

Comparative Study of Parallel Scheduling Algorithm for Parallel Job

Priya Singh
M.Tech (CSE)
Institute of Technology &
Management
Aligarh, India

Zafaruddin Quadri
Al- Barkaat College of
Graduate Studies
Aligarh, India

Anuj Kumar
M.Tech (CSE)
Institute of Technology &
Management
Aligarh, India

ABSTRACT

Job scheduling is a technique which is applied on parallel computing systems, whose main focus is to measure the parameters of a system. For job scheduling menu algorithms are used in technical line such as priority based FCFS reservation, backfilling, improved round robin scheduling all are used to measure the parameters of a parallel computing. Since they all have some limitations and advantages to use all processors equally. This paper describes the various job scheduling algorithm such as priority based FCFS, PFCFS1, PFCFS2, PFCFS3 of static job scheduling algorithm.

General Terms

Parallel Computing, Scheduling algorithm

Keywords

FCFS, parallel processing, Static Job, Speedup, Interdependency of graph,

1. INTRODUCTION

Scheduling algorithms is a way by which work specified by some means is assigned to resources that complete the work. Schedulers are normally implemented so they keep all complete resources busy, allow multiple users to share system resources effectively, or to get a target quality of service. A scheduler may aim at one of many goals to achieve such as maximizing the throughput and minimizing the average waiting time [22,23,24,25]. Scheduling algorithm plays a basic role among different process in parallel computing. Scheduling algorithms are also used to schedules the CPU jobs. Where execution time of job is reduced and speed of CPU will be more. Scheduling is also done in real life. Its main concept came from real world such as maintaining a queue on a ticket window. Scheduling algorithms are used to allocate the jobs to the processors.[1][2][3][4][5] There are various scheduling algorithms for the allocation of jobs to the processors such as- First Come First Serve (FCFS), Priority Based First Come First Serve (PFCFS1), Priority Based First Come First Serve (PFCFS2), Priority Based First Come First Serve (PFCFS3), Back filling algorithm, Easy Backfilling algorithm, Flexible Backfilling algorithm, Multiple Queue Backfilling algorithm, Gang Scheduling algorithm, Paired Gang Scheduling algorithm, Gang earliest deadline First Scheduling algorithm.

1.1 Types of Scheduling Algorithm

There are various types of scheduling algorithms are as follows:-

1.1.1 Non-Preemptive scheduling algorithm

This scheduling algorithm occurs only when a process voluntarily enters to the wait state or terminates. It is only method that can be used on certain hardware platforms

because it does not require the special hardware needed for preemptive scheduling. When a process enters the state of running the state of that process is not deleted from the scheduler until it finishes its scheduler time [2]. A non-preemptive job cannot be interrupted.

1.1.2 Preemptive scheduling algorithm

Preemption means termination of any job can be possible at any time. This scheduling can be possible in all possible cases. A preemptive schedule is prioritized. The highest priority process should always be the process that is currently utilized. Jobs are usually assigned with priorities, at somehow a running job can be interrupted for some time and resume later when the priority job has finish its execution it is known as preemptive scheduling.

1.1.3 Long term scheduling algorithm

Long term scheduling is done when a new process is created. If the number of ready processes in the ready queue becomes very high than there is a overhead on the parallel computing for maintaining long jobs, by this context switching and dispatching increases. It also controls the degree of multi-parallel systems following certain policies which decided whether a system can manage a new job or more than one job can also submitted to processor.

1.1.4 Medium term scheduling algorithm

Medium term scheduling is work with the memory management of processors. Its efficient interaction with short term schedulers it is essential for system performances.

1.1.5 Short Term scheduling algorithm

Short term scheduling algorithm is concerned with the allocation time of processors to processes. Scheduling policy is an overall system design issue and determines the "character" of the operating system from the user's point of view giving rise to the traditional distinctions among "multi-purpose", "time shared" etc..

2. POLICIES FOR SCHEDULING ALGORITHMS

There are three types of scheduling algorithm policies.

2.1 Policies for Long Term scheduling

It is simple a FIFO policy, which determines which job is admitted to the processors for processing and when, and which ones should be exited. All jobs requests a submission of a batch jobs, or an user trying to go into a time shared processors system, further requests can refused or enqueued for later processing. The higher the number of processes, in fact the smaller the time each of them may. Another policy is priority schemes in context of long term scheduling it has different meaning than in dispatching: here it affects the

choice of a program to be entered the system as a process, there the choice of which entered the system as process, there the choice of which ready process should be executed.

2.2 Policies for Long Term scheduling

It is a policy which determines when jobs are to be suspended and resumed. It is specially concerned with memory management, hence it's very often designed as a part of memory management systems of a processors.

2.3 Policies for Short Term scheduling

It is a policy which determines which of the ready jobs can have processors and for how much time. A short term scheduler is essential for system performances, specially in virtual memory of processors, due to this paged system the pager processes is usually run at a very high priority level.

Structure of Job Scheduler

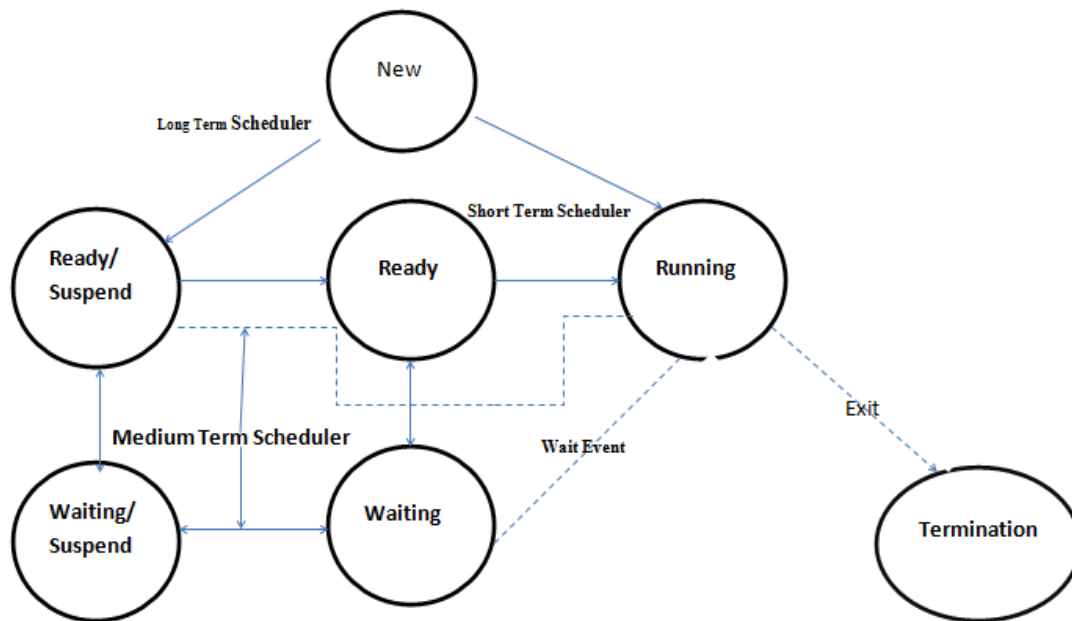


Figure 1. Job Scheduler

3. REVIEW ON VARIOUS JOB SCHEDULING ALGORITHMS

The various scheduling algorithm are as follows:

3.1 First Come First Serve (FCFS)

This technique provides the jobs to come early and execute first. This process is known as first come first serve job scheduling algorithm. In this technique those jobs came late are placed in a waiting queue form, those jobs are late place them at the end of the queue or tail of the queue. This scheduling process has bad performance, more waiting time and lesser throughput [7].

3.2 PFCFS1

It is an advanced version of FCFS job scheduling algorithm in which there is no time frame to preempt a job gang, automatically after 10 minutes it preempt a job gang. A job gang will preempt after 10 minutes until one gang will not blank, a new gang will not go for the processing.

3.3 PFCFS2

It is an advanced version of PFCFS1 in which each job is preempt after 1 minute and already suspended job will start again for 1 minute and new job gang will start. In this manner a wide gang job will complete.

3.4 PFCFS3

It is an advanced version of PFCFS2 in which a job gang will start processing if it is wait for at least 10 minutes in a queue.

Then the job will go for the execution. After executing the suspended job will go for the execution.

3.5 Shortest Job First

In this technique less execution time jobs are assigned to processors. In this technique scheduler kept the jobs in such a way that those have less execution time they will put in the starting of the queue and those have more execution time they will put in the end of the queue. In SJF average waiting time is minimum because it executes less execution time jobs earlier and more execution time jobs in last. A job is divided into two compulsory and optimal part, this optimal portion is scheduled by using shortest job first.

3.6 Round Robin Scheduling

In this scheduling algorithm provide a time slice for each job. The ready jobs are placed in queue and new jobs are placed into end of the queue. In this technique the processor efficiency will worst when the time quantum is too short and when it is too long it will generate a poor response. It is best responsive and effective in nature but the waiting time and turnaround time are less due to known time amount nature. The scheduler provides a known time slice to each job and scheduler repeats them in a cycle. It is same as FCFS but preemption is performing to transfer them among processes. A new round robin scheduling algorithm is proposed with less context switching and provides a best waiting time, turnaround time as compared to round robin scheduling.

3.7 Priority Scheduling Algorithm

In this scheduling processors assigns fixed priority to each process. A low priority job is interrupted when a high priority job is arrived .The high priority jobs have smaller waiting time and there may be a condition of starvation for the low priority jobs.

3.8 Backfilling Algorithm

FCFS is improved by increasing the utilization of the system resources and by decreasing the average waiting time of the job in the queue of the scheduler. If the job at the head of the queue is waiting for resources then it is possible for other short jobs to be scheduled and executed .Maximum execution time for each job must be known for backfilling algorithm. It sends shortest jobs first to increase the response time and node utilization.

3.9 Conservative Backfilling

This backfilling approach allows each job to reserve the resources it needs, when it is inserted into the job queue. If enough nodes are available then the job will be executed .Otherwise the job has to wait and later arriving small job will be executed if enough node for that job are available .It can predict when each job will execute .Starvation will not generate due to reservation for every job is done earlier, when it is submitted for execution.

3.10 EASY Backfilling

It is specially designed for super computers only, the first job in the submission queue is allow to reserve the resources it needs .It is more aggressive approach because it increases resource utilization even if it does not guarantee that a job is not delayed by another job or submitted later on.EASY backfilling has good opportunities to fill all the backfills under accurate user estimates.

3.11 Flexible Backfilling

It is the approach in which jobs are prioritized with some priority policy. In a waiting queue for each job a slack factor (a slack factor is assumed to be equal to three, i.e. a job can waits three times of average waiting time.) is provides, and for the computation of waiting time for each job there is requirement of slack factor and waiting time.

3.12 Multiple Queue Backfilling

This approach is proposed by Lawson and Amirin .In this approach each job is given in a queue according to jobs expected running time and each queue is given to parts of a parallel system and only jobs of the parallel queue can be run.

3.13 Gang Scheduling

It is the approach in which the ,a task which consists jobs grouped together and scheduled concurrently in a set of processors .At any instant of time a highest priority job executed first,but a scheduled job can be preempted before completion. Preemption can possible only when a higher priority job is released.

3.14 Paired Gang Scheduling

In this approach ,the resource utilization improve very well without any cause of interference between the processes of completing jobs. Because a process which occupies the processors most of the time will be matched with a process that occupies the input /output devices.

4. COMPARISON OF EXISTING JOB SCHEDULING ALGORITHMS

These scheduling algorithms are used and implemented in parallel job scheduling algorithms and considered in Table 1

Table 1.Comparison of Parallel Job Scheduling Algorithms

S.No.	Scheduling Method	Scheduling Factors	Researching
1.	FCFS	Job submission time, job start time, job completion time	Where use of resource is less and response time is more
2.	PFCFS1	Waiting time, preemption of job in gang format	A gang will preempt after 10 minutes
3	PFCFS2	Waiting time, running time and preemption of job in gang format	A gang will preempt after 1 minute
4	PFCFS3	Running time, waiting time and preemption of jobs	Job will preempt when it waits for 10 minutes in a queue
5	Shortest Job First	execution time of job, optimal part of scheduling	It use less execution time of job, optimal part division of scheduling required
6	Round Robin Scheduling	Waiting time ,turnaround time , execution time of job	Response time is more due to preemption ,and waiting time is more
7	Priority Scheduling Algorithm	Waiting time. Starvation condition	Waiting time for less jobs ,but more for long jobs ,and there may be condition of starvation
8	Backfilling	Execution time of job ,priority of the jobs in the queue	Shortest jobs will proceed first and increase response time, and increase processor utilization
9	Conservative Backfilling	Reservation time of all jobs ,estimated time of execution time of every job and arrival time of the job	It increase the processor utilization
10	Easy Backfilling	Execution time of job	To specify more execution time of jobs
11	Flexible Backfilling	Assumed time of job execution	Resource utilization is more ,and shortest jobs will execute first

12	Multiple Queue Backfilling	Reservation and priorities of jobs ,submission time of job , estimated time of execution	Cost of reservation of jobs will be lesser
13	Gang Scheduling	Average waiting time, average slow down of processors, average loss of capacity	Processor utilization will be more and increase jobs waiting time
14	Paired Gang Scheduling	Resource utilization, interference of jobs	Resource utilization is more, no interference of jobs will be allowed

5. PERFORMANCE OF SCHEDULING CRITERIA

There are so many different criteria to find best scheduling algorithm for a particular situation and environment. Such as Processor utilization, throughput, turnaround time, waiting time, efficiency and speedup.

Processor Utilization

Parallel computing support multiprocessing system. When a large amount of data is processed by more than one processors where distribution is not proper. Processor utilization read the time utilization of processor for which the processor is activated.

Processor utilization (%) = (Total time taken by the jobs /total number of jobs)*100.

Speedup

It is the ratio of time taken by job in serial manner to the time taken by job in parallel manner.

SpeedUp = Time taken in serial manner/Time taken in parallel manner.

Throughput

Total number of jobs completed per unit time. It may range from ten per second to one per hour depend upon the specific processes.

Turnaround time

Time required for a particular process to complete from submission time to completion time of the processes.

Waiting time

Total time processes spend in the ready queue, waiting for the execution to get the CPU.

Efficiency

Capacity to complete the jobs with in the time specified for each job.

Fairness

Fairness is the term which does not place any job in the starvation condition. Hard fairness provides every job a processor to complete the processing. A scheduler must want a utility which serviced each processor each job equally. The mixing of jobs is a production of external policies .Fairness is measured as the standard deviation of turnaround time and the fairness index. Fairness of a job can be affected by the later arriving of jobs .Average turnaround time is more and utilization is low this difference is known as 'unfairness'.

Response Time

The interval of the time from the moment of a service is requested until response begins to received .In parallel systems a better measure of responsiveness from the users point of view after turnaround time is response time.

6. CONCLUSION AND FUTURE SCOPE

After review the all existing methods we conclude that the paired gang scheduling algorithm generates best results on the basis best processor utilization and no interference of jobs are allowed in parallel scheduling .Very close results are generate by PFCFS1, PFCFS2, PFCFS3 but best results are generate by paired gang scheduling algorithm . In future we can further change in the interference of jobs should allowed in the parallel processing.

7. ACKNOWLEDGMENTS

The authors are thankful to the reviewers for their constructive suggestions and valuable comments. The author is also thankful to Mr. Anuj Kumar, Asst. Prof. of CSE Institute of Technology & Management, Aligarh for his constant support. I am also very thankful of my friends (Mrs. Khushboo Singh).

8. REFERENCES

- [1] S. N. Priyanka and P. S. Verma, "A Novel CPU Scheduling algorithm-Preemptive and Non Preemptive", International Journal of Modern Engineering Research .Vol. 2, 2012.
- [2] Pallab Banerjee ,Probal Banerjee Shweta Sonali Dhal, performance evaluation of a new proposed average Mid Max Round Robin (AMMRR) scheduling algorithm with Round Robin scheduling algorithm ,International Journal of Advanced Research in Computer Science and Software Engineering .Volume 2,Issue 8 August 2012.
- [3] R. Mohany, H.S. behera ,DebashreeNayak, A new Proposed Dynamic Quantum with Re-Adjusted Round Robin scheduling algorithm and its performance analysis .International Journal of computer application .Vol 5,AUGUST 2010.
- [4] SarojHiranwal,Dr. K.C. Roy ,Adaptive Round Robin scheduling using Shortest Burst Approach Based on Smart Time slice ,International Journal of Data Engineering (IJDE), Vol-2,Issue 3.
- [5] Pabitra Pal Choudhury, Operating systems principals and Design, PHI Publication.
- [6] Silberschatz, A. Peterson, J.L., and Galvin, P.B., Operating System concepts, (Addison Wesley 1991),(3rd Edition),pp. 108-118.
- [7] S. Swanson, K .Michelson, A, Schwerin and M.Oskin .WaveScalar. In the 36th Annual International Symposium on Microarchitecture (MICRO-36), 2003.
- [8] U.Schwiegelshohn and R.Yahyapour . Analysis of first come first serve parallel job scheduling. In SODA '98:Proceedings of the ninth annual ACM-SIAM symposium on Discrete Algorithm.USA 1998.
- [9] U.Schwiegelshohn and R.Yahyapour .fairness in parallel job scheduling .Journal of Scheduling, 3(5):297-320, 2000.
- [10] RadheShyam in Improved Mean Round Robin with Shortest Job First Scheduling. India Vol-4, Issue 7, 2014.
- [11] Manish Kumar Mishra in An Improved Round Robin CPU Scheduling Algorithm Vol-3, No.6 2012.
- [12] Imran Queshi in CPU Scheduling Algorithm Vol-05,

- Issue:04 ,Pages 1968-1973(2014)ISSN :0975-0290.
- [13] Hong Jiang ,Tianwai NI “PB-FCFS—A Task Scheduling Algorithm Based on FCFS and Backfilling Strategy For Grid Computing “.978-1-4244-5228-6/09
- [14] S. Gomathi , “ An adaptive grouping base jobs scheduling in grid computing ICSCCN 2011”.
- [15] Rakesh kumar Yadav, Abhishek Kumar Mishra, Naven Prakash, Himanshu Sharma an improved Round Robin Scheduling algorithm for CPU scheduling. Vol 2 No. 4, 2010, 1064-1066.
- [16] Ajeet Singh, Priyanka Goyal, “ An optimized Round Robin Scheduling Algorithm for CPU scheduling “ (IJCSE) International Journal on Computer Science and Engineering, Vol-2,No.07,2010.
- [17] “A.STenenbaum. 2008” Modern Operating Systems.Third Edition.Printice Hall, ISBN: 13:9780136006633.PP: 11004.
- [18] Lalit Kishore and Dinesh Goyal, “Time Quantum Based Improved Scheduling Algorithms”. International Journal of Advance Research in Computer Science and Software Engineering. ISSN: 2277-128X. Volume 3, Issue 4, April 2013.
- [19] AashnaBisht, “Enhanced Round Robin Algorithm for process scheduling using varying quantum precision”. IRAJ International Confrence-Proceedings of ICRIEST –AICEEMCS.29thDEC 2013.pune INDIA. ISBN:978-93-82702-50-4
- [20] A. Kousalya et al, / (IJCSIT)International Journal of Computer Science and Information Technologies. Vol. 6 (3), 2015, 2687-2690
- [21] R. Kettimuthu, V. Subramani, S. Srinivasan, T.B. Gopalsamy and P. Sadayappan.Selective preemption Stategies for parallel job scheduling. To appear in Proceedings of International Conference on Parallel Processing,2002.
- [22] Alam, M., Khan, A., & Varshney, A. K., (In Press). A Review of Dynamic Scheduling Algorithms for Homogeneous and Heterogeneous System. In proceedings of the Springer, International conference of Computer Society of India (CSI) of Transactions on ICT.
- [23] Singh, K., Alam, M., & Sharma, S. K. (2015). A Survey of Static Scheduling Algorithm for Distributed Computing System. Interconnection Network. International Journal of Computer Applications 129 (2), pp.25 – 30.
- [24] Alam, M., & Varshney, A. K. (2015). A Comparative Study of Interconnection Network. International Journal of Computer Applications 127 (4), pp.37 – 43.
- [25] Mahfooz Alam and Ankur K. Varshney “A New Approach of Dynamic Load Balancing Algorithm for Homogeneous Multiprocessor System”, International Journal of Applied Evolutionary Computation (IJAE) 2015. (Accepted).