Marco Aurelio Novaes Esteves

"Rumo a uma Heurística Rápida para a Geração de Escalonamentos Eficientes em Grids Computacionais"

This research concentrates on the problem of statically scheduling tasks of a parallel application in environments with characteristics similar to Computational Grids. This problem, which is known to be NP-Complete, deals with allocating tasks to processors (with distributed memory) efficiently in order to minimize the execution time (or makespan) of the application. Computational Grids can be seen as a cost-effective alternative to supercomputers, making high performance computation accessible to common users. Unfortunately, task scheduling in this high latency distributed environment, where the resources available are of a dynamic and heterogeneous nature, is much more complex than for traditional parallel systems. In order to minimize the complexity of a much needed dynamic scheduler, a static pre-scheduler is proposed to perform an initial allocation of tasks, leaving the dynamic scheduler to make just the adjustments necessary. In order to create efficient schedulers, either static or dynamic, some important characteristics must be considered, e.g. communication costs and the limited number of heterogeneous processors available. Furthermore, the fast schedules they produce should be generated quickly and be economical in terms of the resources required. This dissertation conducts an initial study on task scheduling specifically tuned to attend these requirements. This study focuses on a detailed investigation into scheduling algorithms based on List Scheduling, evaluating both the principal priorities in use and new strategies to reduce makespans. The result of this work is a fast and efficient scheduling algorithm that can be applied to environments with a limited number of heterogeneous processors. Although this algorithm has shown itself effective when compared with other algorithms, this is by no means an end product, rather a starting point for future investigations concerning the problem of static scheduling for Computational Grids.