Scheduling on the Top 50 Machines

Carsten Ernemann, Martin Krogmann, Joachim Lepping, and Ramin Yahyapour

Computer Engineering Institute, University Dortmund, 44221 Dortmund, Germany
{carsten.ernemann,martin.krogmann,joachim.lepping,ramin.yahyapour}@udo.edu

Abstract. The well-known TOP500 list ranks the 500 most powerful high-performance computers. However, the list lacks details about the job management and scheduling on these machines. As this statistic is interesting for researchers and system designers, this paper gives an overview and survey on scheduling relevant information for the first 50 entries in the TOP500 list.

1 Introduction

The task of scheduling computational jobs on parallel computers is subject to research for quite a long time. Despite many different approaches from theory, only a few scheduling strategies are practically in use. The actual statistics of the actual implementations are of interest to researchers, system administrators and manufacturers. The most known statistic about high-performance computers is the TOP500 list which is published every half year [2]. The list contains the 500 most powerful computers according to the LINPACK benchmark [5].

Unfortunately, the TOP500 list focuses on the benchmark result, peak performance, machine size, manufacturer and installation site. That is, there are no information about the scheduling systems that are deployed on these machines. To this end, this paper gives an survey about additional information of the top 50 machines on the TOP500 list from November 2003. The information has been collected from available web sites, publications and by querying the corresponding system administrators. The following section gives a description about the data in the list.

2 List Description

TOP500: Position in the TOP500 ranking for the November 2003 edition of the TOP500 list.
Name: Installation name from the TOP500 list.
Country and City: Location of the installation.
Year: Year of installation or last significant update.
Computer Family Model/Manufacturer: Information about the system model and the manufacturer.

D. Feitelson, L. Rudolph, and U. Schwiegelshohn (Eds.): JSSPP 2004, LNCS 3277, pp. 17–46, 2005.

Type: Type of the computer, e.g. parallel computer (MPP), vector computer, cluster.

Inst. Type: Classification of the application field of the installation (research, academic, industry).

Processors: Number of processors.

Op. System: Operating System of the machine.

Max. Mem./Total Mem.: Maximum available main memory on a single processing node/cummulative total memory.

R_{max}/R_{peak}**:** Maximal LINPACK performance achieved and the theoretical peak performance respectively (both in GFlops).

N_{max}/N_{half}**:** LINPACK problem size for achieving R_{max} and for achieving half of R_{max}.

Queues: Information about the existing queues in the job management system.

Scheduling: Information about the used job scheduling system and strategies.

Prioritization: shows whether priorities are assigned to users and/or jobs.

Backfilling: whether backfilling is used as a job scheduling strategy [4,3]

Reservations: whether processor allocations are reservable in advance.

Checkpointing: The local management supports the checkpointing of a job. A file of a checkpointed job is generated that allows a later continuation from that point. The checkpoint file may also be migratable to other resources, but this feature is not required.

Preemption: A job is preempted on a given processor allocation and later continued [1]. In this case the corresponding application is stopped but remains resident on the allocated processors and can be resumed later. This preemption is not synonymous with the preemption in a multitasking system that typically happens in the time range of milliseconds.

Gang Scheduling: A parallel job can be preempted and continued on a given processor allocation. The scheduling system assures that all tasks of a parallel jobs are active at the same time, so that no process of a job has to wait for communication with another process of the job which is not currently active. That is preemption is synchronized for all processes of a job; within a "gang" all processes are active at the same time. This strategy can be used to allow time-shared execution of several parallel applications within different gangs.

Partitions: Many systems use partitioning to split the existing number of processors into groups for special applications. For instance, dedicated partitions for interactive jobs or data-intensive applications.

Average Utilization: Information about the average utilization of the complete machine.