COMPARATIVE STUDY OF PROCESSES SCHEDULING ALGORITHMS USING SIMULATOR

Article · May 2013

CITATIONS
2

READS
589

2 authors, including:

Vikram Singh
Chaudhary Devi Lal University
86 PUBLICATIONS **68** CITATIONS

SEE PROFILE

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

COMPARATIVE STUDY OF PROCESSES SCHEDULING ALGORITHMS USING SIMULATOR

[1]VIKRAM SINGH, [2] TIRLOK GABBA

[1]Professor, CDLU, Sirsa, Haryana, India

[2] A.P, Department of M.Tech (CSE), BITS, Bhiwani, Haryana, India

ABSTRACT

This paper present a simulating the behavior of CPU scheduling algorithms using exponential distribution function. It is very hard to analyze the waiting time when the number of processes are large .when the processes is loaded in the ready queue, then it passes to CPU for execution. The selection of the processes from the ready queue is done through the various CPU scheduling algorithms. In this paper, simulation is done through random no generator using exponential distribution function for arrival and burst time of processes. After that, comparing various algorithms to know which one has the least average waiting time of processes. And the implementation is done through Microsoft Visual Studio 6.0. Each algorithms policy is simulated on the same reference string during a simulation run and then results are analyzed. The best policy is the one that takes less waiting time.

Keyword: CPU scheduling, Simulation, waiting time, ready queue.

1 INTRODUCTION

The main objective of multiprogramming system is to load many processes in the main memory where they reside in the ready queues making link lists. A process is simply program on execution. A Process is an instance of a computer program that is being executed. It contains the program code and its current activity. CPU Scheduling is the basis of multiprogrammed

operating system [1]. By switching the CPU among processes, the operating system can make the computer more productive.

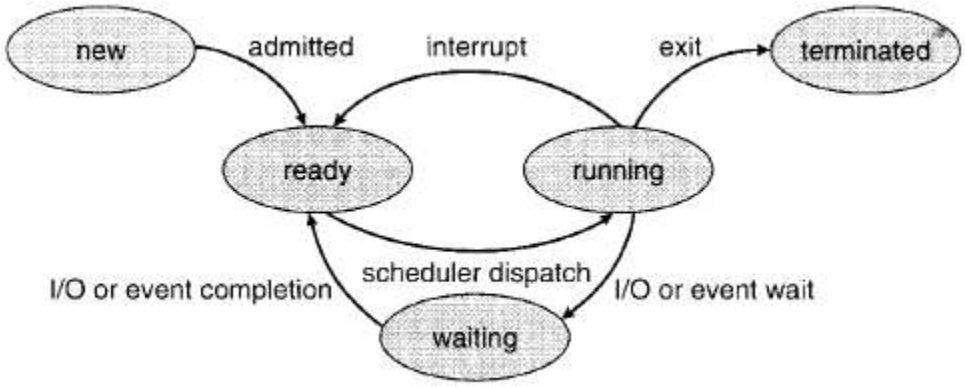

Fig.1: States diagram of Process

There are five states in which a process stay. These states are New, Ready, Run, Wait and Terminate.The Operating System helps in determining which process is allocated to CPU with the helps of CPU schedulers. The algorithms concern with CPU schedulers is known as CPU Scheduling algorithms.

CPU scheduling decisions may take place when a process:

1. Switches from running to waiting state

2. Switches from running to ready state

3. Switches from waiting to ready

4. Terminates

Scheduling under 1 and 4 is nonpreemptive. All other scheduling is preemptive.

1.1. OBJECTIVE OF PROCESS SCHEDULING:

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

The main goal of CPU scheduling algorithms is to utilize the resources effectively and efficiently. It can be accomplished by CPU busy as much as possible. And the number of processes in the job queue must be maximized. It is called the throughput**.** It is the task of operating system is to provide the fair time of CPU to the each process in the ready queue. By this, each process participates in the execution of the CPU time [4].

- CPU utilization – It keep the CPU as busy as possible. It must have maximum value.
- Response time – The amount of time it takes from when a request was submitted until the first response is produced, not output (for time-sharing environment). It must have minimum value.
- Throughput – The number of processes that complete their execution per time unit It must have maximum value.
- Turnaround time –The amount of time to execute a particular process. It must have minimum value.
- Waiting time – The amount of time a process has been waiting in the ready queue. It must have minimum value.

2. LITERATURE REVIEW:

In [2] Nazleeni Samiha Haron ,et.al. has analyzed in any distributed systems, process scheduling plays a vital role in determining the efficiency of the system. The comparative study was done based on the Average Waiting Time and Average Turnaround Time of the processes involved. This paper focuses on the development of comparative simulator for distributed process scheduling algorithms. The objectives of the works that have been carried out include the development of the comparative simulator, as well as to implement a comparative study between three distributed process scheduling algorithms; sender initiated, receiver-initiated and hybrid sender-receiver-initiated algorithms. A. Halim ZAİM has purposed different scheduling algorithms. The scheduler defined in this paper may be thought as the preprocessor in an ATM

switch, a network processor in a router or just an ordinary CPU scheduler where multiple processes with different priorities are present in the system [3].

Silberschatz A.P.B et. al. (2001) in [5] has discussed the real time scheduling .These are two types: hard real-time scheduling, soft real-time scheduling. In hard real-time scheduling, a scheduler guaranteeing the process it will complete on time. In soft real-time computing is less restrictive and receive priority over less fortune ones. Implementing both soft real time scheduling give better results. In [6], Steven Robbins has described disk head scheduling simulator that allows students to explore traditional disk scheduling algorithms as well as the consequences of modern disk technology. The simulator, which is written in Java present methods for modifying the traditional curriculum to make the presentation of disk head scheduling more relevant and interesting.

3. PROCESS SCHEDULING:

3.1 FCFS (First come First Serve): This algorithm allocates the CPU to the process that requests the CPU first. This algorithm is easily managed with a FIFO queue. New process enters the queue through the tail of the queue and leaves through the head of the queue. A process does not give up CPU until it either terminates or performs s I/O.

3.2 SJF (Shortest Job first): The SJF algorithm may be implemented as either a preemptive or non-preemptive algorithms. When the execution of a process that is currently running is interrupted in order to give the CPU to a new process with a shorter next CPU burst, it is called a preemptive SJF. On the other hand, the non-preemptive.SJF will allow the currently running process to finish its CPU burst before a new process is allocated to the CPU.

3.3 RR (Round Robin): It is often used in time sharing system. RR is similar to FCFS except that preemption is added to processes. In this algorithm, a time slice of 3 ms has been taken. After the time slice is expired, executing process will leave the CPU free and allocate the CPU to the next process in the ready queue.

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

3.4 Priority Algorithms: A priority number is associated with each process.The CPU is allocated to the process with the highest priority .the smaller number is generally used for the highest priority. It runs the highest priority algorithms first. The disadvantage of the Priority Based Scheduling is that it may cause low-priority processes to starve. Aging is the solution to this problem.

4. THE SIMULATOR:

The simulation of the various CPU algorithms is giving us a very accurate method to evaluate the scheduling algorithms. The processes arrival time and burst time is generated using the random number generator function and the exponential probability function is used with random no function. A random number string can be generated that follows a particular distribution such as exponential, poison, Beta, Gamma, Normal etc. In the present research work exponential distribution has been used to generate random numbers [9]. Most of the heuristics are based on the idea that some jobs are "compute bound" and other jobs are "I/O bound." The I/O bound jobs should be given priority for the CPU over CPU bound jobs in order to prevent "hogging" of the CPU by a compute bound job and hence poor CPU-I/O overlap [10]. After entering values for the various processes the average waiting time is calculated. And compared with each other to find the least average waiting time.

4.1 Implementation Using Visual basic 6.0:

The burst time of the processes is generated using the exponential distribution function. The implementation is carried out with VB 6.0 and the provide a graphical user interface. As user enters the numbers of processes it generates processes with bursts time and arrival time. For the priority scheduling user can enter priority of the process also using exponential distribution function [9]. The user can calculate the average waiting time of FCFS, SJF. RR, priority algorithms.

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

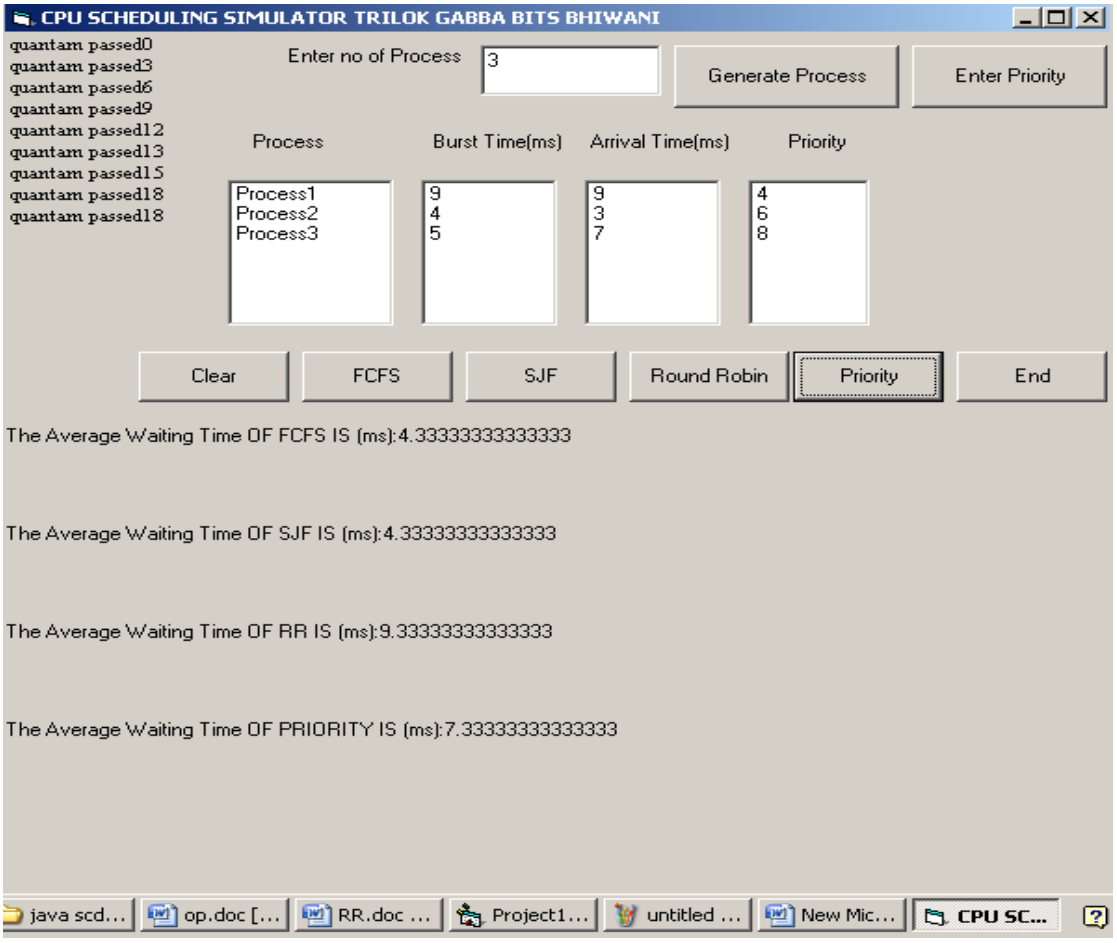

Fig2.Snapshot of implementation of CPU scheduling in VB6.0

After entering the various run for the process.calulating the average waiting time of the different algoritms. The result have been evaluated by the grapical mode of the simulator.The SJF CPU scheduling algorithms has the least avearge waiting time.

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

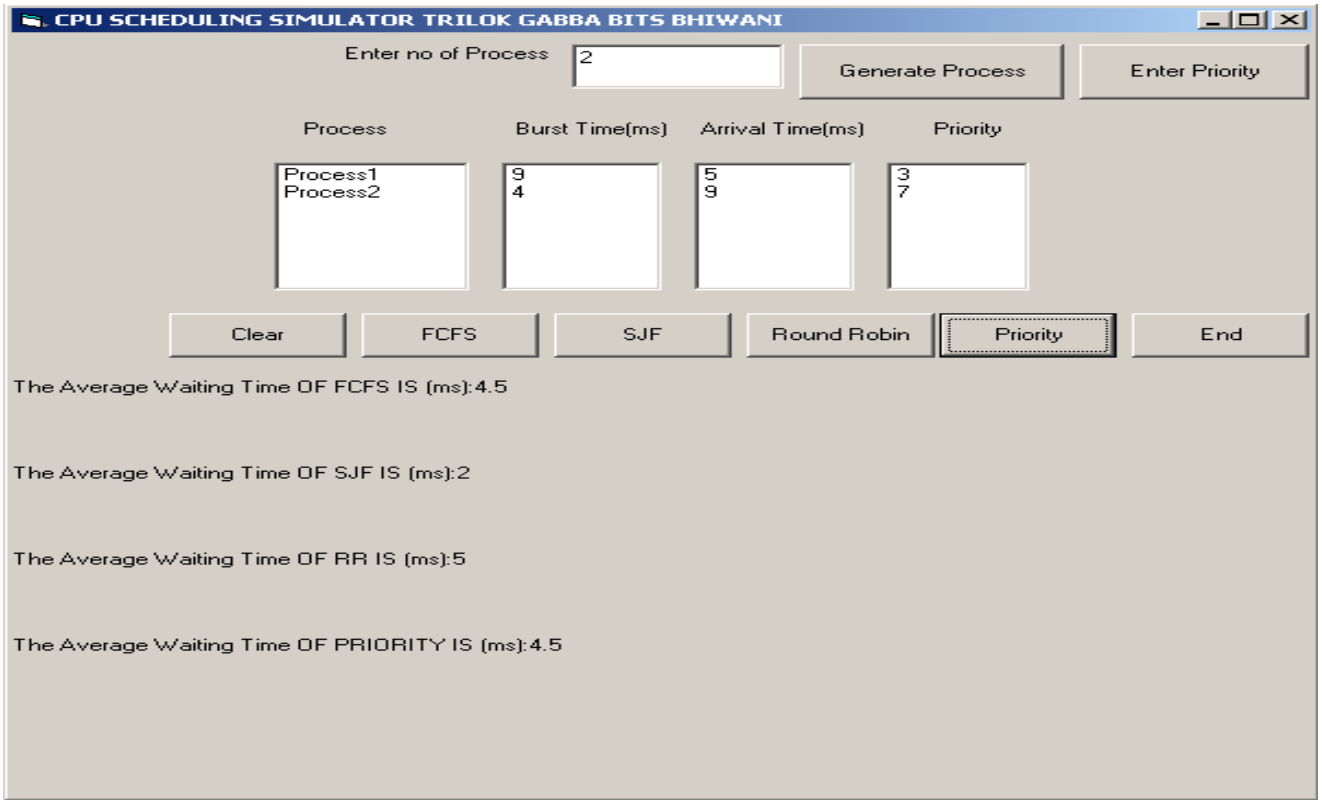

Fig3:Result of Various Scheduling Algorithms

.

5. RESULT AND CONCLUSION: Various models have been evaluated for the CPU scheduling algorithms. We have used here simulation

exponential probability distribution function. The simulator used is for calculating the average waiting time because waiting time is the criteria for analyzing the CPU algorithms. After comparing various algorithms' waiting time, SJF (shortest job first) has the least waiting time and it is the best algorithms among the others.

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

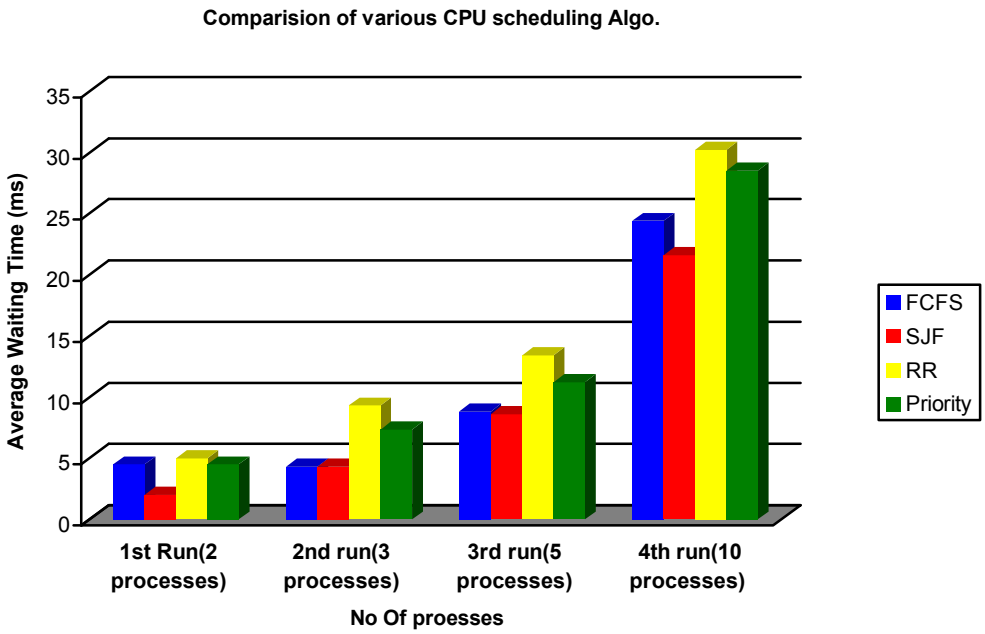

Fig4. Comparing various algorithms result graphilcally

The Figure above gives comparitive details of the various basic types of the CPU scheduling algorithms.The X axix is used to represent the number of the processes and the y axix used for the calculating average waiting time of each algorithm in millisecond.Different color are used for identify the differnet algoritms.From the various run of simulator for process,it is concluded that Shortest job First has the good criteria for selecting the process from the ready queue by the CPU scheduler.

6. REFERENECES:

[1] A. Silberschatz, B. P. Galvin & Greg Gagne, (2002). "***Operating system concepts***", sixth Edition.

International Journal of Computing and Business Research (IJCBR)

ISSN (Online) : 2229-6166

Volume 4 Issue 2 May 2013

[2] Nazleeni Samiha Haron, Anang Hudaya Muhamad Amin, Mohd Hilmi Hasan, Izzatdin Abdul Aziz, and Wirdhayu Mohd Wahid*; "**Time Comparative Simulator for Distributed Process Scheduling Algorithms**";* VOL 13, MAY 2006.

[3] Halim ZAİM**; *"Design of a Scheduler Comparison Of Different Scheduling Algorithm";*** VOL. 3, NO. 2, 2 May 2003.

[4] Stallings, W., "***Operating Systems***," (Third Edition), Prentice-Hall, 1998.

[5] Silberschatz.A., P.B. Galvin et.al, "***Operating System Concept***", 6th –edition, Prentice Hall India,

New Delhi 2001.

[6] Steven Robbins; "***A Disk Head Scheduling Simulator***"; NFS, ACM, 2004.

[7] S. Suranauwarat, "***A CPU Scheduling Algorithm Simulator***", 37th ASEE/IEEE Frontiers in Education Conference.

[8] Maria Abur, Aminu Mohammed, Sani Danjuma and Saleh Abdullahi,"***A Critical Simulation of CPU Scheduling Algorithm using Exponential Distribution***", Vol. 8, Issue 6, No 2, November 2011.

[9]R. S. Murray, J.J. Schiller, R. A Srinivasan, "**Probability *and Statistics***", third edition2009.

[10] B. Shirazi, M. Wang and G. Pathak, ***"Analysis and Evaluation of Heuristic Methods for Static Scheduling,"*** Journal of Parallel and Distributed Computing, no. 10, pp. 222-232, 1990.