

Parallel Computational Fluid Dynamics Conference (ParCFD2013)

Performance analysis and optimization of PalaBos on petascale Sunway BlueLight MPP Supercomputer

Tian Min^a, Gu Weidong^a, Pan Jingshan^a, Guo Meng^{a*}^aShandong Provincial Key Laboratory of computer Network, Shandong Computer Science Center, No19 Keyuan Road, Jinan, 250101, PR China

Abstract

We present some results concerning the computational performances of the open source general purpose CFD code PalaBos, in terms of scalability and efficiency, on the petascale Sunway BlueLight MPP system. Based on the numerical simulated program of 3D cavity lid driven flow, the optimization methods in I/O, communication, memory access, etc, are applied in debugging and optimization of the parallel MPI program. Experimental results of large scalar parallel computing of 3D cavity lid driven flow show that, the parallel strategy and optimization methods are correct and efficient. The parallel implementation scheme is very useful and can shorten the computing time explicitly.

© 2013 The Authors. Published by Elsevier Ltd. Open access under [CC BY-NC-ND license](#).

Selection and peer-review under responsibility of the Hunan University and National Supercomputing Center in Changsha (NSCC)

Keywords: PalaBos, petascale computing, 3D cavity lid driven flow, parallel I/O.

Nomenclature

MPP	Massively Parallel Processor
HPC	High Performance Computing
CFD	Computational Fluid Dynamics
LBM	the lattice Boltzmann method

1. Introduction

As the importance of High Performance Computing (HPC) in Computational Fluid Dynamics (CFD) is increasing, industries are more and more interested in its applications. However, the cost of license of commercial CFD codes is proportional to the number of cores used and thus running large simulations in parallel on multi-core systems maybe economically prohibitive unless open source software like PalaBos is used. By studying computational performance requirement of industrial interest, we found that the performance capabilities of high-end computational resources have increased rapidly over recent years. In particular, the introduction of petascale systems has brought with it massive increases in the number of processing units, where it is now common to have many tens of thousands of cores available for users' codes. This development raises a number of significant challenges for the parallel performance of CFD applications.

As a software tool for classical CFD, particle-based models and complex physical interaction, PalaBos offers a powerful environment for fluid flow simulations based on the lattice Boltzmann method (LBM) [1]. Recently, new parallelization and optimization techniques have been introduced to PalaBos in order to address these challenges at several different stages of

* Corresponding author. Tel.: +86-0531-66680211-808; fax: +86-0531-66680211-803.

E-mail address: tianm@sdas.org

the calculation [2-4]. The introduction of LBM method and the hardware architecture are described in Section 2. In Section 3 we introduce the optimization methods in I/O, communication, memory access, etc. In Section 4, the impacts on scalability and performance of these new features have been analyzed on a range of prototype petascale system.

2. Method and Implementation

2.1. The lattice Boltzmann method

The lattice Boltzmann method is a numerical technique for the simulation of fluid dynamics, and in particular the numerical solution of the incompressible, time-dependent Navier-Stokes equation [5]. Its strength is however based on the ability to easily represent complex physical phenomena, ranging from multiphase flows to chemical interactions between the fluid and the borders. Indeed, the method finds its origin in a molecular description of a fluid and can directly incorporate physical terms stemming from a knowledge of the interaction between molecules. For this reason, it is an invaluable tool in fundamental research, as it keeps the cycle between the elaboration of a theory and the formulation of a corresponding numerical model short.

Compared to other CFD approaches, lattice Boltzmann might at first sight seem quite resource consuming: the discrete probability distribution functions described by the model require more memory for their storage than the hydrodynamic variables used by a classical solver of the Navier-Stokes equation (nine real valued quantities per node against three for 2D incompressible solvers).

This is however never a real issue, especially on modern computers, and it is greatly compensated by an outstanding computational efficiency [6]. Thanks to its explicit formulation and exact advection operator, the lattice Boltzmann scheme involves only a very limited amount of floating points operations per computational node. Furthermore, thanks to the locality of its algorithm, the lattice Boltzmann method is particularly well suited for computations on various parallel architectures, even on those with slow interconnection networks.

The lattice Boltzmann method is a very successful tool for modeling fluids in science and engineering [7]. Compared to traditional Navier Stokes solvers, the method allows an easy implementation of complex boundary conditions and due to the high degree of locality of the algorithm—is well suited for the implementation on parallel supercomputers.

2.2. 3D lid-driven cavity flow

The lid-driven cavity problem has long been used a test or validation case for new codes or new solution methods. Lid-driven cavity flows are not only technologically important, but also they are of great scientific interest. These flows display many kinds of fluid mechanical phenomena, including corner eddies, Taylor-Görtler-like (TGL) vortices, transition, turbulence and so on. Simple geometrical settings and easily posed boundary conditions have made cavity flows become popular test cases for computational schemes.

As a classic benchmark, the 2D lid-driven cavity flows have been extensively studied with numerical methods. However, the pioneering experimental work of Koseff & Street and coworkers in the early 1980s clearly showed that cavity flows were inherently 3D in nature. With the increase of computing capability in recent years, the 3D lid-driven cavity problems have matured as a standard Re-dependent benchmark. This problem has been solved as both a laminar flow and a turbulent flow, and many different numerical techniques have been used to compute these solutions. Since this case has been solved many times, there is a great deal of data to compare with.

2.3. Hardware architecture

Sunway BlueLight MPP Supercomputer is the first publicly announced PFLOPS supercomputer using ShenWei processors solely developed by the People's Republic of China. It ranked #2 in the 2011 China HPC Top100, #14 on the November 2011 TOP500 list, and #39 on the November 2011 Green500 List. The machine was installed at National Supercomputing Jinan Center in September 2011 and was developed by National Parallel Computer Engineering Technology Research Center and supported by Technology Department 863 project. The water-cooled 9-rack system has 8704 ShenWei SW1600 processors (For the Top100 run 8575 CPUs were used, at 975 MHz each) organized as 34 super nodes (each consisting of 256 compute nodes), 150 TB main memory, 2 PB external storage, peak performance of 1.07016 PFLOPS, sustained performance of 795.9 TFLOPS, LINPACK efficiency 74.37%, and total power consumption 1074 kW.

3. Debugging and optimization techniques

Several advances have been made to PalaBos that have improved significantly the petascale performance of the code. These optimizations have focused on the I/O, communication, memory access, etc.

3.1. Parallel I/O

The development of efficient methods for reading data from disk and writing data to disk has become increasingly important for codes that use high-end computing resources. This issue is particularly relevant for PalaBos, as the large-scale simulations that the code undertakes often require the input of huge datasets from disk and the output of large amounts of results files. For example, a dataset comprised of 1 billion cells requires around 70G Bytes of storage. Moreover, the outputs are usually required at frequent intervals in order to model a system that changes with time.

Parallel I/O strategies have been introduced to the code in order to address the I/O bottleneck associated with runs on large numbers of cores. Both serial and parallel I/O implementations are designed to use a common interface. The parallel I/O is fully implemented for reading of preprocessor and partitioned output and restart files.

Generally, the common function for file output in PalaBos is *saveBinaryBlock*, which needs to be distributed in different data collection on the compute nodes to the master node, and file operations shall be conducted by the master node.

Since the Sunway BlueLight MPP Supercomputer is equipped with a parallel file system and parallel I/O is supported, we use *parallelIO*: : *save* and *load* to save and load the data file. That is,

- In the main program, ensure *parallelIO* to be *true*: *global::IOpolicy().activateParallelIO(true)*;
- The following two functions are used for outputting and loading respectively:
parallelIO::save(lattice, "lattice"); parallelIO::load("lattice", lattice);

By doing so, file operations speed will be accelerated so greatly that more than 2 hours of work, can be reduced to 1 minute or so.

3.2. Cache Optimization

The solution of the lattice Boltzmann equation model is a long time of iteration step process, which needs thousands or even millions of steps generally. Considering the stability of the machine, it is a very important and necessary measure to “Writing” breakpoint data to ensure the continuity of iterations. Due to writing files is a costly operation in the underlying operating system, typically it is proposed to write less as far as possible or at a time write as much as possible in the operating system.

3.3. Compiler optimization

Different from compiler directives of GCC compiler, the ShenWei SW1600 processor on Sunway BlueLight MPP Supercomputer has its own compiler directives. To obtain the optimal performance, we have taken the following measures in *Makefile*:

- Set the optimization flags on: *optimize = true*
- Set the MPI_parallel mode on: *MPIparallel=true*
- Set the SW1600 compiler to use: *serialCC=swCC*
- Set the SW1600 compiler to use with MPI parallelism: *parallelCXX=mpiswCC*
- Set the optimization compiler flags: *optimFlags=-O3 -OPT:Ofast*
-

4. Performance tests and results

In this section our performance and scalability tests were performed for parallel simulation of 3D lid-driven cubic cavity flows on Sunway BlueLight MPP Supercomputer.

In the diagonally lid-driven 3D cavity flow, the top-lid is driven with constant velocity in a direction parallel to one of the two diagonals. This benchmark is challenging because of the velocity discontinuities on corner nodes.

In this benchmark test, we set the velocity in lattice units $u=0.01$, the Reynolds number $Re=1$, the lattice resolution $N=400$, the relaxation frequency $\omega=0.08$, the extent of the system $lx=1$, $ly=1$, $lz=1$; the grid spacing $\Delta x=0.0025$ and the time step $\Delta t=2.5e-05$.

In Fig.1 we show the illustration of the velocity field for u_z and u_{Norm} .

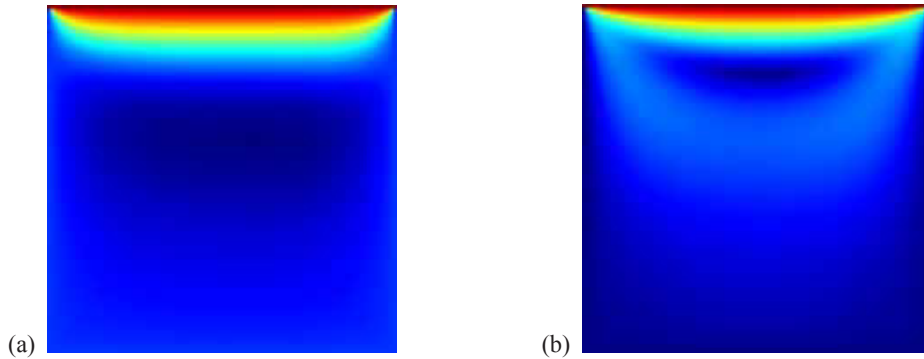


Fig. 1. Illustration of the velocity field for (a) u_z and (b) u_{Norm} .

In the following we select different number of cores on Sunway BlueLight MPP Supercomputer to compare the values of iterations and mega site updates per second, the results are shown in Table 1. The speed of execution of the application is measured in “mega site updates per second”, Msu for short.

Table 1. Comparison of Iterations and Mega site updates per second with different number of cores

Number of Cores	1024	512	256	128	64	32	16
Iterations	4764	2382	1191	596	298	149	74
Mega site updates per second	442.135	264.062	144.646	78.2313	43.5745	23.1192	12.0334

Fig.2 graphically illustrates the speedup of 3D lid-driven cubic cavity flows on Sunway BlueLight MPP Supercomputer with our optimization strategy.

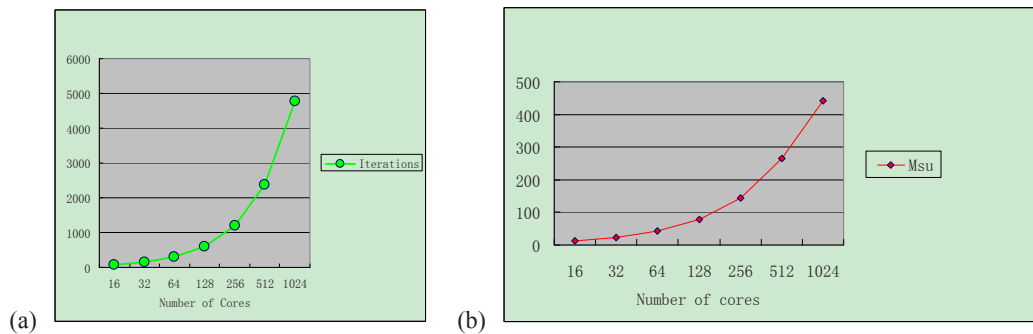


Fig. 2. Illustration of different number of cores for (a) iterations and (b) Mega site updates per second.

5. Conclusion

PalaBos is highly portable, powerful, scalable CFD software for its all mentioned models and ingredients are parallelized with MPI for shared-memory and distributed-memory platforms. Several optimization strategies in I/O, communication, memory access, etc, are supposed for debugging and optimization of the parallel MPI program in this paper. Based on the large scalar parallel numerical computing of 3D cavity lid driven flow, experimental results show that,

our parallel strategy and optimization methods have greatly shorten the computing time of the parallel implementation scheme.

Acknowledgements

This work has been financially supported by the National Key Technology Research and Development Program of the Ministry of Science and Technology of China (Grant No. 2012BAH09B03) and the National Science and Technology Major Project of Jinan, China (Grant No. 201208282).

References

- [1] B Stahl, Bastien Chopard and Jonas Latt, 2010. Measurements of wall shear stress with the lattice Boltzmann method and staircase approximation of boundaries, *Computers and Fluids* 39(9), pp. 1625-1633.
- [2] Krzysztof Kurowski, Michal Kulczewski and Mikolaj Dobski, 2011. Parallel and GPU Based Strategies for Selected CFD and Climate Modeling Models, *Information Technologies in Environmental Engineering* 3(8), pp. 735-747.
- [3] Piotr Kopta, Michal Kulczewski, Krzysztof Kurowski, Tomasz Piontek, Pawel Gepner, Mariusz Puchalski and Jacek Komasa, 2011. Parallel application benchmarks and performance evaluation of the Intel Xeon 7500 family processors, *Procedia Computer Science* 4, pp. 372-381.
- [4] Alexander Thomas White and Chuh Khiun Chong, 2011. Rotational invariance in the three-dimensional lattice Boltzmann method is dependent on the choice of lattice, *Journal of Computational Physics* 230(16), pp.6367-6378.
- [5] Daniel Lagrava, Orestis Malaspinas, Jonas Latt and Bastien Chopard, 2012. Advances in multi-domain lattice Boltzmann grid refinement, *Journal of Computational Physics* 231(14), pp. 4808-4822.
- [6] J Domitner, C Holzl, A Kharicha, M Wu, A Ludwig, M Kohler and L Ratke, 2012. 3D simulation of interdendritic flow through a Al-18wt-Cu structure captured with X-ray microtomography, *IOP Conference Series: Materials Science and Engineering* 27(1), pp. 012-016.
- [7] Orestis Malaspinas and Pierre Sagaud, 2012. Consistent subgrid scale modelling for lattice Boltzmann method, *Journal of Fluid Mechanics* 700, pp.514-542.