Abstract

Efficient resource scheduling is vital to obtain the reliability and performance of task execution in grid environments. In the de-centric grid environment, the highly dynamic, heterogeneous and autonomic resources exhibit the variance of availability and performance. At the same time, grid is vulnerable to failures due to its large scale distribution and network based access. Therefore, the reliability and performance of task execution decreases in grid. In order to maintain the reliability and performance in grid, the probable resource failures and variance of their availability and performance has to be taken into consideration while taking the scheduling decisions. The issue has been extensively explored in literature and many resource scheduling strategies have been proposed. These scheduling strategies are broadly classified as proactive and post-active strategies. In proactive scheduling strategies, the probability of resource failures and variance is minimized through right resource selection at schedule time. In post-active scheduling strategies, resource failures and variance are efficiently detected and recovered at run time to minimize their impact on task execution. Although, these scheduling strategies improve the reliability and performance of task execution in grid, but are constrained by their overheads.

In order to achieve the reliability and performance of task execution in grid computing, an adaptive, post-active ‘Cooperative Resource Scheduling’ strategy is proposed in this thesis. The proactive failure handling features are added into cooperative resource scheduling to obtain ‘Trust Oriented Cooperative Resource Scheduling’ which further improve the reliability and performance. CRS strategy is further used to accomplish the computing task according to its delay tolerance by
‘Execution Backup Customization’ in grid. CRS is also used for the ‘Priority Based Task Execution’ in grid computing.

Cooperative Resource Scheduling is a post-active resource scheduling strategy in grid computing. It organizes multiple resources together as a Cooperative Computing System (CCS) to provide a cooperative task execution environment in grid. CCS is considered as a parallel system which fails only if all its participant resources fail. Sufficient execution backup is provided in CCS to ensure the completion of computing task without failure. During the failure of resources in CCS, execution of the task is reorganized dynamically on the un-failed resources to overcome the impact of failed resources. In order to maintain the service reliability and performance of CCS, its failed resources are dynamically replaced with alternative matching resources from grid. Simulation results show that Cooperative Resource Scheduling strategy is able to maintain the service reliability and performance in the presence of resource failures and their variance. Thus, it achieves better reliability and performance in comparison to the existing post-active resource scheduling strategies used in grid computing.

In Trust Oriented Cooperative Resource Scheduling strategy, resources are selected on the basis of their trust in grid to organize the CCS. A resource with high trust is considered to have low probability of failure. Therefore, the probability of failure of CCS is reduced through right resource selection during scheduling of the computing task. A trust parameter of each resource which represents its trust and reputation in grid is stored in the resource information server. Failure rate of the resources is derived from their trust parameter. Simulation results show that Trust Oriented Cooperative Resource Scheduling improves the service reliability and
performance in comparison to the Cooperative Resource Scheduling and existing Trust Oriented Resource Scheduling strategies used in grid.

Computing tasks arriving for execution in grid are of mixed types with respect to their delay tolerance which can be minimum, average or maximum. In order to accomplish the computing task according to its delay tolerance, Cooperative Computing System with Customized Backup (CCS_CB) is formulated in cooperative resource scheduling. Resource in CCS_CB are organized to provide two-fold backup, active-standby backup or primary backup computing service to the task with minimum, average or maximum delay tolerance respectively.

Finally, cooperative resource scheduling is used to allocate multiple computing tasks to a customized CCS. If multiple tasks are present for execution in parallel, execution of the tasks is accomplished in the order of their priority. The customized CCS is called Cooperative Computing System with Task Prioritization (CCS_TP). Execution priority of the computing tasks is established on the basis of their delay tolerance. A task with maximum delay tolerance has the minimum priority of execution whereas a task with minimum delay tolerance has the maximum priority of execution. CCS_TP is formulated to keep the average execution delay of the computing tasks within a specified limit using the principles of queuing theory.

The state of CCS in the proposed scheduling strategies is modeled using continuous time Markov process for a parallel system with \( n \) repairable components. Reliability and performance of task execution through the proposed resource scheduling strategies is evaluated through extensive simulation experiments under varying environmental conditions. Arrival and duration of computing tasks as well as
durations of failure/replacement of resources in CCS are represented stochastically in the proposed resource scheduling strategies. Simulation results justify the performance improvement of the proposed resource scheduling strategies over the existing scheduling strategies in grid computing. Passive task replication is used in the proposed resource scheduling strategies. The number of replicas is decided so as to minimize the overhead of task execution while optimizing the service reliability. Therefore, the overhead of task execution decreases in comparison to the existing resource scheduling strategies in grid computing. The presented research will provide the users a more reliable, collaborative and cooperative problem solving grid environment with high performance and low overhead.