Abstract of the Thesis

Self-Tuning Job Scheduling Strategies for the Resource Management of HPC Systems and Computational Grids

Achim Streit

In this thesis we develop and study self-tuning job schedulers for resource management systems. Such schedulers search for the best solution among the available scheduling alternatives in order to improve the performance of static schedulers. In two domains of real world job scheduling this concept is implemented. First of all, we study the scheduling in resource management software for high performance computing (HPC) systems. Typically, a single scheduling policy like first come first serve is used, although the characteristics of the submitted jobs permanently change. Using a single scheduling policy might induce a performance loss, as other policies might be more suitable for specific job characteristics. We develop a self-tuning scheduler, which automatically checks all implemented policies and switches to the best one. This improves the performance, in terms of increased utilization and decreased waiting time.

Secondly, we develop and study an adaptive scheduler for computational grid environments. In such grids, several geographically distributed HPC machines are joined in order to increase the amount of computational power. Grid jobs might be scheduled across multiple machines, so that the communication among the job parts involves slow wide area networks. This often induces an additional communication overhead, which has to be considered by the grid scheduler. Our adaptive grid scheduler considers the slower communication over wide area networks by extending the execution time of such multi-site jobs. The developed adaptive multi-site grid scheduler automatically checks, which of the two options is more beneficial: waiting for enough resources at a single site, or using multiple sites and the slower wide area network immediately.

In both cases we use discrete event simulations for evaluating the performance of the developed schedulers. The results for the self-tuning scheduler show, that an increased utilization of the system and a decreased waiting time for the jobs are possible. We think, that such self-tuning schedulers should be used in modern resource management systems for HPC machines. The evaluation of the grid scheduler shows, that in general a combination of many small machines and multi-site scheduling can not perform as well as a single large machine with the same amount of resource. However, the adaptive multi-site scheduler decreases the performance difference significantly. We think that the participation in computational grid environments is beneficial, as larger problems requiring more computational power can be solved.