Introduction to the Minitrack on Partitioning and Scheduling

Ishfaq Ahmad
Dept. of Computer Science
Hong Kong Univ of Sci & Tech
Kowloon, Hong Kong
iahmad@cs.ust.hk

Sekhar Darbha
Dept. of Electrical & Comp. Engg
Rutgers Unviersity
Piscataway, NJ 08855
darbha@ece.rutgers.edu

Parallel and distributed computing has made rapid progress in the last decade and we can anticipate continued improvement in the performance of available systems in the near future. However, the development of software environments for parallel machines has lagged behind the hardware. For parallel systems to be highly successful, we need efficient software support which can automatically exploit the inherent parallelism of the application programs. Thus, a significant amount of research for developing a software environment for parallel computers is being performed.

The research efforts can be broadly classified into three categories: compilers, languages, and support tools. Some of the example support tools which are currently being developed are the debuggers, visualization tools, performance evaluation and prediction tools.

One of the most important support tools for parallel and distributed computing are the resource management tools. These tools enable the system to realize its the maximum utilization with the proper management of processors, communication channels, and I/O devices. With proper resource management the workload is allocated to the resources of the system such that its performance, in terms of the average response time, throughput, and processor utilization, is enhanced. An important aspect of efficient execution of a parallel program is a technique to decompose and assign the associated data onto the processors of the target system. Decomposition requires partitioning the main task into multiple sub-tasks. The goal of partitioning and scheduling schemes is to efficiently partition the application into several tasks and assign the indivudual tasks onto the various processors of the system. With the recent proliferation of networked computing such as using a cluster of workstations as a virtual computer, the problem of proper management of the network has also become an important research issue.

A number of high quality papers were submitted to this minitrack and unfortunately, due to time and space constraints, we are able to accept about one-third of the submissions. The seven regular papers and the four extended abstracts accepted for this minitrack reflect a range of important and innovative work in the area of partitioning and scheduling for parallel and distributed computing.

The paper by Zomaya, Clement, and Olariu proposes a dynamic scheduler that makes its decisions through reinforcement learning. The neural network base learning allows the scheduler to be adaptive to the changing state of the system.

Thornton and Andrews describe a method for reducing the memory latency in parallel programs running on a multi-threaded architecture. The paper goes on to propose a compile-time analysis technique and scheduling algorithms using the analysis results.

The paper by Arapov, Kalinov, and Lastovesky describes a language for distributed computing. The language has attractive features such as task and data parallelism, as well as static and dynamic creation of processes. The language also includes important facilities for efficient resource management.

Diderich and Gengler have studied the prob-

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE