Implementation on performance of parallel computing by introducing upgraded gang scheduling algorithm

Pooja, Dr.Raghuvinder

Research Scholar,   Department of CSA, CDLUniversity,Sirsa, pbhambhu23@gmail.com

A P, Department of CSA, CDLUniversity,Sirsa. raghuvinder.bhardwaj@gmail.com

Abstract:- *In computers, parallel processing is processing of program instructions by dividing them among multiple processors with objective of running a program in less time. In this paper we have explain to speed up processing by introducing concept of cache & ram & customization of existing algorithm to provide additional support. We have study to simulation of enhanced gang scheduling algorithm with additional parameters &   also   make comparative analysis traditional & proposed Methods.*

Keyword:- Parallel processing, Gang scheduling,

[1] Introduction

Gang scheduling is a scheduling algorithm for parallel systems that schedules related threads or processes to run simultaneously on different processors. Usually these would be threads all belonging to same process, but they may also be from different processes. For example, when processes have a producer-consumer relationship, or when they all come from same MPI program.

Fig 1 Gang scheduling

Gang scheduling is used so that if two or more threads or processes communicate with each other, they would all be ready to communicate at same time. If they were not gang-scheduled, then one could wait to send or receive a message to another while it is sleeping, & vice versa. When processors are over-subscribed & gang scheduling is not used within a group of processes or threads which communicate with each other, it can lead to situations where each communication event suffers overhead of a context switch.

[2] Multiprocessor Scheduling

In computer architecture, multithreading is ability of a central processing unit or a single core in a multi-core processor to execute multiple processes or In uniprocessor systems, shortest job first is a well-known algorithm for batch **scheduling**.