Parallel Computation in Weather Modeling

Skylar Thompson

November 20, 2005

Contents

1   Introduction

Since the beginning of computer science, computers have been used to solve problems in other disciplines. While computers are rarely necessary in solving problems, they can greatly increase the speed at which problems ranging from chemistry to mathematics are solved.

One such application is that of weather modeling. Weather modeling involves taking the current environmental conditions (temperature, precipitation, etc.) and producing likely conditions for a given time and place. The more fine-grained the data

is, the more precise the model, which means that the computational requirements of modeling can be among the highest of any problem.

While in the past, weather modelers typically turned to supercomputers, recently clusters of computers have been built to perform the same task as the supercomputer at a fraction of the cost. This involves tailoring the execution of the application to run on many separate computers, and the focus of this paper is on the methods and results of the parallelization of weather modeling.

2   Parallel Computation

The current trend in high-performance computation is away from single specialized computers, and more towards networked general-purpose computers. While this complicates the interactions between different parts of a single system, this cost is small compared to the benefits brought by the use of smaller, more numerous components. [5]

2.1   Paradigms

There are several ways of altering a serial algorithm into a parallel one. The most common approaches are partitioning, divide-&-conquer, and pipelining. The decision of which one to use is very specific to the problem or problem class. The wrong approach can even substantially reduce the performance of a parallel algorithm compared to its serial implementation.

Partitioning is the simplest method of attacking parallelization problem. It involves a mother node sending a chunk of information to each compute node, and each one performs the same operation on that chunk. The mother node then coaleseces all the data before terminating. [5, page 106]

Divide-and-conquer is related to partitioning, but is more complex. It is frequently used in developing a parallel algorithm for a problem that can be recursively defined, such as tree and sorting operatins. The problem is successively divided into smaller chunks, and the leaves are solved and then passed up one level. The new leaves are then solved, and the process continues until there are no more leaves to be made. [5, page 111]

Pipelining is an alternative to partitioning, and is much closer to the traditional architecture of processing units. Rather than engineer for each compute node to work on the same task as all the rest, each one specializes in one stage, and then passes on its work to the next node in the sequence. The advantage to this comes in

cache coherency on the CPUs of the nodes, because each step can be smaller relative to the all-encompassing steps that partitioning involve.

2.2 Cost Advantages over Single-image Supercomputing

One advantage is the ability to use of COTS [1] components. Standard PC hardware can be used, along with standard Ethernet networking gear. Not only does this reduce the initial cost, but it also reduces the maintenance costs by avoiding components only supplied by a specific vendor.

Parallel compute clusters tend to scale more easily as well, to a point. Because of the commodity nature of the equipment, new interconnects and nodes can easily and cheaply be added. The problem comes in when too many nodes are needed on a given problem. Because parallelization adds an inherent communication overhead, the cost of nodes talking to each other can reduce the degree to which the problem can scale.

2.3 Algorithmic Advantages of Parallelism

3 Weather Modeling

3.1 General Concepts

Weather modeling is dependent on many factors, but it comes to be implemented as a grid with each section containing data on certain characteristics. Whether manual or computerized, it is a technique that is and will continue to be useful, particularly in regions that are already vulernable to climate change. [3, page 435]

Weather models are dependent on many factors, such as atmosphere, oceans, snow cover, land, and biological presence. [3, page 436] The precision on these data can obviously vary widely, and the size of each data point can also clearly vary widely as well. As Haltiner points out, weather models are "limited by computing capability and is expected to remain so for the forseeable future", so the potential for parallelization is high. [3, page 437]

Weather models have many uses. For example, here in Wayne County, weather models are becoming increasingly important as there is a move to green energy sources such as wind and solar generation plants. [1, page 15]

---

[1]Commodity Off-The-Shelf

3.2 Parallelization

Must be fast [3, page 428], real-time [4, page 539]

A similar problem achieved a 4.11x speedup using PVM and 16 processors. [4, page 542]

4 Data

5 User interface

References

[1] Steve Blankinship. Archive-based weather modeling can aid wind site selection and design. *Power Engineering*, 109(1):15–17, January 2005.

[2] A.B. Carroll and R.T. Wetherald. Application of parallel processing to numerical weather prediction. *Journal of the Association for Computing Machinery*, 14(3):591–614, July 1967.

[3] George J. Haltiner and Roger Terry Williams. *Numerical Prediction and Dynamic Meteorology*. John Wiley & Sons, 1980.

[4] Gary Sabot, Skef Wholey, KJonas Berlin, and Paul Oppenheimer. Parallel execution of a fortran 77 weather prediction model. Technical report, Thinking Machines Corporation, 245 First Street, Cambridge, MA 02142, 1993.

[5] Barry Wilkinson and Michael Allen. *Parallel Programming*. Pearson Education, Ltd., Upper Saddle River, NJ 07458, 2005.