Journal of Theoretical and Applied Information Technology
15th July 2016. Vol.89. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REAL TIME SCHEDULING FOR DYNAMIC PROCESS EXECUTION USING ROUND ROBIN SWITCHING

[1]R AKHILESH REDDY, [2]G SIVA NAGESWARA RAO

[1]Student Of M.Tech(CSE)., Department of Computer Science Engineering, KL University, India.

[2]Assoc. Prof., Department of Computer Science Engineering, KL University, India.

E-mail: [1]rakhilesh574@gmail.com, [2]sivanags@kluniversity.in

ABSTRACT

Scheduling in operating systems is allocate certain amount of CPU time with use of different processes. Task scheduling is also one of the key process in running of different processors in with in life time. Round Robin (RR) is a popular scheduling algorithm allows to utilize the CPU short time for individual task scheduling events in real time process execution. The advancement of RR scheduling performs fine tuning for time slice which do not stipulated time to allocate them in events proceedings based on CPU scheduling. RR also maintains turnaround time, waiting time and response time with processing frequency of context switches. In this paper we improve the performance of RR with integer programming to refine in arrival time analysis in process scheduling with proceedings of all the requirement CPU processes. Every process has reasonable response and arrival time analysis in allocation of scheduling in process allocation. A method the usage of integer programming has been proposed to resolve equations that determine Changeable Time Quantum (CTQ) value that is neither too massive nor too small such that each system has reasonable response time and the throughput of the system is not decreased because of unnecessarily context switches.

Keywords: *Arrival Time, Burst Time, Time Slice, Turnaround Time, Number of Switches, CPU Scheduling, Residual Time, Cyclic Queue.*

1. INTRODUCTION

Management of different processes that handles the removal of running process from the CPU and selection of another process on the basis of particular strategy. Process scheduling is the essential process in multi programming operating system, such type of operating systems allows more than one process to be loaded into executable memory and then assign time for each process shares individual time for multiplexing events in scheduling of different operating processes. Scheduling queues refers to queues of procedures or gadgets. When the method enters into the system, then this system is placed right into an activity queue. This queue includes all procedures within the system. The working system also continues other queues which includes tool queue. Device queue is a queue for which multiple tactics are looking ahead to a selected I/O tool. Every tool has its personal tool queue.

CPU scheduling deals with hassle of identifying which of the process within the geared up queue is to be allocated to the CPU for processing .For doing this FCFS ,SJF ,precedence and spherical Robin Algorithms are to be had. spherical Robin (RR)([9],[10]) is one of the handiest scheduling algorithms for procedures in an working system, which assigns time slices to every system in equal quantities and in round order ([2], [3], [4], [5],[13]), handling all tactics without precedence ([22], [23], [24]), arguably, the principal problem in RR is the time slice ([4], [14]),[16], [19]) Round Robin scheduling is both simple and clean to put in force, and hunger-free. Effectiveness and performance of RR ([25],[26]) are bobbing up from its low scheduling overhead of O (1), which means scheduling the subsequent venture takes a constant time ([1],[7],[8]). In laptop technological know-how, a scheduling algorithm is the technique via which threads, processes or facts flows are given get entry to gadget assets (e.g. processor time, communications bandwidth). That is generally

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

achieved to load stability a machine efficaciously or reap a target first-class of carrier. The need for a scheduling set of rules arises from the requirement for maximum present day structures to perform multitasking (execute multiple technique at a time) and multiplexing (transmit a couple of flows concurrently).

The scheduler is worried especially with:

CPU usage – To Keep the CPU as busy as available.

Throughput - Wide variety of approaches that complete their execution per time unit.

Turnaround - Overall time among submission of a procedure and it's of completion.

Ready time - Quantity of time a process has been ready inside the geared up queue.

Response time - Quantity of time it takes from while a request became submitted until the primary reaction is produced.

Arrival Time-The Amount of time takes for request entire with appreciate to response time to every thread.

Fairness - identical CPU time to every thread.

An application takes a set of records files as input, processes the records, and produces a fixed of output records files. This working surroundings is named as "batch processing" due to the fact the input statistics are amassed into batches on documents and are processed in batches by this system. In spite of their long history, batch programs are nonetheless critical in most groups. Even as online systems at the moment are used when guide intervention isn't preferred, they're not well acceptable to the excessive-quantity ([15]), repetitive responsibilities. Therefore, even new systems normally incorporate a batch software for instances together with updating statistics on the give up of the day, producing reports, and printing files. Cutting-edge batch applications employ present day batch frameworks including Spring Batch, that's written for Java ([27], [28]) to offer the fault tolerance and scalability required for high-quantity processing. As a way to make sure high-speed processing, batch packages are frequently incorporated with grid computing solutions to partition a batch process over a huge range of processors. Either preemptive or non-preemptive ([11, [12]) may be utilized in batch environment.

2. RELATED WORK

Mok proposed a display scheduling version wherein vital sections of a system is nonpreemptible. Sha, Raj Kumar, and Lehoczky proposed the priority inheritance protocol (PIP) in which processes can inherit the higher precedence of a process they block and the concern ceiling protocol (PCP) to handle procedure synchronization with different sizes of essential sections. The concern ceiling of an aid is the priority of the best precedence procedure which may additionally use the useful resource. A process's useful resource request is blocked if its precedence is not any higher than the priority ceiling of any aid which has been grabbed by using every other procedure however has no longer yet been released.

For constant-precedence structures, Lehoczky proposed a bandwidth keeping mechanism, called deferrable server, which the execution price range may be suspended and deferred. Observe that the execution budget of a deferrable server is consumed while the server executes and replenished to authentic quantity while the server invokes. The principal disadvantage of the deferrable server is it can delay lower precedence procedures for more time than a periodic technique with the same duration and execution time of a deferrable server. Spuri and Buttazzo proposed the sporadic server to solve this problem. The sporadic server emulates a periodic method when its finances is damaged up into chunks to be replenished at exclusive instances. Such a technique also improves the schedulability of deferred server.

There are various researches going around the globe on enhancing the overall performance of spherical robin set of rules. The author of [1] proposes a fuzzy approach to find the perfect time slice for the strategies. They have offered the consequences the usage of distinctive simulations. The novelists of [2] advise an average based method to discover the time slice, combining the traditional shortest task first and round Robin algorithms. The authors of [3] proposes a brand new approach the usage of most and minimal burst time of the set of strategies in the ready queue and calculating a modified time slice. The authors of [4] talks approximately calculating the time slice the usage of median and highest burst time after which executing the processes as in keeping with the brand new calculated time slice. The researchers of [5] furnished a mathematical model for calculating the waiting time and turnaround time. The authors of [6] talks about calculating the imply of the burst instances of all the tactics and then finds the distinction among the imply of the burst time and the burst time of a

He input to the ANN is the value of exponent of reactive power load-voltage characteristic (n_q) and the output is the desired proportional gain (K_P) and integral gain (K_I) parameters of the SVC. Normalized values of n_q are fed as the input to the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ANN the normalized values of outputs are converted into the actual value. The process of particular manner and allocates the CPU to the method which has the most distinction.

3. PROPOSED ALGORTHIM

Historically we are able to be editing the time slice of most effective those methods which require a slightly greater time than the allocated time slice cycle(s) [1]-[3]. The ultimate approaches might be done within the fundamental spherical Round Robin way. Hence we calculate the reaming burst time and no. cycles for each method [2] [5]. Based totally at the reaming burst time, we type the system, if the reaming burst time is less than or equal to the only time slice then execute the equal process otherwise go for subsequent procedure. If more than one system having the same ultimate burst time then use the Shortest task First Scheduling set of rules.

As recognized, that Integer Programming (IP) ([25])trouble is any mathematical optimization or feasibility program in which some or all the variables are limited to be quintessential. In many settings, the term integer software is used as quick-hand for integer linear programming.
The integer programing is based up on the changeable time quantum policy (CTQ) .in this method we consider time slice based up on the conditions it can be changed with the help of quantum requirement method can be used…

Input:
TS: Time Slice
BT: Burst Time
RBT: remaining Burst time
RBT [Pi] = BT [Pi] % TS
NOC: Number of Context Switches
NOC [Pi] = ceil (BT [Pi] / TS), in which ceil feature offers the biggest integer extra than or identical to the wide variety.

Round Robin Scheduling Procedure for Process Allocation

Step1: begin
Step2: Make a prepared queue of the processes say Request.
Step3: Calculate the Time Slice. (TS = floor ((□ (BT [Pi]) / N)))
Step4: Calculate the remaining burst time and quantity of Cycles for all approaches (RBT = BT [PI] %TS, NOC = BT [PI]/TS)
Step5: sort the all techniques based totally on closing burst time & NOC.
Step6: pick the technique from the geared up queue and allocate the CPU to it for a
Time c programming language of up to 1 time quantum.
Step7: If the closing CPU burst time of the presently running technique is less than or equal to the one time quantum then allocate CPU once more to the currently walking technique, otherwise to the following process.
Step8: Repeat Step6 & Step 7 until all process are scheduled.
Step9: give up

4. RR WITH INTEGER PROGRAMMING SCHEDULING

As recognized, that Integer Programming (IP) trouble is any mathematical optimization or feasibility program in which some or all the variables are limited to be quintessential. in many settings, the term integer software is used as quick-hand for integer linear programming.
The integer programing is based up on the changeable time quantum policy (CTQ) .in this method we consider time slice based up on the conditions it can be changed with the help of quantum requirement method can be used…

$$Minimize\ z = cx$$

Subject To

1)x_j integer j€ J {1……m},J#0
2)X_j>0, j=1…….m,
3)A_x=b

In this paper, a method using integer programming hassle has been proposed that comes to a decision a cost this is neither too huge nor too small such that each manner has were given reasonable response time and the throughput of the machine is not decreased due to unnecessarily context switches. This overall performance achieves extremely good throughput values in arrival of different processes with switch linear programming.
The process of Linear Programming with round robin process explained with following example:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5. RESULTS AND GRAPHS:

EXAMPLE 1: CONSIDER THE FOLLOWING SET OF PROCESSES, ASSUMED TO HAVE ARRIVED AT DIFFERENT TIME.

Table 1: Consider The Five Process And Their Time Slice Is Given In The Following Table

Number of Processes	Burst Time	Arrival Time
PS1	12	0
PS2	15	12
PS3	23	16
PS4	37	19
PS5	21	22

Table 2: RR With Integer Policy Of Allocation Of Different Tasks In Scheduling Using CTQ.

Job id	Service time	Arrival time	Start time	Finish time	Quantum time R1	R2	R3	pre-emption	Average turnaround time	Average waiting time
Ps1	12	0	0	12	12				12	0
Ps2	15 1	12	12 26	26 27		16		End of quantum ps3 starts	15	11
Ps3	23 9	16	27 41	41 50		16		End of quantum Ps4 starts	34	13
Ps4	37 15	19	50 72	72 97			22	End of quantum ps5 starts	78	38
Ps5	21	22	97	108			22		86	65
Mean									45	25.4

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 3. Comparison Table For Various Scheduling Algorithms Using Dynamic Time Slice

Algorithm	Avg Turnaround time	Avg Waiting Time	No Of Context Switches
Round robin	59	40.4	7
Round robin with dynamic time slice	53.2	31.8	4
My proposal	45	25.4	5

In the above Table My Proposal1 Algorithm has minimum Avg. TAT, minimum TWT and minimum no. of Context switches. Due to context switching, the processor idle time is low and resource utilization is very high.

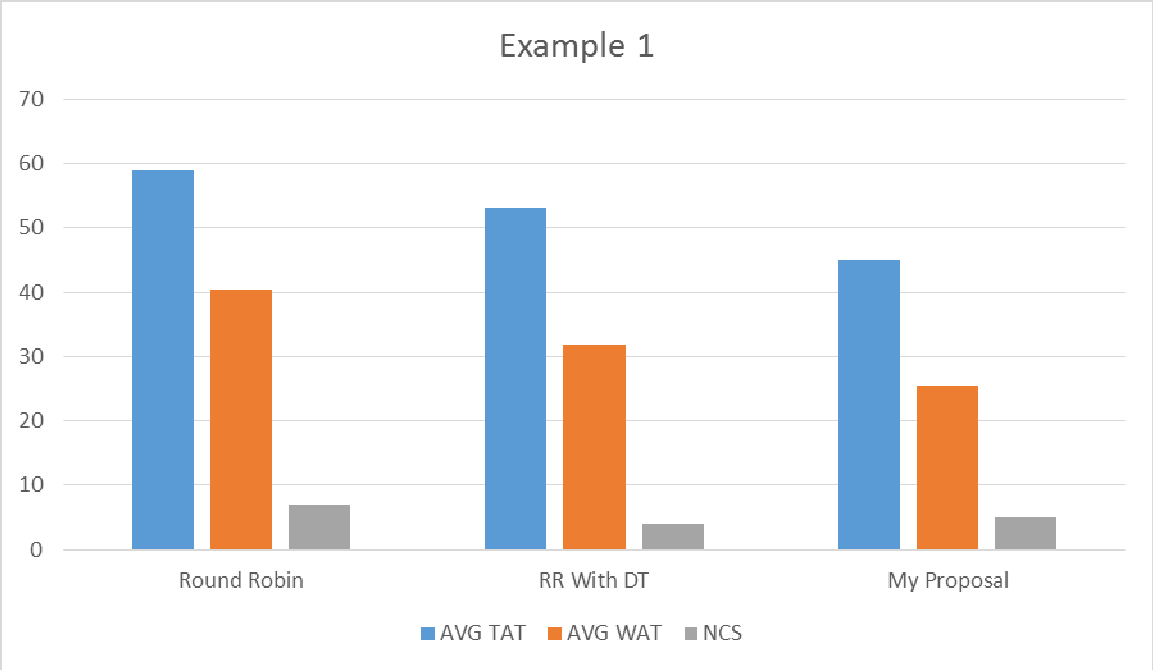

Figure 1. Comparison Table For Various Scheduling Algorithms Using Dynamic Time Slice

With the reference of the [7], we altered the calculation with preferred results over before plan planning process. We plot takes lower time than the fundamental round robin technique, which is appeared in the Figure 1. In the above Figure 1 My Proposal1 Algorithm has least Avg. TAT, least TWT and least no. of Context switches. Because of less setting exchanging, the processor moving time is low and asset usage

6. CONCLUSION

In this paper a development for the conventional spherical robin set of rules is proposed that is being supported by way of a fixed of hypothetical examples and a higher amount of development is located. In light of the effectiveness and the performance of the RR set of rules, we advanced a brand new CPU nearby scheduling based on RR named as RR with Integer Programming. From the simulation look at, we get an important conclusion; that the overall performance of CTQ policy is higher than that of RR with dynamic time slice.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The work supplied on this paper may be increased in lots of directions. Some of the instructions are:

- Employing exceptional overall performance criteria for contrast together with the make span. The make span is defined as most time needed to complete the execution of all the responsibilities arriving to the device.
- Making use of scheduling approach on obligations that have dependencies amongst every different.
- Reading performance in actual time applications where tasks have priorities and closing date constraints.

REFERENCES

[1] G. Siva Nageswara Rao, N. Srinivasu, S.V.N. Srinivasu, G. Rama Koteswara Rao, "Dynamic Time Slice Calculation for Round Robin ProcessScheduling Using NOC", International Journal of Electrical and Computer Engineering (IJECE) Vol. 5, No. 6, December 2015, pp. 1480~1485 ISSN: 2088-8708.

[2] G Siva, Nageswara Rao, *et al.*, "Comparison of Round Robin CPU Scheduling Algorithm with Various Dynamic Time Quantum", *International Journal of Applied Engineering Research,* Vol. 9, No. 18, pp. 4905-4916, 2014.

[3] G Siva, Nageswara Rao, *et al.*, "A NEW PROPOSED DYNAMIC DUAL PROCESSOR BASED CPU SCHEDULING ALGORITHM", *Journal of Theoretical and Applied Information Technology 20th March 2015,* Vol.73, No. 2, 1992.

[4] Bashir Alam, "Fuzzy Round Robin CPU Scheduling Algorithm", *Journal of Computer Science,* Vol. 9, No. 8, pp. 1079-1085, 2013.

[5] Lalit Kishor, Dinesh Goyal, "Time Quantum Based Improved Scheduling Algorithm", *Issue International Journal of Advanced Research in Computer Science and Software Engineering,* Vol. 3, 2013.

[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*, Vol. 2, No. 11, 2012.

[7] K. N. Rout, G. Das, B. M. Sahoo, and A. K. Agrawalla, "Improving Average Waiting Time Using Dyanamic Time Quantum".

[8] Shahram Saeidi, Hakimeh Alemi Baktash, Determining the Optimum Time Quantum Value in Round Robin Process Scheduling Method, International Journal of Information Technology and Computer Science, pp: 67-73 2012.

[9] H. S. Behera, Simpi Patel, and Bijayalakshmi Panda, "A New Dynamic Round Robin and SRTN Algorithm withVariable Original Time Slice and Intelligent Time Slice for Soft Real Time Systems".

[10] Supriya Raheja, Reena Dhadich, and Smita Rajpal, "An Optimum Time Quantum Using Linguistic Synthesis for Round Robin CPU Scheduling Algorithm", *International Journal on Soft Computing (IJSC),* Vol. 3, No. 1, 2012.

[11] Nayana Kundargi, Sheetal Bandekar, "CPU Scheduling Algorithm Using Time Quantum for Batch System,"
International Journal Of Latest Trends In Engeneering And Technology (IJLTET), 2013.

[12] Himanshi Arora, Deepanshu Arora, Bagish Goel, and Parita Jain, "An Improved Scheduling Algorithm", *International Journal of applied Information Systems(IJAIS), Foundation of Computer Science FCS,* New York, USA, Vol. 6, No. 6, pp 7- 9, 2013.

[13] Sandeep Negi, "An Improved Round Robin Approach Using Dyanamic Time Quantum For Improving Average
Waiting Time", *International Of Computer Applications,* Vol. 69, No. 14, pp. 12-16, 2013.

[14] Rakesh Patel, Mrs. Mili Patel, "SJRR CPU Scheduling Algorithm", *International journal of Engineering And*
Computer Science, Vol. 2, No. 12, pp. 3396-3399, 2013.

[15] Adela Jamal, Aiman Jubair, "A Varied Round Robin Approach Using Harmonic Mean Of The Remaining Burst Time Of Processes", *Special Issue of International Journal of Computer Applications*, *3rd International IT Summit Confluence*, pp. 11- 17, 2012.

[16] Mohd Abdul Ahad, "Modyfying Round Robin Algorithm For Process Scheduling Using Dynamic QuantumPrecision", *Special Issue Of International Journal of Computer Applications On Issues AND Challenges In Networking ,Intelligence And Computing Technologies, ICNICT 2012,* pp. 5-10, 2012.

[17] C. Yaashuwanth, Dr. R. Ramesh, "A New Scheduling Algorithms for Real Time Tasks", *(IJCSIS) InternationalJournal of Computer Science and Information Security,* Vol. 6, No. 2, pp. 61-66, 2009.

[18] Sanjaya Kumar Panda, and Sourav Kumar Bhoi, "An Effective Round Robin Algorithm using Min-Max Dispersion Measure", *IJCSE,* Vol. 4, No. 01, pp. 45-53, 2012.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[19] Aashna Bisht, Mohd Abdul Ahad, Sielvie Sharma, "Enhanced Round Robin Algorithm for Process Scheduling using varying quantum precision", *Proceedings of ICRIEST AICEEMCS,*pp. 11-15, 2013.

[20] R. N. D. S. S Kiran, Polinati Vinod Babu, and B. B. Murali Krishna, "Optimizing CPU Scheduling for Real Time Applications Using Mean-Difference Round Robin (MDRR) Algorithm", *ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India,* Vol. 1, Advances in Intelligent Systemsand Computing, Vol. 248, pp. 713-721, 2014.

[21]. Siva Nageswara Rao, G., Srinivasu, S.V.N., Srinivasu, N., Ramakoteswara Rao, G.Enhanced precedence scheduling algorithm with dynamic time quantum EPSADTQ) (2015) Research Journal of Applied Sciences, Engineering and Technology, 10 (8), pp. 938-941.

[22.] Siva Nageswara Rao, G., Srinivasu, N., Sagar, K.V.D., Sai Madhuri, P.Comparison of round robin CPU scheduling algorithm with various dynamic time quantum (2014) International Journal of Applied Engineering Research, 9 (18), pp. 4905-4916.

[23]. Siva Nageswara Rao, G., Srinivasu, S.V.N., Srinivasu, N., Naga Raju, O.A new proposed Dynamic dual processor based CPU scheduling algorithm(2015) Journal of Theoretical and Applied Information Technology, 73 (2), pp. 226-231.

[24]. Rao, G.S.N., Srinivasu, N., Girish Kumar, K., Abhishek, B.An enhanced dynamic Round Robin CPU scheduling algorithm (2014) International Journal of Applied Engineering Research, 9 (15), pp. 3085-3098.

[25]. Rao, G.S., Krishna, C.V.P., Rao, K.R.Extreme Programming for service-based application development architecture(2014) Proceedings of the 2014 Conference on IT in Business, Industry and Government: An International Conference by CSI on Big Data, CSIBIG 2014, art. No. 7056955.

[26]. Rao, G.S., Krishna, C.V.P., Rao, K.R.Multi Objective Particle Swarm Optimization for Software Cost Estimation (2014) Advances in Intelligent Systems and Computing, 248 VOLUME I, pp. 125-132.

[27]. Rao, G.S., Krishna, C.V.P., Rao, K.R.Rational unified process for service oriented application in extreme programming (2013) 2013 4th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2013, art. No. 6726586.

[28]. Daya Sagar, K.V., Sivanageswara Rao, G., Srikanth, T., Raghavendra, K. A relational analytic platform with Hadoop using the On Demand Integration (ODI) capability (2014) International Journal of Applied Engineering Research, 9 (13), pp. 2095-2102.