CHAPTER 1
INTRODUCTION AND MOTIVATION

1.1 Problem Statement

Power of internet to connect economical computational resources spread across world with help of high speed networks led to formation of "super virtual computer". Super virtual computer consist of various loosely coupled computational resources working hand in hand to accomplish huge jobs. This mechanism of voluntary computation has been used for mathematical, scientific, academic problems, requiring severe computation. Various industries are utilizing grid computing [I. Foster, 2001] for data processing related to e-commerce, drug discovery, economical forecasting and various other purposes.

Grid computing [I. Foster, 2000], [F. Berman, 1998] is a technology which yields this high potential super virtual computer. Grid computing helps us in utilizing combined potential of distributed resources. Potent scheduling algorithms are vital to realize the tremendous potential of distributed computational resources. Old-fashioned scheduling algorithms originally designed for parallel and distributed systems [T. Casavant, 1988] are not effective in grid computing. Unlike distributed systems which have homogeneous and dedicated resources, grid consists of loosely coupled heterogeneous computational resources. Assumptions like single scheduler control all resources, invariant computational node pool and various other assumptions which were present in distributed systems are not true for grid systems. Numerous attributes of grid computing
make it puzzling [Y. Zhu, 2003], [I. Foster, 2003], [I. Foster, 2001], to design scheduling algorithms for grid systems. These distinctive attributes are as follows:

1.1.1 Autonomous and Heterogeneous nature of nodes, participating in grid

In grid complete control of computational resources does not lie with grid scheduler. Moreover grid scheduler has to respect local policies of nodes spread across various administrative domains. This autonomy also gives a new dimension to heterogeneity. Priority settings of different nodes can be different for different applications. Hence grid scheduler must be flexible enough to handle dissimilar local policies [R Buyya, 2005]

1.1.2 Performance fluctuation

Ability of grid resources changes dynamically. This dynamicity in performance is caused by site autonomy and race among applications for grid nodes. Also bandwidth can be severely affected by Internet traffic. Such competition yields performance fluctuation for grid application. This performance fluctuation was not present in distributed systems. Scheduling algorithm for grid system must have mechanism to cope with this dynamic behavior.

1.1.3 Choosing Resource

Grid resources chosen for executing application are shortlisted by grid scheduler on basis of existing status of resources and selected performance models. Additionally communication cost of sending application data from one resource to another depend upon network bandwidth. Therefore grid node which has inferior computational capabilities can yield results faster if it has less communication cost associated with it. Therefore shortlisting a resource is also a tricky task and efficient scheduling algorithm for grid should possess mechanism to choose suitable most resource for application.
These issues make designing scheduling algorithms for grid computing very interesting and thought provoking task. Traditional scheduling algorithms designed for distributed systems can be modified to adapt to requirements of grid systems.

Recently researchers have designed scheduling algorithms for grid computing. Though, most of these scheduling algorithms are based of central or meta scheduler. However this central scheduler causes scalability issues and act as single point of failure. Therefore new efficient decentralized scheduling algorithms were required to solve above mentioned problems. This thesis explores decentralized scheduling algorithms and various issues related to it.

1.2 Contributions

This thesis explores the various techniques used to increase efficiency of decentralized grid Scheduling algorithm. Peer to peer (P2P) grid systems is suitable example for decentralized grid. First technique explored is recent P2P grid resources management schemes which discover resources before task are generated.

The contribution towards this research work is published and is as follows: Piyush Chauhan and Nitin, Resource based optimized decentralized grid scheduling algorithm, Advances in Computer Science, Engineering & Applications Advances in Intelligent and Soft Computing, Volume 167, (2012), pp.1051-1060.

These P2P grid resources management schemes have potential to include the step of organizing shortlisted resources in non-increasing order on basis of optimization criteria. This proactive step shortens time to find schedule when DAG (directed acyclic graph) based task is generated on grid node. There are issues which were not addressed in this algorithm. These issues are as follows:
1. Computation cost of task was not considered while scheduling.
2. Communication cost associated with task was also not considered.
3. Fault tolerance mechanism didn’t exist in above stated algorithm.

In order to overcome these shortcomings of scheduling algorithm proposed in chapter 3, author has proposed a new decentralized scheduling algorithm to schedule independent tasks among P2P grid nodes. In this method, an independent task sovereignly selects a most suitable grid node based on local information of immediate neighbors.

The contributions towards this research work are published and are as follows:


In decentralized computation and communication intensive task scheduling algorithm for P2P grid, an independent task sovereignly selects a most suitable grid node based on local information of immediate neighbors. A vital feature of this method is that it can schedule both computation intensive and communication intensive tasks to make grid system’s workload balanced.

In chapter 4 author has proposed a fault tolerant decentralized scheduling algorithm for P2P grid, which reschedules jobs of grid resource when node failure happens. This mechanism handles node failure in fully decentralized manner. Key feature of this fault tolerant approach is that it reschedules jobs depending upon communication and computation cost associated with jobs.
However fault tolerant decentralized scheduling algorithm for P2P grid given in chapter 4 are effective for scheduling independent tasks only. Therefore existing genetic algorithms were modified by author in chapter 5 to schedule interdependent tasks on P2P grid resources. The contribution towards this research work is published and is as follows:


Subtasks present at each level are scheduled in parallel among P2P grid resources using genetic algorithm. Solution obtained using genetic algorithm yield good results. Its fault tolerant mechanism is also effective and efficient. Another research article where fault tolerant mechanism of author was used is published and is as follows:


Further author modify work in chapter 4, to schedule DAG based task on P2P grid. Contribution towards this research work is published and is as follows:


Schedule obtained using this algorithm is better than one obtained using genetic algorithm.
Hence decentralized scheduling algorithms proposed by author yield efficient schedules as compared to existing decentralized scheduling algorithms; also independent fault tolerance mechanism proposed is helpful and gives results faster. Scheduling decision for DAG based task are taken in algorithms proposed in this thesis are efficient because they schedule on basis of computation and communication cost associated with them and present states of shortlisted P2P grid nodes.

1.3 Outlines of thesis

This thesis explores scheduling algorithms for decentralized grid in systematic manner. Firstly background and preliminary is given in chapter 2, it explores scheduling basics and existing algorithms. Chapter 3 gives details of resource based optimized decentralized grid scheduling algorithm. In chapter 4, author explained in detail decentralized computation and communication intensive task scheduling algorithm for independent tasks on P2P grid and fault tolerance mechanism for the same. Precedence level based genetic scheduling algorithm to schedule subtasks of DAG based task on P2P grid and fault tolerance mechanism for the same is mentioned in chapter 5. Next in chapter 6, decentralized scheduling algorithm for DAG [E. Gansner, 1988] based task on P2P grid is explained. Chapter 7 concludes the thesis.