Chapter Seven

CONCLUSION
Chapter 7
Conclusion

7.1 Overview

Computational grids will provide a platform for new generation of applications. Grid applications will include sequential, parallel and distributed programs that can be executed at a number of grid resources simultaneously. An effective scheduling algorithm alone can exploit the true potential of the grid resources. The design of scheduling algorithm is a challenging issue for mapping the tasks to the resources, which are heterogeneous in nature. An application consists of more number of tasks which can be scheduled to different computational resources in the grid environment.

Since the scheduling problem is NP-Complete in general, solutions are found using heuristics. However, most of the early heuristics are based on simplifying assumptions about the structure of the parallel program and the environment. Common simplifying assumptions include uniform task execution times, zero inter-task communication times, contention-free communications, etc. But these assumptions are not result oriented in practical situations and restrict the applicability of the heuristics.

A scheduling algorithm requires addressing with a number of issues to exploit parallelism by identifying the task graph structure, viz., task granularity, load balancing and arbitrary task weights. The highly complicated scheduling algorithms may give good performance but it limits the scalability. A scheduling algorithm should be faster in practical situation. These conflicting requirements make the design of an efficient scheduling more complex and challenging.
In the present study, the newly developed scheduling algorithms meet the conflicting goals of high performance and low time complexity. The four essential issues in the design of efficient algorithms, viz., performance, time complexity, scalability and applicability have been addressed.

Firstly, the proposed algorithms provide good performance implying that they offer quality solutions. The algorithms are also robust in that they can tackle the problem under different input parameters, namely, the number of resources in the grid system, different sets of input tasks, etc. After conducting exhaustive experiments, it is found that the performance is better than the contemporary heuristics.

Secondly, the time complexity of the algorithm is a prime factor provided the quality of solution is not compromised. The newly developed LCTSA algorithm is used for efficient scheduling.

Thirdly, the applicability of the algorithm has been achieved by considering realistic assumptions about the program and grid environment. Moreover, the proposed algorithms have been evaluated using the regular task graphs representing the parallel programs such as Gauss Elimination Method and Laplace equation method.

Finally, the proposed algorithms are consistent even when presented with large number of tasks.

In nutshell, following are the investigations in the present study with DAG/workflow model of task graphs in a grid environment.

1. Low Complexity Task Scheduling Algorithm.
2. Efficient Dual Objective Scheduling Algorithm.
3. Task Duplication Based Efficient Multi Objective Scheduling algorithm for regular and irregular graphs.

5. The scheduling tool.

7.2 Low Complexity Task Scheduling Algorithm

Grid environment consists of more number of resources. These resources are used by processing the application programs. The efficient scheduling requires appropriate rank function to prioritize the tasks. Since, in the task scheduling the completion time is based on the scheduling order of the tasks. This algorithm has been developed using simple rank function based on nodes that needs minimum computation with low time complexity compared to other list scheduling algorithms. The results envisaged consistent even for large number of tasks.

7.3 Efficient Dual Objective Scheduling Algorithm

This algorithm has been proposed for a new rank function to fulfill dual objective functions. The new rank function requires the nodes, current level and its predecessor level in the DAG. By using this new rank function, this algorithm is found to be better in experimental results. Which lead to minimize the makespan while maximizing the resource utilization.

7.4 Task Duplication Based Efficient Multi Objective Scheduling Algorithm

In this algorithm the duplication mechanism is effectively utilized to reduce the communication cost/time between the predecessor and successor tasks. When the candidate tasks and its predecessor are scheduled on the same resource, the communication time becomes zero. Hence, the idle time slots of each resource are determined for implementing the duplication mechanism. In this algorithm, three
objectives are achieved, namely, reducing the makespan and communication cost while increasing the resource usages.

7.5 Grid Scheduling Strategies using Genetic Algorithm

The new task scheduling algorithm is developed by using a popular evolutionary method, Genetic Algorithm. In this algorithm, a new fitness function is defined to check the validity of a chromosome. The Roulette wheel method for selection, single point crossover and single point mutation are used as operators in the GSSGA. The experimental results are highly encouraging when compare them with list scheduling algorithm in minimizing the makespan and communication time while maximizing the resource utilization.

7.6 Scheduling Tool

All newly developed algorithms in the present study are combined together and made available for conducting experiments in a form of a scheduling tool. It has been designed effectively by incorporating Graphical User Interface to make the tool more user friendly to the researchers. The tool has been three modules, namely, Input module, Scheduling module and Display module. The Input module supplies the task graph with the required number of tasks with a specified CCR value. The Scheduling module selects the scheduling strategy and Display module produces the schedule of assignment of tasks to the resources.

7.7 Limitations of Study and Future Research Directions

The methods are mostly evaluated with the arbitrary task graphs with maximum of two hundred tasks. The present research study could be generalized when the task graphs having few thousand tasks are considered for testing. In all the
algorithms, the computational speed of the resources as the heterogeneous factor in the grid environment were considered. But the other heterogeneous factors of resources like memory contention, number of registers etc. and network related factors should also be studied.

In the present research work the focus was on Makespan, Resource utilization and Communication cost. Other performance metrics are also available in Grid such as flow time, throughput, load balancing, etc., which can be explored by further research work. The duplication mechanism is found to be suitable only for communication intensive applications. But further research can also be carried out to explore the possibility of using duplication mechanism in all type of graphs.

While applying the GA in the scheduling strategy, the initial population can be generated using duplication based list scheduling algorithms. Moreover different types of GA operators can also be applied which may give better optimal schedule.