Large-scale simulation using parallel computing toolkit and server message block

Article *in* WSEAS Transactions on Mathematics · February 2007

Large-Scale Simulation Using Parallel Computing Toolkit and Server Message Block

PANJIT MUSIK
Science and Technology
Nakhon Si Thammarat Rajabhat University
Maung, Nakhon Si Thammarat 80280
THAILAND

KRISANADEJ JAROENSUTASINEE
School of Science
Walailak University
Thasala, Nakhon Si Thammarat 80160
THAILAND

Abstract: -

This research presents a parallel simulation technique for large-scale or complex problems. Simulation of large scale problems is almost impossible to run on a single personal computer and cannot be solved in a reasonable time. Parallelization of problems is achieved by using horizontal domain decomposition while each decomposed domain is computed on each processor of personal computers, PC Cluster. We developed a PC Cluster, namely Walailak University Cluster of 16 Slave processors Pentium 4 (WAC16P4 Cluster). A software of Parallel Computing Toolkit and the Server Message Block Protocol are applied with a new parallel computing technique, which can communicate with adjacent slave processors via the Server Message Block protocol. A WAC16P4 Cluster is an efficient parallel computing facility. This WAC16P4 Cluster is tested by running large-scale water flow simulation by Lattice Boltzmann Method to achieve faster speed. The results are comparable to a small-scale simulation of previous work and the speedup is linearly proportional to the number of processors.

Key-Words: - PC Cluster, Parallel Computing, Parallel Lattice Boltzmann Method, Parallel Computing Toolkit, Server Message Block Protocol

1 Introduction

Since the price of personal computer (PC) nowadays decreased, whereas the performance increased. Furthermore, the peripheral and spared parts are available for upgrade and maintenance. Concerning to the parallel processing concept, the PC cluster system, a group of PCs connected together with high-speed network interconnection in order to achieve faster speed in large-scale computational problems, is a cheaper alternative to provide the data processing potential as high as the high performance computer (HPC). HPC can replace any supercomputer in applications that requires super computing powers such as weather modeling, calculation and prediction, geophysics data processing, and seismic data processing. HPC is appropriate for research, education, military, and aerospace projects. It can also be used in chemical, medical, and graphics industries.

Parallel Computing Toolkit in *Mathematica* is used in this study and starts with the master processor dividing works into smaller tasks. Then the master processor distributes tasks to all connected processors known as slaves. All computing slaves work simultaneously and send the results to their master. In order to reduce communication time between nodes during large scale simulations, the Server Message Block (SMB) protocol is used. SMB can speedup the communication between slave processors by reading and writing files on each slave node. Slave nodes compute simultaneously and send the results to their master node in the last time step which is shortening the computing time in large-scale simulations.

Since a large scale simulation is a complex task, a computer with high efficiency is really required. This study aims to (1) build a WAC16P4 Cluster, (2) use this WAC16P4 Cluster to simulate a large-scale water flow simulation by Lattice Boltzmann Method and (3) compare the speedup time when varying the number of processors.

2 Parallelization

2.1 Computer resource

The WAC16P4 Cluster comprises of one master node, 2.4 GHz Intel Pentium 4 processors, 1.00 GB of RAM, one VGA card and one monitor, and 16 compute (slave) nodes. Four slave node consist of 2.4 GHz Intel Pentium 4 processors, 1.00 GB of RAM and 12 slave nodes consist of 863 MHz Intel Pentium 3 Xeon processors, 256 MB of RAM. These processor nodes are interconnected by 2 Fast Ethernet Switch 10/100/1000 mbps and 8-Station KVM Switch.

2.2 Software configuration

The operating system is Windows XP Profession on the master node and Linux Redhat 9 on slave nodes. The parallel computing software are Parallel Computing Toolkit and SMB protocol.

2.2.1 Parallel Computing Toolkit

The Parallel Computing Toolkit is a *Mathematica* application package that provides most functionalities for parallelization. Details can be found in [1].

2.2.2 Server Message Block

The SMB protocol is a protocol for sharing files, printers, serial ports, and etc. between computers. The SMB protocol can be used over the Internet on top of its TCP/IP protocol or on top of other network protocols such as Internetwork Packet Exchange and NetBEUI. Using the SMB protocol, an application (or the user of an application) can access files at a remote server as well as other resources, including printers, mail slots, and named pipes. Thus, a client application can read, create, and update files on the remote server. It can also communicate with any server program that is set up to receive an SMB client request [2].

2.3 Domain decomposition in two-dimension

The computational domain is divided into horizontal slices. Each slice is assigned to each processor shown in Fig. 1. A horizontal domain-decomposition is configured according to [3, 4] which will give the highest speedup time. The particles near the top or bottom of a processor's slice would need to interact with particles that did not exist on that processor. To facilitate the data passing between domain boundaries, an extra row of ghost points is used to store the data which are sent at the message passing boundary. The ghost points are necessary to spare the neighbor slice data for each update. This performance saves time in sending and receiving data among slave nodes.

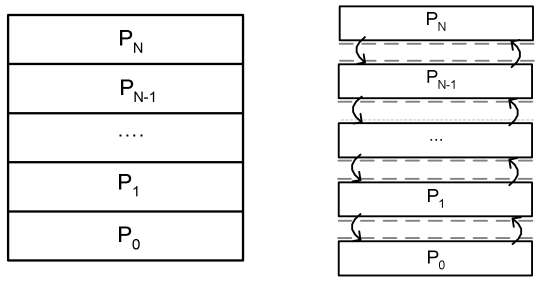


Fig. 1 Domain decomposition (a) horizontal domain decomposition, (b) Ghost point configuration for horizontal domain decomposition. Dashed lines represent extra rows of ghost point.

2.4 Parallel Computation

The parallel program is written in *Mathematica* 5 and parallel computation is done with the Parallel Computing Toolkit which can communicate between the master and slave nodes. The SMB protocol can communicate between slave processors by reading and writing files on each slave node.

The algorithm of parallel computation on a large-scale water flow simulation with *Mathematica* for windows is defined as

(1) Loading the Parallel Computing Toolkit, Parallel Commands and Parallel VirtualShared packages and starting a few remote kernels.

(2) Passive connection (16 nodes), start remote math kernel by typing this following command: math –mathlink –linkcreate –linkprotocol TCP.

(3) Checking status, the result is a list of each remote kernel's unique ID, the remote host's name, *Mathematica*'s identifier for the remote operating system, the remote kernel's process ID, and the *Mathematica* version running on the remote computer.

(4) Running a parallel water flow program as Parallel processing starts with the master node dividing works into smaller tasks and the master node distributes tasks to slave processors. The slave nodes compute simultaneously. The SMB protocol can communicate between slave processors by

reading and writing files on each slave node. So the master sends commands to the slave nodes, saves data to files in the top and the bottom of a domain on each slave node, sends the commands to the slave nodes to read data from files between adjacent slave processors and writes on the ghost point of domain on each slave nodes in each time step. The results from each slave node are sent to their master node in the last time step and combined all results at their master node.

3 Results and discussion

Testing a large-scale water flow simulation on a WAC16P4 Cluster varying from 1 node to 16 nodes, the computing time is reduced to half from 1 node to 2 nodes in all lattice sizes (Fig. 2 line a-d) and keeps decreasing as the number of processors increases (Fig. 2). As the number of lattice sizes increases from 500x500 to 2000x2000 lattice sizes, the computing time increases (Fig. 2 line a-d). The linear speedup increases as lattice sizes increase (Fig. 3). The parallel efficiency increases as the number of lattice sizes increases (Fig. 4). The parallel efficiency of 16 processors with 2000×2000 lattice size is the highest which is about 92.93 % (Fig. 4).

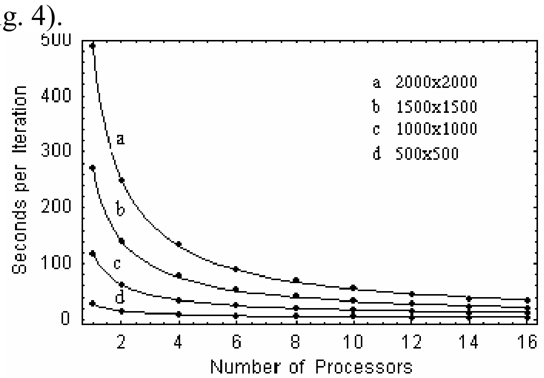


Fig. 2 Computing time of WAC16P4 Cluster and the number of processors with various lattice sizes (a-d). Dots represent computing time for one step from actual simulation and the line is the result of an exponential fitted function.

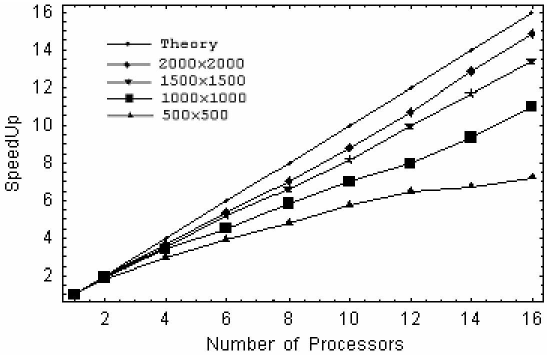


Fig. 3 Speedup (T_1/NT_N) of WAC16P4 Cluster

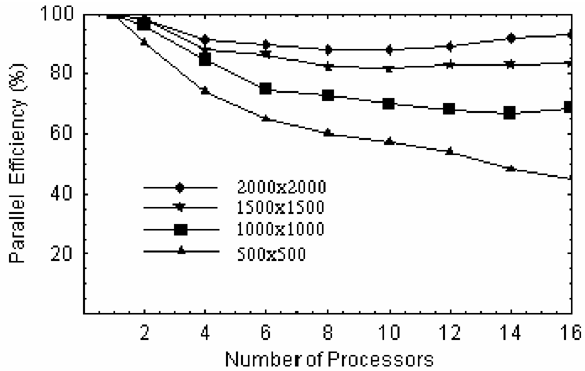
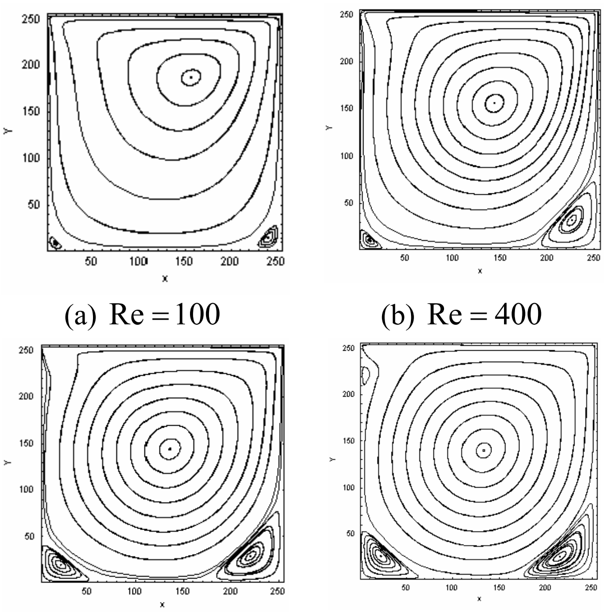


Fig. 4 Parallel efficiency (%) of WAC16P4 Cluster with various lattice sizes.

We vary 4 Reynolds numbers (Re) and generate fluid streamlines of fluid flow with 256 x 256 lattice size (Fig. 5a-d). When the system reaches its steady state, the primary vortexes move towards the center and the streamlines become more circular as the Re increases (Fig. 5a-d). The secondary vortexes are observed at the bottom left and right and become bigger when Re increases (Fig. 5a-d).



(a) Re = 100 (b) Re = 400

(c) Re = 1000 (d) Re = 2000

Fig. 5 Fluid streamlines of fluid flow in 256×256 lattice size for different Re (a-d).

We verify our large-scale water flow simulation using Lattice Boltzmann Method with previous works from Hou and Zou [4] and Guo, Shi and Wang [5] (Table. 1). The x, y coordinates of the primary vortex and the lower left and right secondary vortices from this study are in good agreement with previous works [4, 5] for all Re considered (Table. 1).

Table.1 Primary and secondary vortex centers and locations

Re	Simulation works	Primary vortex		Lower left secondary vortex		Lower right secondary vortex	
		x	y	x	y	x	y
100	a	0.6196	0.7373	0.0392	0.0353	0.9451	0.0627
	b	0.6132	0.7344	0.0371	0.0352	0.9492	0.0545
400	a	0.5547	0.6940	0.0508	0.0469	0.8867	0.1250
	b	0.5608	0.6078	0.0549	0.0510	0.8902	0.1255
	c	0.5625	0.6094	0.0508	0.0449	0.8828	0.1250
1000	a	0.5313	0.5625	0.0859	0.0781	0.8672	0.1172
	b	0.5333	0.5647	0.0902	0.0784	0.8667	0.1137
	c	0.5313	0.5644	0.0820	0.0781	0.8594	0.1133
2000	a	0.5234	0.5469	0.0898	0.1016	0.8438	0.1016
	b	0.5255	0.5490	0.0902	0.1059	0.8471	0.0980
	c	0.5234	0.5469	0.0879	0.1054	0.8417	0.0976

Note: (a) Guo, Shi and Wang [5]; (b) Hou and Zou [4]; (c) present work

4 Conclusion

Our parallelization of Lattice Boltzmann Method could be improved further for better speedup. The linear speedup for two-dimension cavity flow is independent of the computation lattice size. About 90% efficiency has been obtained with a large scale simulation on WAC16P4 Cluster. We improve the communication time between nodes by using with the Parallel Computing Toolkit and SMB protocol.

We develop a new way of communicating necessary information with adjacent slave processors via the SMB protocol. There are multiple features that enable large-scale problems to be run quickly. A WAC16P4 Cluster has an efficient parallel program for high performance computing. Future works should be focused on three-dimension cavity and other shapes to simulate flow patterns in large-scale systems.

Acknowledgements

We would like to thank Asst. Professor Dr. Mullica Jaroensutasinee and Ms. Suwimun Jutin for numerous useful discussions and proof reading the previous versions of this manuscript. We thank Mr. Piyawit Jayangkool and Mr. Pitoon Jayangkool for assisting in building a WAC16P4 Cluster Cluster. We thank Nakhon Si Thammarat Rajabhat University and the Commission of Higher Education for providing financial support and Walailak University for financial and laboratory support at the Complex System Key University Research Unit of Excellence (CXKURUE), the Institute of Research and Development.

References:

[1] http://www.mathconsult.ch

[2] http://www.reallylinux.com/docs/sambaserver.shtml

[3] Satofuka, N. and Nishioka, T. *Parallelization of lattice Boltzmann method for incompressible flow computations.* Computational Mechanics, Vol. 23, 1999, pp. 164-171.

[4] Hou, S., Zou, Q., Chen, S., Doolen, G., and Cogley, A. C. *Simulation of cavity flow by the lattice Boltzmann method.* Journal of Computational Physics, Vol. 118, 1995, pp. 329-347.

[5] Guo, Z., Shi, B., Wang, N. *Lattice BGK Model for Incompressible Navier-Stokes Equation.* Journal of Computational Physics, Vol. 165, 2000, pp. 288-306.