Recent Trends in Parallel Computing

Lokendra Singh Umrao
Indian Institute of Technology (BHU), Varanasi, India

Dharmendra Prasad Mahato
Indian Institute of Technology (BHU), Varanasi, India

Ravi Shankar Singh
Indian Institute of Technology (BHU), Varanasi, India

INTRODUCTION OF PARALLEL COMPUTING

Basic Concepts in Parallel Computing

Parallel computing is an important research area with a long development history in computer science. A parallel computer is a collection of processing elements that cooperate and communicate to solve large problems rapidly. In parallel computing, a problem decomposed in multiple parts can be solved concurrently by using multiple compute resources which run on multiple processors; an overall control/coordination mechanism is applied. The multiple instructions of decomposed computational problem parts should be executed in less response time at any moment of time. The computer resources might be a single computer with many processors, or many computers connected by a network, or a combination of both, whereas in serial computing a program run on a single computer with a single Central Processing Unit (CPU) and instructions are executed one by one; at any moment of time only one instruction executes (Wilkinson & Allen, 2009). Parallel computing is parallel with respect to time and space.

Figure 1 depicts the general process of solving a problem using parallel computing. Therefore, we can observe that parallel computing consists the following phases: parallel computers (hardware platforms), parallel algorithm (theoretical basis), parallel programming (software supports), and parallel applications (large problems).

Now "Theoretical Science," "Experimental Science" and "Computational Science" has become the three major types of science to accelerate technological development and social progression (Quinn, 2002).

BACKGROUND

1936 was a key year for computer science and the development of modern computer began and was divided into two different parts: serial computing and parallel computing. Both of them were started with the development of architecture, system software, application software, and finally they reached to the problem solving environment.

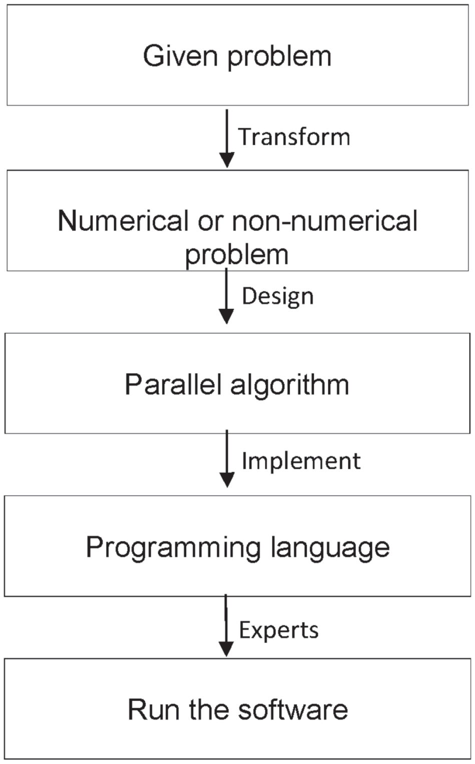
To overcome performance bottlenecks in serial computation is the main reason in the development of parallel computing. Parallelism has been applied for many years, mainly in high performance computing. As power consumption by computers has become a current issue in recent years, parallel computing has become the prevalent prototype in computer architecture in the form of multi-core processors. For the specific task, specialized parallel computers are sometime used (Chakravarty, Leshchinskiy, Jones, & Keller, 2008).

Parallel computer programs are more difficult to write than sequential one because concurrency introduces many new software bugs and race condition. Communication and synchronization between the different tasks are typically some of the greatest obstacles to getting good parallel program performance.

The well-known Amdahl's law provides the maximum possible speed-up of a program as a result of

DOI: 10.4018/978-1-4666-5888-2.ch350

Figure 1. Steps of solving a problem using parallel computing



parallelization. The speed-up of a program from parallelization is limited by how much of the program can be parallelized. For example, if 70% of the program can be parallelized, then the theoretical maximum speed-up using parallel computing would be 30x, no matter how many processors are used.

Amdahl's developed the first speedup model using fixed workload and Gustafson's proposed the scaled speedup model after reevaluating Amdahl's law. Gustafson's law is closely related to Amdahl's law. Gustafson's law is for scaled problems, where the problem size increases with the increases in machine size (i.e., number of processors). Both Amdahl's law and Gustafson's law assume that the running time of the sequential portion of the program is independent of the number of processors.

In 1993, Sun and Ni developed a memory-bounded speedup model which generalizes both Amdahl's law and Gustafson's law to maximize the use of both processor and memory capacities. The idea is to solve the maximum possible size of the problem, limited by the memory capacity. In fact, many large-scale scientific and engineering applications of parallel computers are memory-bound rather than CPU-bound and I/O bound (Hwang, 2005).

Frequency scaling has been the dominant reason for improvements in computer performance. The run-time of a program is equal to the number of instructions multiplied by the average time per instruction. Increasing the clock frequency increases the power consumption and as well as decreases the average time it takes to execute an instruction. An increase in frequency thus decreases run-time for all compute-bound programs.

Morre's law predicted in 1965 that the transistor density of semiconductor chips would double roughly every 18 months. In spite of power consumption issues, and repeated predictions of its end, Moore's law is still in effect.

In the early 1960s, parallel computer was mainly a large-size computing machine called Mainframe. IBM360 was a typical representative of this period. FORTRAN language, the main language of parallel computer programming was developed significantly. In this period, most applications of parallel computing were numerical-based and parallel algorithm researches were mainly on the classical numerical algorithms (Fast Fourior Transform).

By late 1960s, multi-functional unit with the same function began to be placed in one processor. And at the same time, pipeline technology emerged. Compared with simply increasing the clock frequency, these parallel features inside processors greatly enhanced the performance of the system. Project IlliacIV was to develop a 64 CPU's SIMD host system, involving research topics such as architecture, operating systems, hardware technology, I/O devices, programming languages, and software applications (Dongarra, Kennedy, & Torczon, 2003).

Research Methods of Parallel Computing

Parallel computing has a wide range of research fields, mainly including the hardware platform (parallel computer), parallel algorithm (parallel computing theoretical basis), software support (parallel programming), as well as the specific application of parallel computing. Due to lack of good research methods in