Parallel Natural Convection Simulation on GPU Using Lattice Boltzmann Method

Adhika Widyaparaga[a] and Pranowo[b]

[a] Department of Mechanical Engineering, Gadjah Mada University
Jl. Grafika 2, Yogyakarta 55281, Indonesia
E-mail: adhikayev@gmail.com

[b] Department of Informatics, Atma Jaya Yogyakarta University
Jl. Babarsari 43, Yogyakarta 55281, Indonesia
E-mail: pran@staff.uajy.ac.id

KEY WORDS: natural convection, LBM, Parallel, GPU

ABSTRACT: In this paper, we propose the implementation of a parallel lattice Boltzmann method (LBM) algorithm for the simulation of two-dimensional natural convection heat transfer problems. The LBM code is written using NVIDIA C language and is run on the graphical processing unit (GPU) of a Nvidia GeForce9600 card. The behaviour of the convective flows, which are driven by buoyancy forces, was numerically studied for Rayleigh numbers (Ra) from $1\times10^3$to 1×10^6. Our numerical results show a good agreement with the experimental and numerical results obtained in the literature. Based on performance comparison, it is shown that performance utilizing the processing power of the GPU is significantly faster than unaided CPU processing.

1. Introduction

Natural convective flows occur in many industrial situations. These kind of flows are particularly important in passive cooling electronic components, heat exchanger, solar energy and thermal storage systems. Such flows are very complicated. Therefore, it is necessary to clarify the flow structure and the mechanism in detail to establish an improved design technique. The numerical method has a powerful ability to investigate the flows and to clarify the predominant factors that influenced the flows.

Extensive numerical research works have been done in this field. Most of them are based on traditional numerical methods such as the finite difference, finite volume and finite element methods for solving the Navier-Stokes equations. De Vahl Davis(de Vahl Davis, 1983) provided "benchmark solutions" that were obtained with a second-order finite-difference scheme. The Navier Stokes equations are nonlinear equations and hard to be solved by traditional methods numerically, so recently the Lattice Boltzmann method (LBM) was introduced as an alternative method for solving Navier Stokes Equations especially in natural convection (Dixit and Babu, 2006; Guo, Shi, and Zheng, 2002). Lattice Boltzmann algorithms are simpler and easier to be implemented than the traditional methods. The main advantage LBM is its data parallel nature, so one can parallelize the LBM code relatively easily and accelerate the execution time.

In this paper, we propose the implementation of a parallel LBM scheme for the simulation of two-dimensional natural convection heat transfer problems on a graphical processing init (GPU) (Obrecht et al., 2011). The LBM code is written using NVIDIA C language and is run on the graphical processing unit (GPU) of a Nvidia GeForce9600 card.

2. Problem Description

The physical domain of problem is described by figure 1. The domain consists of a square enclosure with side length of 1. Constant uniform temperatures T_h and T_c $(T_h > T_c)$ are imposed at the left and right walls of the domain respectively. Buoyancy forces drive the circulation flow in the enclosure. These forces are induced by constant heating from the left side. An appropriate scaling for the flow regime is characterized by the Prandtl (Pr) and Rayleigh (Ra) number. In LBM, both of the parameters are related to density relaxation $_f$ and temperature relaxation $_g$.

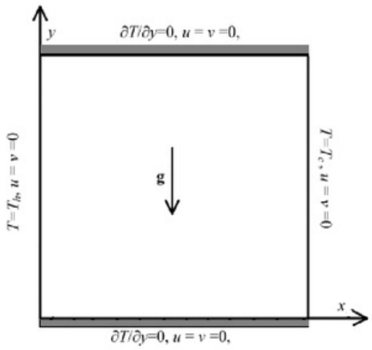

Figure1. Square enclosure and boundary conditions

3. Lattice Boltzmann Method

The Lattice Boltzmann originates from the lattice gas cellular automata method(Sukop and Thorne, 2006). It is based on discretization of the Boltzmann Equation with the Bhatnagar-Gross-Krook (BGK) relaxation for the collision operator. Through the Chapman Enskog expansion, the discrete Boltzmann Equation with the BGK relaxation can easily derive the Navier Stokes Equations.

Two kind of particle distribution functions are used for modeling the natural convective flows, i.e. density distribution functions f_i for calculation of density and velocity fields and energy distribution functions g_i for calculation of the temperature(Guo, Shi, and Zheng, 2002).

$$f_i(\mathbf{x}+e_i \quad \mathbf{x}, t+e_i \quad t) \quad f_i(\mathbf{x},t) = \quad \frac{1}{\quad_f}\left(f_i(\mathbf{x},t) \quad f_i^{eq}(\mathbf{x},t)\right)+\mathbf{F}$$

$$g_i(\mathbf{x}+e_i \quad \mathbf{x}, t+e_i \quad t) \quad g_i(\mathbf{x},t) = \quad \frac{1}{\quad_g}\left(g_i(\mathbf{x},t) \quad g_i^{eq}(\mathbf{x},t)\right) \qquad (1)$$

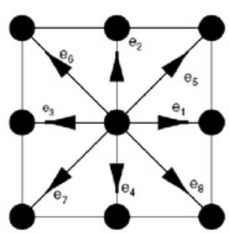

Figure 2. Discrete velocity for D2Q9

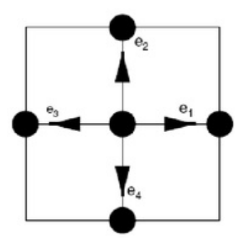

Figure 3. Discrete velocity for D2Q5 Lattice

In this paper, the D2Q9 lattice is used to represent density distribution function and D2Q5 lattice is used to represent energy distribution function. The macroscopic variables, such as density, velocity and temperature can be calculated as follow:

$$\quad = \sum_{i=0}^{8} f_i \quad ; \quad u_j = \sum_{i=0}^{8} f_i e_{ij} \; ; \quad T_j = \sum_{i=0}^{4} g_i e_{ij} \qquad (2)$$

The discretized equilibrium distribution functions are given by:

$$f_i^{eq} = \quad _{ns\,i}\left(1 + 3e_i.u + 4.5(e_i.u)^2 \quad 1.5|u|^2\right)$$
$$g_i^{eq} = T \quad _{T\,i}\left(1 + 3e_i.u_i\right) \tag{3}$$

The Boussinesq approximation is used to calculate the buoyancy force $(\mathbf{F})$

$$F_i = 3 \quad _{ns\,i}\left(e_{xi}g_{xi} + e_{yi}g_{yi}\right)\frac{(T \quad T_0)}{(T_h \quad T_c)} \tag{4}$$

where $\mathbf{g}$ is gravitational acceleration , is weight function and T_0 is averaged temperature.

The no slip boundary condition at the walls is implemented by using bounce back rule. For the top and bottom boundary nodes, the energy distribution functions are set equal to value g_i at nearest nodes, which are perpendicular to the wall. The Dirichlet boundary conditions of the left and right sides were implemented as follow:

```
for ( j=0; j<nj; j++){
     i0 = I2D(ni,0,j);
     i1 = I2D(ni,ni-1,j);
     T3[i1]=T_cold-T0[i1]-T1[i1]-T2[i1]-T4[i1];
     T1[i0]=T_hot-T0[i0]-T2[i0]-T3[i0]-T4[i0];}
```

3. GPU CUDA

The GPU was originally built for graphics rendering purposes, but now the GPU can be applied to accelerate numerical computation processing. In GPU CUDA programming, tasks of an application are grouped into instruction sets and passes onto the GPU such that each thread core works on different data but executes the same instruction(Thibault, 2009). In LBM methods, each node is addressed into a block thread and takes into account the value of the offset input. Basically the algorithm of LBM consists of two steps, namely, streaming and colliding. The parallel streaming step is implemented as follows:

```
i = blockIdx.x*TILE_I + threadIdx.x;
j = blockIdx.y*TILE_J + threadIdx.y;
i2d = i + j*pitch/sizeof(float);
f1_data[i2d] = tex2D(f1_tex, (float) (i-1)  , (float) j);
f2_data[i2d] = tex2D(f2_tex, (float) i      , (float) (j-1));
f3_data[i2d] = tex2D(f3_tex, (float) (i+1)  , (float) j);
f4_data[i2d] = tex2D(f4_tex, (float) i      , (float) (j+1));
 ...
```

The implementation of the parallel of the colliding step is:

```
i = blockIdx.x*TILE_I + threadIdx.x;
j = blockIdx.y*TILE_J + threadIdx.y;
i2d = i + j*pitch/sizeof(float);
  if (!solid_data[i2d]==0) {

// Read all f's and store in registers
f0now = f0_data[i2d];
f1now = f1_data[i2d];
f2now = f2_data[i2d];
...

// Macroscopic flow props:
ro =  f0now + f1now + f2now + f3now + f4now + f5now + f6now + f7now + f8now;
vx = (f1now - f3now + f5now - f6now - f7now + f8now)/ro;
vy = (f2now - f4now + f5now + f6now - f7now - f8now)/ro;

 float Ti2d = T0_data[i2d] + T1_data[i2d] + T2_data[i2d] + T3_data[i2d] + T4_data[i2d];
 T_data[i2d]= Ti2d;

// Calculate equilibrium f's
```

```
v_sq_term = 1.5f*(vx*vx + vy*vy);
f0eq = ro * faceq1 * (1.f - v_sq_term);
f1eq = ro * faceq2 * (1.f + 3.f*vx + 4.5f*vx*vx - v_sq_term);
f2eq = ro * faceq2 * (1.f + 3.f*vy + 4.5f*vy*vy - v_sq_term);
...
     force0 = 3.0*ro*faceq1*(Ti2d-T_mid)*( 0.0*gr)/(T_hot-T_cold);
     force1 = 3.0*ro*faceq2*(Ti2d-T_mid)*( 0.0*gr)/(T_hot-T_cold);;
     force2 = 3.0*ro*faceq2*(Ti2d-T_mid)*( 1.0*gr)/(T_hot-T_cold);;
     ...
// Do collisions
f0_data[i2d] = rtau1 * f0now + rtau * f0eq + force0;
f1_data[i2d] = rtau1 * f1now + rtau * f1eq + force1;
f2_data[i2d] = rtau1 * f2now + rtau * f2eq + force2;
...
     T0eq = Ti2d * Tfaceq1 * (1.f +3.0*( 0.0*vx+0.0*vy));
     T1eq = Ti2d * Tfaceq2 * (1.f +3.0*( 1.0*vx+0.0*vy));
     ....
     // Simulate collisions by "relaxing" toward the local equilibrium
     T0_data[i2d] = rtauT1 * T0_data[i2d] + rtauT * T0eq;
     T1_data[i2d] = rtauT1 * T1_data[i2d] + rtauT * T1eq;
     ...
```

5. Results and Discussions

The LBM source codes are based on Graham Pullan's LBM codes(Pullan, 2012). The original codes were developed for hydrodynamical simulations. Modifications were done by adding energy distribution functions and incorporating handling of the temperature boundary conditions. The implementation of numerical calculations was done by using the Visual C++ on Intel I5 processors. The Rayleigh (Ra) numbers were varied from $1x10^3$ to $1x10^6$ and constant Prandtl number Pr= 0.717. We used 3 types of grids for all the ranges of Ra numbers. The grids consisted of 320×320, 640×640, 960×960 nodes.

Fig. 4a. Isotherm $Ra=10^3$

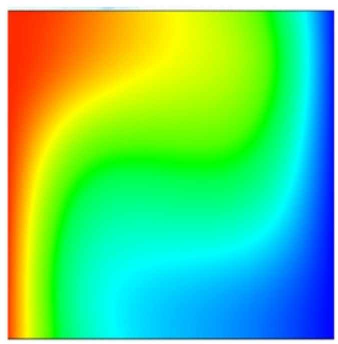

Fig. 4b. Isotherm $Ra=10^4$

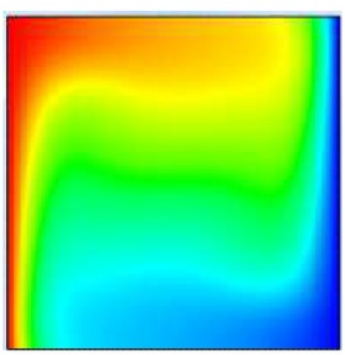

Fig. 4c. Isotherm $Ra=10^5$

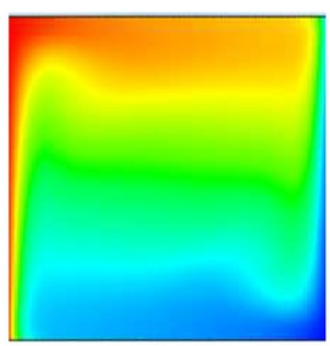

Fig. 4d. Isotherm $Ra=10^6$

The calculation results of the isotherms at steady states for different Ra are shown in Fig. 4a-4d. At low Ra number (10^3), the fluid motion driven by the buoyancy force is slow, leading to the strong diffusion. Consequently, the corresponding isothermals exhibit rather slight difference as compared to those of pure heat conduction between the left and right sides. As the Ra number is increased, the buoyancy force accelerates the circulation of fluid flow and the heat transfer of the natural convection, which is described by Nusselt Number (Nu), is significantly enhanced (figure 5) .

Figure 5 shows the comparison of the present Nusselt number calculation with the literature results (Dixit and Babu, 2006; de Vahl Davis, 1983). The experiment data was taken from the empirical formulation of Martynenko and Khramtsov(Martynenko and Khramtsov, 2005). The present results agree well with other numerical results, but slighty different when compared with experiment results for high Ra number.

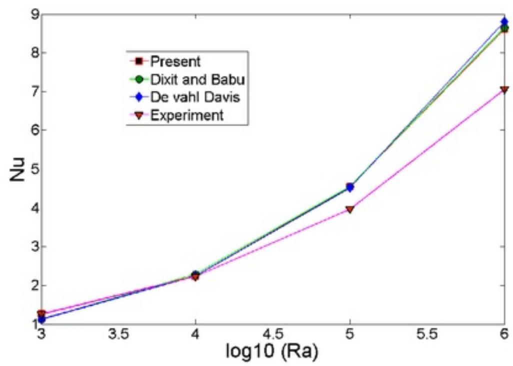

Fig.5. Comparison of the Nusselt Number

Grid independence of the present numerical results have been established for two different Rayleigh numbers, i.e.10^4 and 10^5. Table 1 gives the Nusselt numbers on three grids. The maximum change of the Nusselt number is 3.2 % for Ra=10^4 and 0.7 % for Ra=10^5. Those calculations show that present numerical results are grid independent.

Table 1. Grid independence study

Mesh	320×320	640×640	960×960
Nu (Ra=10^4)	2.241	2.244	2.158
Nu (Ra=10^4)	4.535	4.503	4.533

The GPU performances are compared with CPU performance Ra 10^5 on the three types of grids. For obtaining the steady state, all simulations were run for 30000 time steps. The GPU code performs 1.67, 6.83 and 7.53 times faster than CPU as shown in Table 2. The accelerated processing is more apparent with larger grid sizes as the intensity of arithmetic calculation and data communications in the memory of the GPU also increase, therefore aiding the calculation more and thus reducing the processing time more significantly.

Table 2. Comparison of performance for Ra=10^5

Grids	320×320	640×640	960×960
CPU Execution Time (s)	836.80	5129.49	11005.98
GPU Execution Time (s)	500.32	751.03	1461.86
Speed up	1.67	6.83	7.53

6. Conclusions

We have presented the Lattice Boltzmann method for solving natural convection heat transfer on GPU Nvidia GeForce 9600 card. It has been shown that the numerical results of the LBM have a good agreement with the numerical and experimental results reported in the previous studies. The numerical computation on GPU is 1.67 – 7.53 faster than on CPU and the increase in processing speed is dependant the size of the grids.

7. References

de Vahl Davis, G. 1983. Natural convection of air in a square cavity: a bench mark numerical solution. *International Journal for Numerical Methods in Fluids*, 3 (3),pp.249-264.

Dixit, H.N., and V. Babu. 2006. Simulation of High Rayleigh number natural convection in a square cavity using lattice Boltzmann method. *International Journal of Heat and Mass Transfer* 49,pp.727-739.

Guo, Z., B. Shi, and C. Zheng. 2002. A coupled lattice BGK model for the Boussinesq equations. *International Journal for Numerical Methods in Fluids*, 39 (4),pp.325-342.

Martynenko, O.G., and P.P. Khramtsov. 2005. *Free-convective heat transfer: with many photographs of flows and heat exchange*: Springer Verlag.

Obrecht, Christian, Frédéric Kuznik, Bernard Tourancheau, and Jean-Jacques Roux. 2011. A new approach to the lattice Boltzmann method for graphics processing units. *Computers & Mathematics with Applications*, 61 (12),pp.3628-3638.

Pullan, Graham. 2012. *A 2D Lattice Boltzmann flow solver demo* 2012 [cited August 16 2012]. Available from http://www.many-core.group.cam.ac.uk/projects/LBdemo.shtml

Sukop, M.C., and D.T. Thorne. 2006. *Lattice Boltzmann Modeling: An introduction for geoscientists and engineers*: Springer Verlag.

Thibault. 2009. Implementation of cartesian grid incompressible Navier-Stokes solver on multi-GPU desktop platforms using CUDA. , Boise State University, Boise, ID.

C7_04_IJJSS_2012

GRADEMARK REPORT

FINAL GRADE

/0

GENERAL COMMENTS

Instructor

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

C7_04_IJJSS_2012

ORIGINALITY REPORT

18%	11%	13%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	maidroc.fiu.edu Internet Source	1%
2	www.onepetro.org Internet Source	1%
3	Submitted to Santa Clara University Student Paper	1%
4	dblp.dagstuhl.de Internet Source	1%
5	Submitted to Indian Institute of Technology Student Paper	1%
6	repositorio.uchile.cl Internet Source	1%
7	www.jourlib.org Internet Source	1%
8	ir.uiowa.edu Internet Source	1%
9	Gottardi, Raphaël, Po-Hao Kao, Martin O. Saar, and Christian Teyssier. "Effects of permeability	1%

fields on fluid, heat, and oxygen isotope transport in extensional detachment systems : EFFECTS OF PERMEABILITY FIELDS", Geochemistry Geophysics Geosystems, 2013.
Publication

10	etd.ohiolink.edu Internet Source	1 %
11	Hideo Inaba. "Numerical simulation of natural convection of latent heat phase-change-material microcapsulate slurry packed in a horizontal rectangular enclosure heated from below and cooled from above", Heat and Mass Transfer, 01/25/2007 Publication	1 %
12	Submitted to University College London Student Paper	1 %
13	Gaboury, G.. "A numerical flow simulation based on the rate form of the equation of state", Computers and Fluids, 199311 Publication	1 %
14	Y. WANG. "LATTICE BOLTZMANN MODEL FOR SIMULATING VISCOUS COMPRESSIBLE FLOWS", International Journal of Modern Physics C, 2010 Publication	1 %
15	Submitted to University of Teesside Student Paper	1 %

16 The Jin Ai. "A Particle Swarm Optimisation for Vehicle Routing Problem with Time Windows", International Journal of Operational Research, 2009
Publication
1%

17 Adhistya Erna Permanasari. "A decision support framework for a zoonosis prediction system: case study of Salmonellosis", International Journal of Medical Engineering and Informatics, 2011
Publication
1%

18 hal.archives-ouvertes.fr
Internet Source
1%

19 Submitted to University of Huddersfield
Student Paper
1%

20 Tehrani, A., and A. Moosavi. "Investigation of particle dispersion and deposition in a channel with elliptic obstructions using lattice Boltzmann method", 2012 7th IEEE International Conference on Nano/Micro Engineered and Molecular Systems (NEMS), 2012.
Publication
<1%

21 www.digipen.edu
Internet Source
<1%

Yos S. Morsi. "Numerical Investigation of

22 Natural Convection inside Complex Enclosures", Heat Transfer Engineering, 3/1/2003 <1%
Publication

23 Blackledge, . "Appendix A Solutions to Problems", Digital Image Processing, 2005. <1%
Publication

24 Moufekkir, F., M. A. Moussaoui, A. Mezrhab, and H. Naji. "Computation of coupled double-diffusive convection–radiation including lattice Boltzmann simulation of fluid flow", Journal of Fluid Mechanics, 2013. <1%
Publication

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		