0975 – 8887.

Neha, A. J. (2018). An Inproved Round Robin CPU Scheduling Algorithm, *Iconic Research and Engineering Journals*, Mar 2018|IRE Journals|Volume 1 Issue 9| ISSN: 2456-8880.

Olofintuyi S.S., Omotehinwa T.O., Oyekanmi E.O. and Olajubu E.A. (2019) An Improved Time Varying Quantum Round Robin CPU Scheduling Algorithm. *Achievers Journal of Scientific Research* Vol 2, Issue 2, December 2019 – p.31 -38.

Omotehinwa O. T., Azeez I .S., and Olafunyi S.S (2019) A Simplified Improved Dynamic Round Robin (SIDRR) CPU Scheduling Algorithm. *Internationa Journal of Information Processing and Communication* (IJIPC) vol.7 No.2 [December 2019], pp.122-140.

Onat, G. Y., Mathews, C., Neville, F. S., Guitouni, A., Gnat, S. and Coady, Y. (2010). Dynamic Resource Allocation in Computing Clouds using distributed Multiple Criteria. Decision Analysis. *IEEE*.

Oyetunji, E.O. and Oluleye A. E. (2009). Performance Assessment of Some CPU Scheduling Algorithms, *Research Journal of Information Technology1(1*): pp 22-26, 2009.

Pallab, B., Biresh, K. and Probal, B. (2017). Mixed Round Robin (MRR) Scheduling for Real Time System *International Journal of Computer Trends and Technology (IJCTT)* – Volume 49 Number 3July 2017.

Pandaba, P., Prafulla, K., Behera, K.P., and Ray B.N., (2016). Modified Round Robin Algorithm for Resource Allocation in Cloud Computing, *in International Conference on Computational Modeling and Security(CMS*) 878-890

Patel, R. and Patel, S. (2013). Survey on Resource Allocation Strategies in Cloud Computing *International Journal of Engineering Research & Technology (IJERT)*

Pradhan, P., Behera, K. P., and Ray, B. N. (2016). Modified Round Robin Algorithm for Resource

Allocation in Cloud Computing. *Elsevier*.

Prakash, G. L., Manish, P., and Inder, S. (2014). Data Encryption and Decryption Algorithms using Key Rotations for Data Security in Cloud. *International Journal of Engineering and Computer Science, 3(4)*, 5215-5223.

Rajput, S. I. and Gupta, D. (2012). A Priority based Round Robin CPU Scheduling Algorithm for Real Time Systems. *International Journal of Innovations in Engineering and Technology (IJIET)*.

Ritu, A. (2018). Resource Provision and Resource allocation in Cloud. *International Journal of Scientific Research in Computer Science, Engineer and Information Technology, 2018| JSRCSEIT|Volume 3*|Issue 3|ISSN2456-3307

Ronak, P., Sanjay P, (2013). Survey on Resource Allocation Strategies in Cloud Computing. *International Journal of Engineering Research & Technology (IJERT) Vol. 2* Issue 2, February- 2013 ISSN.

Sakshi, C., Sharma S., Kautish S., Alsallami S., Khalil E.M. and Mohamed A. (2022) A New Median-Average Round Robin Scheduling Algorithm: An Optimal Approach for Reducing Turn Around and Waiting Time, *Alexandria Engineering Journal*. 61,10527-10538.

Sheth, A. and Ranabahu, A. (2010). Semantic Modeling for Cloud Computing, ‖ *IEEE Computer Society — Semantics & Services, 2010 Part I.*

Singh, A., Goyal, P. andBatra, S. (2010). An Optimized Round Robin Scheduling Algorithm for CPU Scheduling, *International Journal of Computer and Electrical Engineering, Vol. 2,* No. 7, pp. 2383-2385.

Sundarraj, B. and Venkatesan K.G. (2011). *A Stochastic Model to Investigate Data centers for Better Performance.*

Vinothina, V., Sridaran R. and Padmavathi, G. (2012). Resource Allocation Strategies in Cloud Computing. *International Journal of Advanced Computer Science and Applications [IJACSA], Vol. 3,* No.6. ISSN: 2158-107X (Print), DOI: 10.14569/issn.2156-5570.

Yaashuwanth, C. and Ramesh. R. (2015). Design of Real Time Scheduler Simulator and Development of Modified Round Robin Architecture for Real Time System, *International Journal of Computer and Electrical Engineering, Vol. 10,* No. 3, 2010, pp. 43-47.

Zhang, Q., Cheng, L. and Boutaba, R. (2010). Cloud Computing: state of the-art and research challenges. Springer, pp. 7-18.