

GPU Accelerated Multiple-Relaxation-Time Lattice Boltzmann Simulation of Convective Flows in a Porous Media

M. M. Molla^{a1}, M. J. Haque^b, M. A. I. Khan^c and S. C. Saha^d

^a Department of Mathematics & Physics, North South University, Dhaka, 1229, Bangladesh.

^b Department of Electrical & Computer Engineering, North South University, Dhaka, 1229, Bangladesh.

^c School of Civil Engineering, University of Leeds, Leeds LS2 9JT, UK

^d Department of Mechanical Engineering, University of Technology Sydney, Sydney, Australia.

Abstract

A two-dimensional (2D) multiple-relaxation-time (MRT) lattice Boltzmann method (LBM) is used for porous media with the Brinkman-Forchheimer extended Darcy model to investigate the natural and mixed convection flows in a square cavity. This Brinkman-Forchheimer model is directly used through the forcing moments as source term. A Tesla K40 NVIDIA graphics card has been used for the present Graphics Process Unit (GPU) parallel computing via Compute Unified Device Architecture (CUDA) C platform. The numerical results are presented in terms of the velocity, temperature, streamlines, isotherms and local and average Nusselt numbers. For the wide range of Rayleigh numbers, ($Ra = 10^3$ to 10^{10}), Reynolds numbers, Darcy numbers and the porosities, the average Nusselt number is compared with the available results computed by finite element method (FEM) and single-relaxation-time (SRT) lattice Boltzmann method and the agreement shows quite well. [The results are also validated with the previous experimental results.](#) The simulations speed up maximum 144x using CUDA C in GPU comparing with the time of FORTRAN 90 code using a single core CPU simulation.

Keywords: GPU parallel computing; CUDA C; Porous media; Brinkman-Forchheimer model; MRT-LBM; Natural and mixed convection

¹Corresponding author (Md. Mamun Molla)

Tel: +880255668200, Ext.1519, Fax: +880255668211.

Email: mamun.molla@northsouth.edu

NOMENCLATURE

English Symbols:

A	aspect ratio
c	lattice speed ($m.s^{-1}$)
c_s	sound speed ($m.s^{-1}$)
Da	Darcy number
e_i	discrete velocity components ($m.s^{-1}$)
f	momentum distribution function
G	buoyancy term
Gr	Grashof number ($g\beta(T_h - T_c)H/\nu$)
g	acceleration due to gravity ($m.s^{-2}$)
g	thermal distribution function
H	height of the cavity (m)
K	permeability of the porous media
M	collision matrix for $D2Q9$ model
m	moment vectors
m^{ep}	equilibrium moment vectors
N	collision matrix for $D2Q5$ model
Nu	local Nusselt number
\overline{Nu}	Average Nusselt number
p	pressure (Pa)
Pr	Prandtl number (ν/α)
Re	Reynolds number (UH/ν)
Ra	Rayleigh number ($g\beta(T_h - T_c)H/\nu\alpha$)
t	time (s)
T	temperature of the fluid (K)
T_w	wall temperature (K)
T_m	mean temperature of the fluid (K)
U	bulk velocity ($m.s^{-1}$)
u_x, u_y	velocity along the horizontal and vertical directions ($m.s^{-1}$)
x, y	horizontal and vertical coordinate ($m.s^{-1}$)

Greek Symbols:

α	thermal diffusivity ($m^2.s^{-1}$)
ϵ	porosity of the media
μ	viscosity of fluid ($kg.m^{-1}s^{-1}$)
ν	kinematic viscosity ($m^2.s^{-1}$)
Ω	Collision operator
ρ	density of fluid ($kg.m^{-3}$)
τ	relaxation time (s)
θ	non-dimensional temperature

1. Introduction

In recent years, Graphics Process Units (GPUs) play a large role in using High-Performance Computing (HPC) due to the significant larger performance than the traditional Central Process Unit(CPU) based processors. A GPU has many slim processing units on a single chip and performs in parallel a very large number of operations on a correspondingly large number of operands. GPU computing is a heterogeneous simulation system, a small part is run sequentially by the CPU and the major part is run in GPU. It has already been established that the GPU parallel computing is well accepted due to the remarkable floating point arithmetic performance. Now the GPU is considered a computational accelerator processing a massively multi-threaded architecture which has been widely used for graphical and general purpose computations, such as molecular dynamics simulations and computational fluid dynamics [1, 2].

Fluid flow and convection heat transfer through porous media is an interesting and useful research area due to its application in the fields of science and engineering, such as hydrology, civil and mechanical engineering, chemical engineering and petroleum engineering, thermal management of electronic cooling and the improvement of heat transfer system, (Guo and Zhao [3]). Guo and Zhao [4] first proposed Bhatnagar-Gross-Krook (BGK) or single-relaxation-time lattice Boltzmann method for the isothermal incompressible flow in porous media. In their model, porosity parameter introduced through equilibrium distribution function and added the forcing term into the evolution equation to account for the linear and nonlinear drag forces of media using Darcy's term and {blueForchheimer's term. They successfully applied their model for the two-dimensional (2D) Poiseuille flow, Couette flow, and lid-driven cavity flow and compared their LBM results with the analytical and finite difference solutions. Latter on, Guo and Zhao [3] extended their study of the convective heat transfer using the same [single-relaxation-time lattice Boltzmann method \(SRT-LBM\)](#) and successfully simulated the temperature field for the natural convection of side-heated cavity and mixed convection of the channel flow through porous media.

Following Guo and Zhao [4], Seta et al. [5, 6] studied the Poiseuille flow and natural convection flow in a side-heated cavity filled with porous media using the SRT-LBM for the non-Darcy regime considering the Rayleigh number $10^3 \leq Ra \leq 10^6$. For the high Rayleigh number $10^3 \leq Ra \leq 10^{10}$, an extensive investigation has been done by Vishnampet et al. [7] considering the Darcy and non-Darcy regime using SRT-LBM and compared their results with available numerical and experimental results. The above-mentioned SRT-LBM has many advantages in the computational viewpoint, is intrinsic parallelism of the algorithm and the simplicity of programming over traditional finite difference, finite volume and finite element methods. It has been successfully used in various complex fluid system, such as, multiphase flow [8], suspension in fluids [9], magneto-hydrodynamics flow [10, 11], nanofluids [12] and flow through porous media with variable porosity [3]-[7]. Even though these advantages, it has some shortcomings, leads numerical instability when the dimensionless relaxation time τ is close to 0.5 [13].

It has already been established that the shortcomings in SRT-LBM are removed by using the [multiple-relaxation-time lattice Boltzmann method \(MRT-LBM\)](#). The MRT-LBM is numerically more stable, and it has more degrees of freedom than the SRT-LBM. Initially, Lallemand and Luo [14] constructed a generalized lattice Boltzmann equation in moment space rather than in discrete velocity space based on d'Humieres [15], that is now familiar as multiple-relaxation-time lattice Boltzmann method. Lallemand and Luo [14] analyzed the stability of the model and compared with the BGK LB equation model and found that the mechanism of separate relaxations for the kinetic modes leads to a model which was much more stable than the BGK LB equation model. There are many studies have been done by MRT-LBM, and from them, some are cited in the following sections.

Hua et al. [16] simulated high Reynolds number flow in a lid-driven cavity by MRT-LBM for Re ranges from 20000 to 100000 and they mentioned that was the first attempt to simulate

2D cavity flow for maximum $Re = 100000$. Using the MRT-LBM two lid-driven cavity flow had been investigated by Guo et al. [17]. Du et al. [13] proposed an incompressible MRT-LB model and simulated the lid-driven cavity flow for $Re = 100000$. They also simulated the double shear layer flow for high Reynolds number and showed that the numerical results are better than that of SRT-LBM. Computation of channel flow past a square cylinder with an upstream control, bi-partition has been done using the MRT-LBM by Moussaoui et al. [18].

Du and Liu [19] investigated the natural convection flow in a side-heated cavity using the MRT-LBM for the fluid flow and SRT-LBM for the temperature field. Computation of heat transfer and fluid flow in an obstructed channel was done by Moussaoui et al.[20]. They solved the fluid flow LB equation using the MRT technique and the energy equation for temperature by the finite difference method. Few articles are available in the literature with double MRT-LBM for velocity and thermal flow simulation. Guo et al. [21] investigated the mixed convection flow in a slender rectangular cavity with the D2Q9 model for velocity and D2Q5 model for the temperature simulation. For the pure fluid flows and heat transfer simulation have been studied in [22, 23, 24, 25, 26] using double MRT-LBM. These studies are conducted the CPU simulations.

Multi relaxation time lattice Boltzmann simulations of transitional flows in a deep 2D lid-driven cavity is investigated by [27] using GPU computing. They concluded that the GPU showed the efficient performance for larger grid size problems. Xu et al. [28] have studied the double MRT lattice Boltzmann simulation of lid-driven and side-heated cavity flows using GPU in the directive-based OpenACC platform and found that the optimized data layout gives a better performance over CPU. Ren and Chan [29, 30] studied SRT-LBM simulation of natural convection flow using the GPU in CUDA platform. They also investigated the SRT-LBM for velocity and MRT-LBM for total enthalpy simulation of PCM melting process in an enclosure using GPU computing [31].

In this paper, a multiple-relaxation-time lattice Boltzmann method is used for porous media with the Brinkman-[Forchheimer](#) extended Darcy model to investigate the natural and mixed convection flows in a side or top heated square cavity using GPU computing via CUDA C platform. A FORTRAN 90 code has also been used for comparing the CPU simulation time with the simulation by CUDA C in GPU. The Brinkman-Forchheimer model is used as a source term as forcing moments . Following [21]-[25] in double MRT-LBM, the fluid flow is simulated by D2Q9 model and the temperature field by D2Q5 model. In the natural convection simulation, the Rayleigh number is considered from $Ra = 10^3$ to 10^{10} , Darcy number $Da = 10^{-2}$ to 10^{-7} and the porosity parameter is $\epsilon = 0.4$ and 0.6 . The Reynolds numbers $Re = 400$ and 1000 and the Grashof number $Gr = 100$ are considered for the mixed convection case with the same ϵ .

2. Governing Equations in Macro Scale

The non-dimensional governing equations for the laminar natural and mixed convection incompressible 2D flows for porous media are [32]:

$$\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} = 0 \quad (1)$$

$$\frac{\partial u_x}{\partial t} + \frac{u_x}{\epsilon} \frac{\partial u_x}{\partial x} + \frac{u_y}{\epsilon} \frac{\partial u_x}{\partial y} = -\epsilon \frac{\partial p}{\partial x} + A \left(\frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_x}{\partial y^2} \right) + \mathbf{D} \quad (2)$$

$$\frac{\partial u_y}{\partial t} + \frac{1}{\epsilon} u_x \frac{\partial u_y}{\partial x} + \frac{1}{\epsilon} u_y \frac{\partial u_y}{\partial y} = -\epsilon \frac{\partial p}{\partial y} + A \left(\frac{\partial^2 u_y}{\partial x^2} + \frac{\partial^2 u_y}{\partial y^2} \right) + B\epsilon\theta + \mathbf{D} \quad (3)$$

$$\frac{\partial \theta}{\partial t} + u_y \frac{\partial \theta}{\partial x} + u_x \frac{\partial \theta}{\partial y} = C \left(\frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} \right) \quad (4)$$

where u_x and u_y are the velocities of the fluid along the x and y directions, p is the pressure, θ is the temperature of the fluid and ϵ is the porosity of the porous media. Here A , B and C varies for the natural and mixed convection case and \mathbf{D} is the body force term for the porous media that is given in the next section. In the Natural convection case: $A = Pr$, $B = RaPr$, $C = 1$ and in the Mixed convection case: $A = 1/Re$, $B = Gr/Re^2$ and $C = 1/RePr$, where Pr is the Prandtl number, Re is the Reynolds number, Ra is the Rayleigh number and Gr/Re^2 is the Richardson number respectively. For the present simulation, the corresponding lattice Boltzmann model of the above equations are described below:

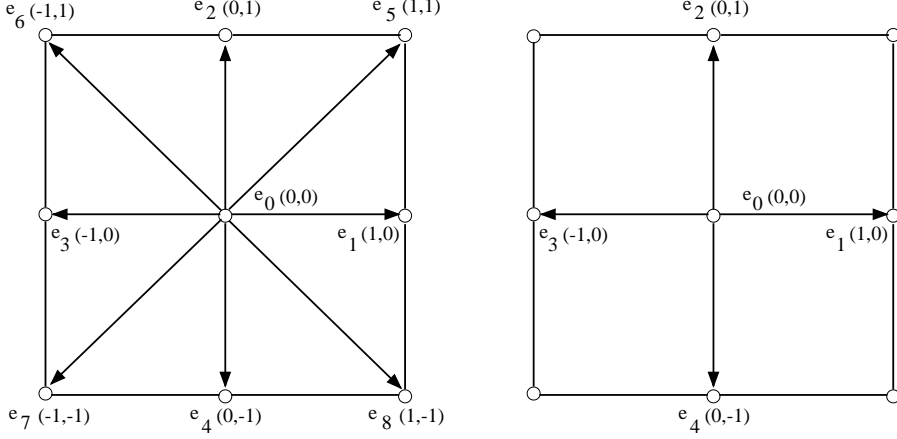


Figure 1: Lattice discrete velocity structure as in (Trouette [25]) (a) D2Q9 and (b) D2Q5 model.

3. Formulation of the problem in LBM

In this paper, it has been employed D2Q9 lattice model MRT-LBM for simulating the fluid velocity field and D2Q5 lattice model MRT-LBM (Trouette [25]) for simulating the temperature distribution, which is briefly described below:

3.1. Multiple-Relaxation-Time Lattice Boltzmann for Fluid Flow

The multiple-relaxation-time lattice Boltzmann equation for fluid flow the collision operation can be generalized as

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) - f_i(\mathbf{x}, t) = -\Omega [f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)] + \Delta t \mathbf{F}_i(\mathbf{x}, t) \quad (5)$$

where Ω is the collision operator and \mathbf{F}_i are components of the body force which will be defined later on. It is convenient to perform the collision process in the momentum space instead of velocity space. So Eq. (5) can be transformed as

$$\mathbf{f}(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) - \mathbf{f}(\mathbf{x}, t) = -\mathbf{M}^{-1} \mathbf{S} [\mathbf{m}(\mathbf{x}, t) - \mathbf{m}^{eq}(\mathbf{x}, t)] + \mathbf{M}^{-1} \left(\mathbf{I} - \frac{\mathbf{S}}{2} \right) \mathbf{F}(\mathbf{x}, t) \quad (6)$$

where $\mathbf{f} = (f_0, f_1, f_2, \dots, f_n)^T$ and \mathbf{m} and \mathbf{m}^{eq} are vectors of moments, $\mathbf{m} = (m_0, m_1, m_2, \dots, m_n)^T$ and $\mathbf{m}^{eq} = (m_0^{eq}, m_1^{eq}, m_2^{eq}, \dots, m_n^{eq})^T$ as well as the forcing components are $\mathbf{F} = (F_0, F_1, F_2, \dots, F_n)^T$.

The mapping between velocity and moment spaces can be transformed by linear transformation $\mathbf{m} = \mathbf{M}\mathbf{f}$ and $\mathbf{f} = \mathbf{M}^{-1}\mathbf{m}$. The matrix \mathbf{M} for D2Q9 is

$$\mathbf{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 \\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{bmatrix} \quad (7)$$

The collision matrix S in a moment space is a diagonal matrix. The nine eigen-values of \mathbf{S} are all between 0 and 2 so as to maintain linear stability and the separation of scales, which means that the relaxation times of non-conserved quantities are much faster than the hydrodynamic time scales. The LBGK model is the special case in which the nine relaxation times are all equal, and the collision matrix $\mathbf{S} = \frac{1}{\tau}\mathbf{I}$, where \mathbf{I} is the identity matrix. The diagonal collision matrix \mathbf{S} can be written as

$$\mathbf{S} = \mathbf{diag}[s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8] \quad (8)$$

where $s_0 = s_3 = s_5 = 1.0$, $s_1 = s_2 = 1.4$, $s_4 = s_6 = 1.2$ and $s_7 = s_8 = \frac{1}{\tau}$. Here τ is the relaxation time relates to the kinematic viscosity of fluid

$$\nu = c_s^2 \Delta t (\tau - 1/2) \quad (9)$$

where $c_s^2 = \frac{1}{3}$. The body force \mathbf{F} encompasses the viscous diffusion, the inertia due to the presence of a porous media, and an external term (Seta et al. [5]). The widely used Ergun's relation [33], the body force for the porous media can be defined as

$$\mathbf{F} = \mathbf{D} + \epsilon \mathbf{G} \quad (10)$$

$$\mathbf{F} = -\frac{\epsilon \nu}{K} \mathbf{u} - \frac{1.75}{\sqrt{150 \epsilon K}} |\mathbf{u}| \mathbf{u} + \epsilon \mathbf{G} \quad (11)$$

where $K \stackrel{\text{def}}{=} DaH^2$ is the permeability of the porous medium and $\mathbf{G} = g\beta(T - T_m)$ is the buoyancy term. Here Da is the Darcy number, H is the height of the cavity and $T_m = T_h + T_c$ is the reference temperature, T_h and T_c indicate the temperature for the heated and cold walls respectively.

Including the porosity in the equilibrium moment vector \mathbf{m}^{eq} is

$$m_0^{eq} = \rho \quad (12a)$$

$$m_1^{eq} = -2\rho + 3 \frac{(j_x^2 + j_y^2)}{\epsilon} \quad (12b)$$

$$m_2^{eq} = \rho - 3 \frac{(j_x^2 + j_y^2)}{\epsilon} \quad (12c)$$

$$m_3^{eq} = j_x \quad (12d)$$

$$m_4^{eq} = -j_x \quad (12e)$$

$$m_5^{eq} = j_y \quad (12f)$$

$$m_6^{eq} = -j_y \quad (12g)$$

$$m_7^{eq} = \frac{(j_x^2 - j_y^2)}{\epsilon} \quad (12h)$$

$$m_8^{eq} = \frac{j_x j_y}{\epsilon} \quad (12i)$$

where $j_x = \rho u_x$ and $j_y = \rho u_y$. Here u_x is the horizontal and u_y is the vertical velocity components of the fluid.

The discrete velocity components of the two dimensional (D2Q9) lattice model are: (see Figure 1(a)),

$$\mathbf{e}_i = \begin{cases} (0, 0), & i = 0 \\ c(1, 0), c(0, 1), c(-1, 0), c(0, -1), & i = 1, 2, 3, 4 \\ c(1, 1), c(-1, 1), c(-1, -1), c(1, -1), & i = 5, 6, 7, 8 \end{cases} \quad (13)$$

where, $c \stackrel{\text{def}}{=} \Delta x / \Delta t$ is the lattice speed, where $\Delta x = \Delta t = 1$.

The macroscopic fluid density ρ and velocity \mathbf{u} are obtained from the moments of the distribution function as follows:

$$\rho = \sum_{i=0}^8 f_i \quad (14)$$

$$\mathbf{u} = \frac{1}{\rho} \sum_{i=0}^8 f_i \mathbf{e}_i + \frac{\epsilon \mathbf{G}}{2} \quad (15)$$

3.2. Multiple-Relaxation-Time Thermal Lattice Boltzmann

Similarly, the multiple-relaxation-time thermal lattice Boltzmann the collision operation can be written as

$$\mathbf{g}(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) - \mathbf{g}(\mathbf{x}, t) = -\mathbf{N}^{-1} \mathbf{S} [\mathbf{m}(\mathbf{x}, t) - \mathbf{m}^{eq}(\mathbf{x}, t)] \quad (16)$$

where $\mathbf{g} = (g_0, g_1, g_2, \dots, g_n)^T$ is the thermal distribution function. Here D2Q5 model has been used for the thermal lattice Boltzmann method. The collision matrix \mathbf{N} for D2Q5 is

$$\mathbf{N} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ -4 & 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 1 & -1 \end{bmatrix} \quad (17)$$

The five eigen-values of \mathbf{S} are all between 0 and 2. The diagonal collision matrix \mathbf{S} can be written as

$$\mathbf{S} = \mathbf{diag}[s_0, s_\alpha, s_\alpha, s_e, s_\nu] \quad (18)$$

The values of s_i is given in detail in Troutte [25]. They are

$$s_0 = 1, \frac{1}{s_e} - \frac{1}{2} = \frac{1}{s_\nu} - \frac{1}{2} = \frac{1}{6} \quad (19)$$

$$\frac{1}{s_\alpha} - \frac{1}{2} = \frac{\sqrt{3}}{6} \quad (20)$$

These parameters corresponds to the thermal diffusivity

$$\alpha = \frac{\sqrt{3}(4+a)}{60} \quad (21)$$

For the D2Q5 thermal LB model, a depends on the thermal diffusivity α and is less than 1 to avoid the numerical instability.

Corresponding to the distribution function g_i , the equilibrium moments \mathbf{m}^{eq} is

$$m_0^{eq} = T \quad (22a)$$

$$m_1^{eq} = u_x T \quad (22b)$$

$$m_2^{eq} = u_y T \quad (22c)$$

$$m_3^{eq} = aT \quad (22d)$$

$$m_4^{eq} = 0 \quad (22e)$$

In this case, for the 5-bit two dimensional (D2Q5)lattice the discrete velocities are (see Figure 1(b)),

$$\mathbf{e}_i = \begin{cases} (0, 0), & i = 0 \\ c(1, 0), c(0, 1), c(-1, 0), c(0, -1), & i = 1, 2, 3, 4 \end{cases} \quad (23)$$

The temperature T can be calculated as

$$T = \sum_{i=0}^4 g_i \quad (24)$$

3.3. Application of the MRT-LBM to convective flows in a square cavity

To show the applicability of this MRT-LBM for porous media for the convective flows two cases have been considered, one is for natural convection and the other one is mixed convection flow.

(i) In the cases of natural convective flow, a 2D square cavity filled with porous media in which the top and bottom walls are adiabatic and, but the left and right walls are isothermal and maintained at different temperatures T_h and T_c , respectively. Here $T_h = 1.0$ and $T_c = 0.0$ which is shown in Figure 2(a). (ii) Figure 2(b) for the mixed convection flow, the top wall is moving along the horizontal direction and maintained at a constant temperature $T_h = 1.0$ and at the bottom wall $T_c = 0.0$. The left and right walls are adiabatic. For the present problem, the Boussinesq approximation is used in which all fluid properties are assumed to be constant, but the density whose variation with temperature is allowed through the buoyancy term.

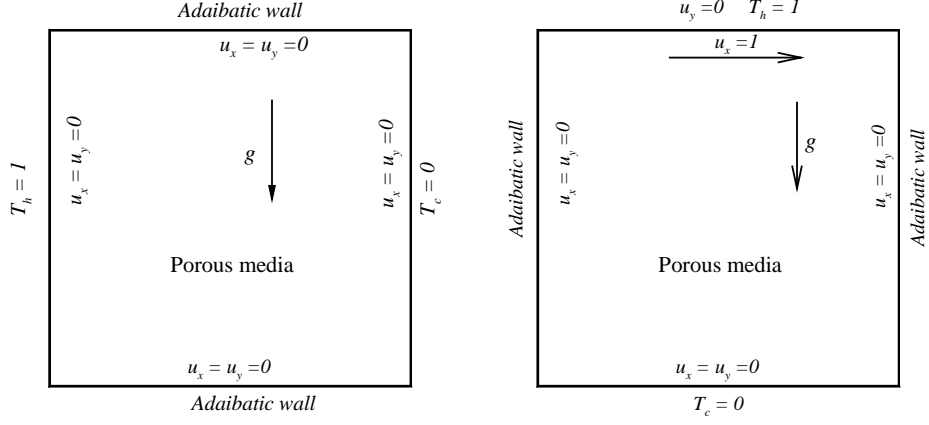


Figure 2: Schematic model for convective flows in square cavity: (a) Natural convection and (b) Mixed convection case.

3.4. Boundary Conditions

For the no-slip wall the bounce-back boundary condition is applied on velocity fields. As like Trouette [25], the incoming unknown distribution function $f_i(\mathbf{x}_f, t + \Delta t)$ is equal to the outgoing post-collision distribution function $f_i^c(\mathbf{x}_f, t)$:

$$f_i(\mathbf{x}_f, t + \Delta t) = f_i^c(\mathbf{x}_f, t) \quad (25)$$

At a wall with a fixed temperature T_w , the following boundary condition is applied for the D2Q5 model MRT thermal lattice Boltzmann simulation:

$$g_i(\mathbf{x}_f, t + \Delta t) = -g_i^c(\mathbf{x}_f, t) + 2\sqrt{3}\alpha T_w \quad (26)$$

For an adiabatic wall the anti-bounce-back condition is applied

$$g_i(\mathbf{x}_f, t + \Delta t) = g_i^c(\mathbf{x}_f, t) \quad (27)$$

3.5. Average rate of heat transfer

In the heat transfer problem, the interest in calculating the rate of heat transfer in terms of the local and average Nusselt number that are defined below respectively:

For natural convection flow:

$$Nu(y) = -\frac{\partial \theta}{\partial x} \Big|_{wall} \quad (28)$$

and

$$\overline{Nu} = \frac{1}{H} \int_0^H Nu(y) dy \quad (29)$$

For mixed convection flow:

$$Nu(x) = -\frac{\partial \theta}{\partial y} \Big|_{wall} \quad (30)$$

and

$$\overline{Nu} = \frac{1}{L} \int_0^L Nu(x) dx \quad (31)$$

where H is the height and L is the length of the cavity.

4. Implementation of Forcing Term in MRT-LBM

The implementation of the forcing term (11) by adding explicitly with nine forcing moments as Guo and Shu [34]:

$$F_0 = 0 \quad (32a)$$

$$F_1 = 6\rho \left(1 - \frac{s_1}{2}\right) (u_x F_x + u_y F_y) / \epsilon \quad (32b)$$

$$F_2 = -6\rho \left(1 - \frac{s_2}{2}\right) (u_x F_x + u_y F_y) / \epsilon \quad (32c)$$

$$F_3 = F_x \quad (32d)$$

$$F_4 = -\rho \left(1 - \frac{s_4}{2}\right) F_x \quad (32e)$$

$$F_5 = F_y \quad (32f)$$

$$F_6 = -\rho \left(1 - \frac{s_6}{2}\right) F_y \quad (32g)$$

$$F_7 = \rho \left(1 - \frac{s_7}{2}\right) (u_x F_x - u_y F_y) / \epsilon \quad (32h)$$

$$F_8 = \rho \left(1 - \frac{s_8}{2}\right) (u_x F_y + u_y F_x) / \epsilon \quad (32i)$$

where $F_x = -\frac{\epsilon\nu}{K}u_x - \frac{1.75}{\sqrt{150\epsilon K}}|\mathbf{u}|u_x$ and $F_y = -\frac{\epsilon\nu}{K}u_y - \frac{1.75}{\sqrt{150\epsilon K}}|\mathbf{u}|u_y + \epsilon g\beta(T - T_m)$.

With the above forcing moments, the collision process of MRT-LBM is implemented in moment space as

$$m'_i(\mathbf{x}, t) = m_i(\mathbf{x}, t) - s_i [m_i(\mathbf{x}, t) - m^{eq}_i(\mathbf{x}, t)] + \Delta t F_i \quad (33)$$

4.1. CUDA C Programming in GPU

The GPU computing is a heterogeneous simulation system, a small part is done sequentially by the CPU and the major part is simulated by GPU parallel computing. CUDA programming model mainly depends on the concept of the kernel. A kernel is a function that is executed in concurrent threads on the GPU. In NVIDIA Tesla k40 architecture, maximum 1024 threads formed a block and blocks are grouped into execution grid (Figure 3). In CUDA, there is two programming language, one is CUDA FORTRAN and the other one is CUDA C (or C++). In this study, we used CUDA C that is a slight modification of C programming language (NVIDIA [35]). For example, in a CUDA C programming, a new keyword is introduced as: `__global__` for the device (GPU) code and this function is calling from the main code using a triple angle `<<< ... >>>`. The heterogeneous simulation system in a CUDA code can be classified into four categories (Obrecht et. al. [36]): (i) Sequential functions run by the CPU (ii) Launching functions allowing the CPU to start a kernel (ii) Kernel run by GPU and (iv) Auxiliary functions that are inlined into the kernel at compile time.

At the run-time, the execution grid's layout is specified and a grid may have up to three dimensions. The blocks of threads with in a grid must be identical and the threads are identified with respect to the grid using the two structures `threadIdx` and `blockIdx`, containing the three fields x , y and z . A block may only be executed on a single streaming multiprocessor, that yields an upper bound of the number of threads with in a block and in a Tesla k40 this is maximum 1024. In the present CUDA code, grid layout is using as follows:

Table 1: Features of the Tesla k40 GPU device and host CPU

GPU	No. of Streaming Multiprocessor (MPs):	15
	No. of CUDA cores/MP:	192
	Total CUDA cores:	2880
	Total Global Memory:	12 GB GDDR5
	GPU Max Clock rate:	745 MHz
	Memory Clock rate:	3004 Mhz
	Memory Bus Width:	384-bit
	L2 Cache Size:	1572864 bytes
	Maximum No. of threads per MP:	2048
	Maximum No. of threads per block:	1024
	Memory bandwidth up to:	288 GB/s
CPU	Processor Intel Skylake Core i7: model	6700
	Processor speed:	3.40 GHz
	CPU memory:	16 GB DDR3
	CPU memory BUS:	3200 BUS
	L3 Cache Size::	8MB

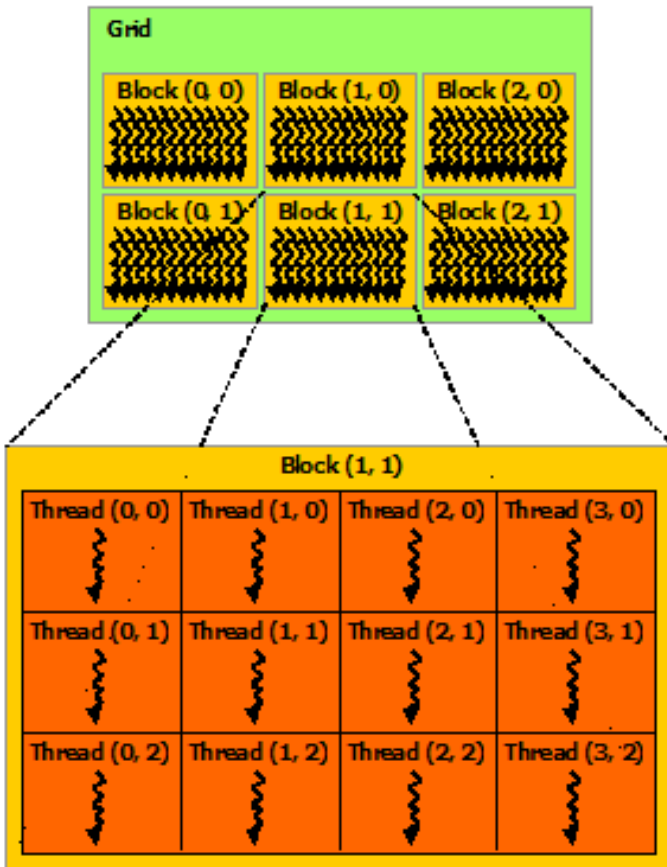


Figure 3: CUDA programming: grid of thread blocks (Source NVIDIA).

```

int main(int argc, char *argv[])
{
  cudaMalloc((void **)d_f, M * N * 9 * sizeof(double));
  cudaMalloc((void **)f_post, M * N * 9 * sizeof(double));
  .....
  .....
  int threadsAlongX = 32, threadsAlongY = 32;
  dim3 dimBlock(threadsAlongX, threadsAlongY, 1);
  dim3 dimGrid(ceil(float(m)/float(dimBlock.x)), ceil(float(m)/float(dimBlock.y)), 1);
  .....
  .....
  kernel <<< dimGrid, dimBlock >>> (d_f, f_post...);
  .....
}

```

here we have given the streaming part of the CUDA C program:

```

__global__ kernel(double *d_f, double *f_post...)
{
  int i = blockIdx.x * blockDim.x + threadIdx.x ;
  int j = blockIdx.y * blockDim.y + threadIdx.y ;
  if((i >= 0)&&(i <= m - 1)&&(j >= 0)&&(j <= n - 1)){
    double ixy = i + j * m;
    x plus 1
    int ip = (i == m - 1)?(0) : (i + 1);
    x minus 1

```

```

int im = (i == 0)?(m - 1) : (i - 1);
y plus 1
int jp = (j == n - 1)?(0) : (j + 1);
y minus 1
int jm = (j == 0)?(n - 1) : (j - 1);
d_f[ixy + 0 * m * n] = fpost[ixy + 0 * m * n];
d_f[ixy + 1 * m * n] = fpost[im + j * m + 1 * m * n];
.....
.....
d_f[ixy + 8 * m * n] = fpost[im + jp * m + 8 * m * n]; }
__syncthreads(); }

```

Double precision enable (Computing capability 3.5) a Tesla k40 NVIDIA graphics card (Table 1) is used for GPU computation and the PGI FORTRAN compiler for the CPU computation.

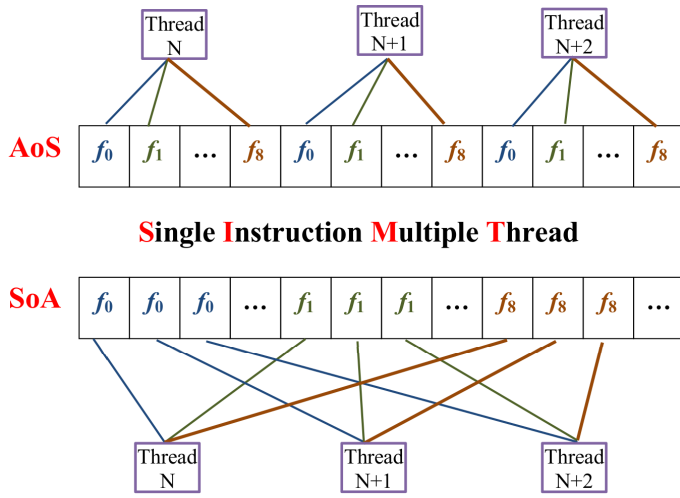


Figure 4: Data structure of AoS and SoA.

4.2. Data structure modification: AoS to SoA

One of the most important changes of the data structure modification in the GPU code from the array of structure (AoS) to the structure of array (SoA). In CPU based LBM code, the data layout of the distribution function is usually arranged as AoS because CPU can use of cache memory. For example, the distribution function $f_i(\mathbf{x}, t)$ is stored with index $(i + 9 \times x + 9 \times Nx \times y)$ for the D2Q9 model, as depicted in Figure. 4. But GPU is functioning based on single instruction multiple threads (SIMT) execution model. In CUDA C, to meet the requirement of coalescing memory access, the data layout should be changed to SoA in LBM implementation. Then the distribution function $f_i(\mathbf{x}, t)$ is stored with index $(x + Nx \times y + Nx \times Ny \times i)$, so that the parallel threads running the same instruction can access consecutive locations in memory [37, 28].

5. Results Discussion

In this paper, a new approach is proposed for the convective heat transfer from the fluid saturated porous media using the GPU parallel computing via CUDA C platform. Before conducting the main simulation the present CUDA C code has been validated for the lid-driven cavity flow, natural convection flow for side heated square cavity and mixed convection lid-driven cavity flow. The different cases are given below:

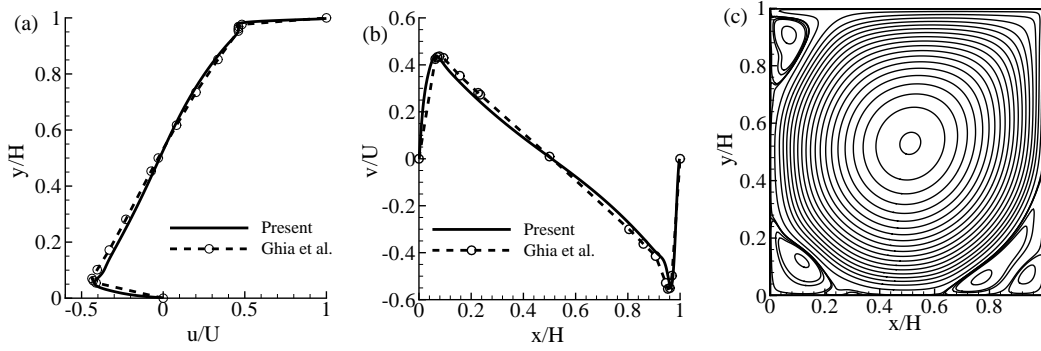


Figure 5: Comparison of the Lid-driven cavity flow with the results of Ghia et al. [38] (finite difference solutions) (a) u/U - velocity at mid x/H (b) v/U - velocity at mid y/H (c) Streamlines while $Re = 10000$ after 200000 iterations.

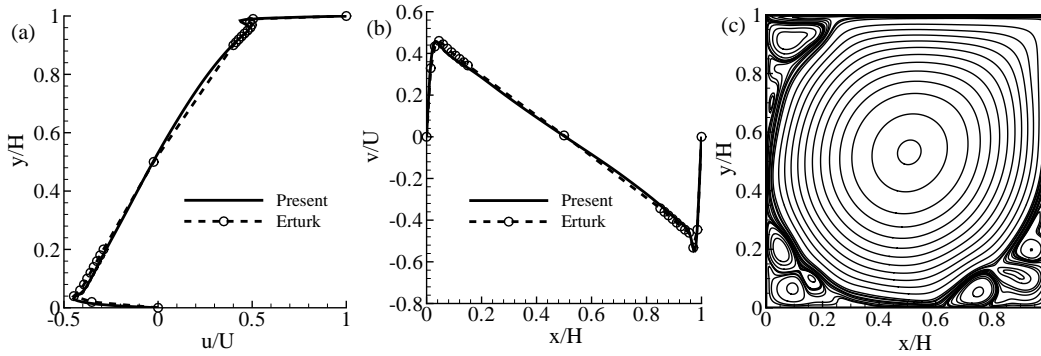


Figure 6: Comparison of the Lid-driven cavity flow with the results of Erturk et al. [39] (finite difference solutions) (a) u/U - velocity at mid x/H (b) v/U - velocity at mid y/H (c) Streamlines while $Re = 20000$ after 5000000 iterations.

Table 2: Comparison of the present results in terms of the average Nusselt number, \overline{Nu} , with pure fluid results for $Pr = 0.71$, $Da = 10^8$ and $\epsilon = 0.9999$

Ra	lattice size	de Vahl Davis [40]	FEM [32]	SRT-LB [5]	Present
10^3	256×256	1.116	1.127	1.1166	1.1299
10^4	256×256	2.238	2.245	2.2423	2.2655
10^5	256×256	4.509	4.521	4.5082	4.5106
10^6	256×256	8.817	8.800	8.3263	8.8135
	lattice size	[7]($Pr = 1.0$)	[42]	[43]	Present
10^7	512×512	16.81	16.79	16.523	16.609
10^8	640×640	30.81	30.51	30.225	30.221
10^9	1024×1024	55.80	57.35	—	54.561

Table 3: Comparison of the present average Nusselt number, \overline{Nu} for $\epsilon = 0.5$, with the previous experimental results of Sathe et. al [41] in a tiled cavity with porous media while the aspect ration $A = 10$.

$Da \times 10^{-4}$	$Ra \times 10^6$	Pr	Experimental [41]	Present	Error (%)
1.048	1.72	6.30	2.75	2.78	2.2
	2.47	6.11	3.30	3.24	1.8
	3.04	6.07	3.70	3.64	1.6
3.672	1.02	6.18	3.35	3.35	0.0
	1.67	6.16	4.07	4.16	2.2
	2.38	6.22	4.69	4.63	1.3

Table 4: The average Nusselt number, \overline{Nu} , with the Brinkman-Forchheimer model for $Pr = 1.0$.(Here FEM [32], SRT-LBM [3] and SRT-LBM [7])

Da	Ra	lattice size	$\epsilon = 0.4$				$\epsilon = 0.6$			
			[32]	[3]	[7]	Present	[32]	[3]	[7]	Present
10^{-2}	10^3	256×256	1.010	1.008	—	1.0197	1.015	1.012	—	1.0240
	10^4	256×256	1.408	1.367	—	1.3546	1.530	1.499	—	1.5079
	10^5	256×256	2.983	2.998	—	3.0293	3.555	3.422	—	3.4855
	10^6	256×256	—	—	—	6.1853	—	—	—	7.1410
10^{-4}	10^5	256×256	1.067	1.066	1.060	1.0681	1.071	1.068	1.063	1.0914
	10^6	256×256	2.550	2.603	2.614	2.6263	2.725	2.703	2.725	2.7418
	10^7	512×512	7.810	7.788	7.783	7.7831	8.183	8.457	8.576	8.1243
	10^8	640×640	—	—	16.960	16.841	—	—	19.210	18.895
10^{-6}	10^7	640×640	1.079	1.077	1.068	1.1095	1.079	1.077	1.068	1.1100
	10^8	640×640	2.970	2.969	3.152	2.9603	2.997	2.962	3.170	2.9801
	10^9	1024×1024	11.460	11.395	12.590	11.673	11.790	11.594	13.05	11.761
10^{-7}	10^8	640×640	—	—	—	1.1590	—	—	—	1.1727
	10^9	1024×1024	—	—	—	5.0369	—	—	—	5.0402
	10^{10}	2048×2048	—	—	—	12.623	—	—	—	13.111

5.1. Lid-driven cavity flow for $Re = 10000$ and 20000

Firstly, the MRT-LBM code has been validated only for fluid flow simulation considering the well benchmark lid driven cavity flow and then validated the double MRT-LBM code for the natural and mixed convection flow in a square cavity flow. The results of the lid-driven cavity flow are depicted in the Figures. 5 and 6 for the $Re = 10000$ and 20000 , respectively. A 1024×1024 lattice size is considered for both the Reynolds number. For $Re = 10000$ the results are compared with the results of Ghia et al. [38] and the results for $Re = 20000$ are compared with the results of Erturk et al. [39]. The center of the primary vortex is located at $(x, y) = (0.5119, 0.5271)$ and $(0.5081, 0.5291)$ for $Re = 10000$ and 20000 respectively and the corresponding locations were found by Erturk et al. [39] are $(0.5117, 0.5300)$ and $(0.5100, 0.5267)$. From these figures, it is clearly seen that the agreement between the present and previous simulations is quite excellent.

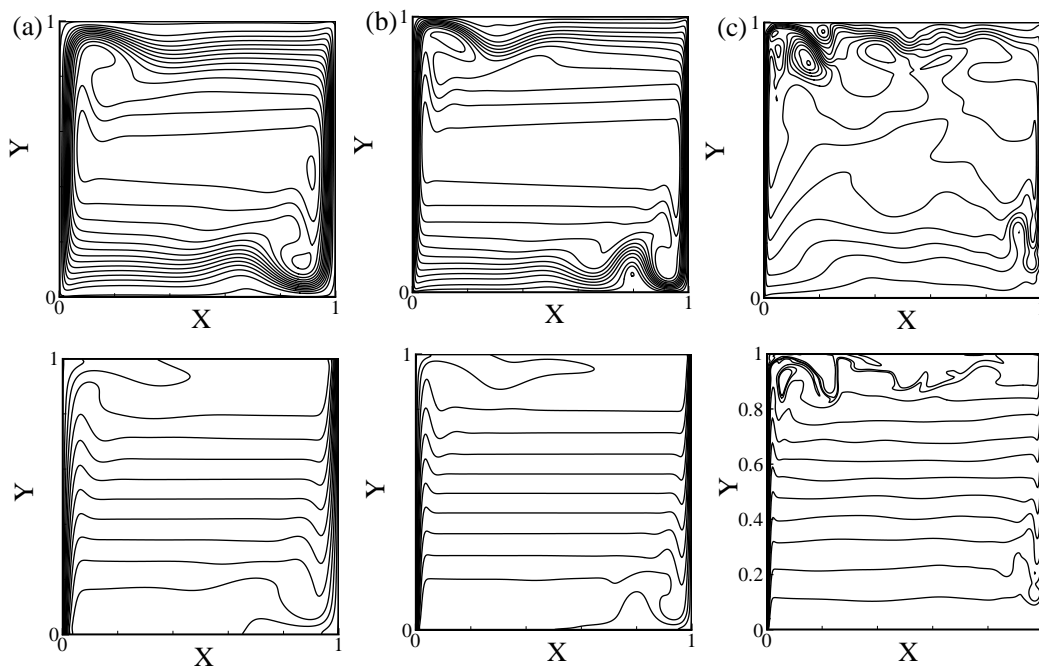


Figure 7: Streamlines (top) and isotherms (bottom) for pure fluid $Pr = 0.71$ (a) $Ra = 10^7$ (b) $Ra = 10^8$ (c) $Ra = 10^9$ while $\epsilon = 0.9999$ and $Da = 10^8$.

5.2. Natural convection flow in square cavity for $Ra = 10^3$ to 10^9

The dimensionless quantity governing this problem is the Prandtle number $Pr \stackrel{\text{def}}{=} \nu/\alpha$ and the Rayleigh number $Ra \stackrel{\text{def}}{=} g\beta(T_h - T_c)H/\nu\alpha$. In the simulation, the fluid velocity is normalized by the characteristic velocity $\sqrt{\beta g \Delta T H}$ and the dimensionless temperature is $\theta = (T - T_c)/(T_h - T_c)$.

In order to validate the fluid flow and heat transfer case, the MRT-LBM results in terms of the average Nusselt number, \overline{Nu} , obtained from the simulations are compared with the available data of SRT-LBM [5], finite difference data of de Vahl Davis [40], finite Element data of Nithiarasu et al. [32] for the clear fluid ($Pr = 0.71$) in a side-heated square cavity under the same boundary conditions for the laminar flow where $10^3 \leq Ra \leq 10^6$. For the transition-to-turbulent flows, the present results are also compared with the available results of SRT-LBM of Vishnampet et al. [7], Dixit and Babu [42] and the spectral method of Le Quéré [43] where the Ra ranges from 10^7 to 10^9 . In these cases, the Darcy number $Da = 10^8$ and the porosity $\epsilon = 0.9999$ have been considered. The results are given in Table 2 and the agreement shows

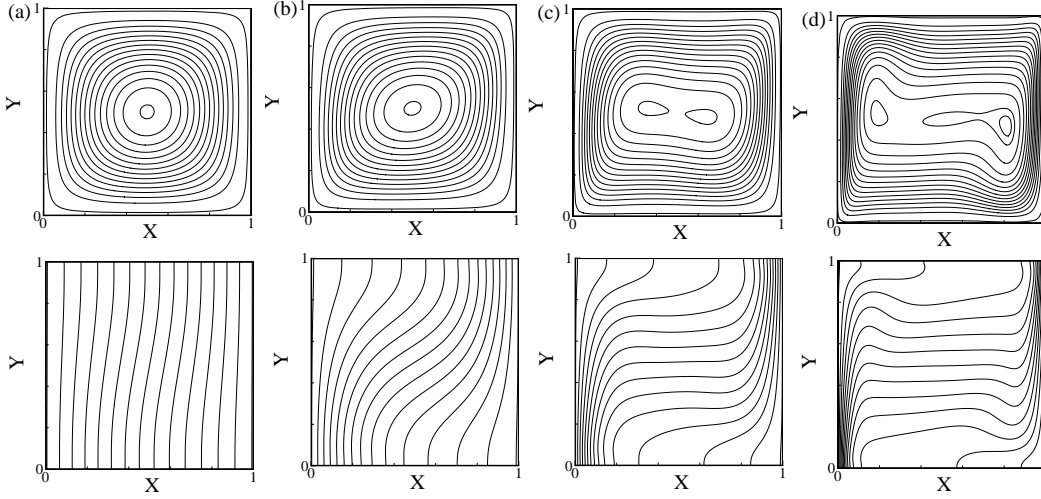


Figure 8: Streamlines (top) and isotherms (bottom) (a) $Ra = 10^3$ (b) $Ra = 10^4$ (c) $Ra = 10^5$ (d) $Ra = 10^6$ while, $Pr = 1.0$, $Da = 10^{-2}$ and $\epsilon = 0.6$.

quite well up to $Ra = 10^8$. For $Ra = 10^9$, the average Nusselt number of Dixit and Babu [42] is 57.35 but in the present case it is 54.56.

Figure 7(a)-(c) depicts the streamlines and isotherms for the transition-to-turbulent flows while $Pr = 0.71$ and $Ra = 10^7$ to $Ra = 10^9$ while $Da = 10^8$ and $\epsilon = 0.9999$. From these figures, it has been seen clearly the pattern of the streamlines and isotherms for $Ra = 10^7$ and 10^8 coincides with the available results in the literature, but for $Ra = 10^9$ varies slightly. Since the results of $Ra = 10^7$ and 10^8 are for transitional flows, but the flow is turbulent for $Ra = 10^9$ (Dixit and Babu [42]).

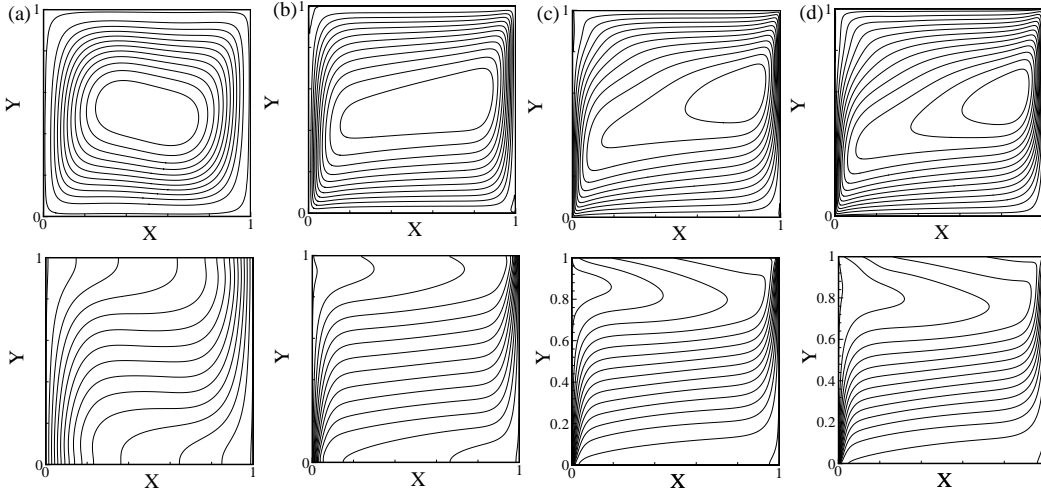


Figure 9: Streamlines (top) and isotherms (bottom) for (a) $Da = 10^{-2}$, $Ra = 10^5$ (b) $Da = 10^{-4}$, $Ra = 10^7$ (c) $Da = 10^{-6}$, $Ra = 10^9$ and (d) $Da = 10^{-7}$, $Ra = 10^{10}$ while $Pr = 1.0$ and $\epsilon = 0.4$.

5.3. Natural convection flow with porous media for $Ra = 10^3$ to 10^{10}

In Table 3, the present numerical results are compared with the previous experimental results of Sathe et al. [41] for the different Ra , Pr and Da while the aspect ratio $A = 10$. The comparison shows that the agreement is excellent and the maximum error is 2.2%. So, the present MRT-LBM method is suitable for simulating the flow phenomena and heat transfer for the porous media.

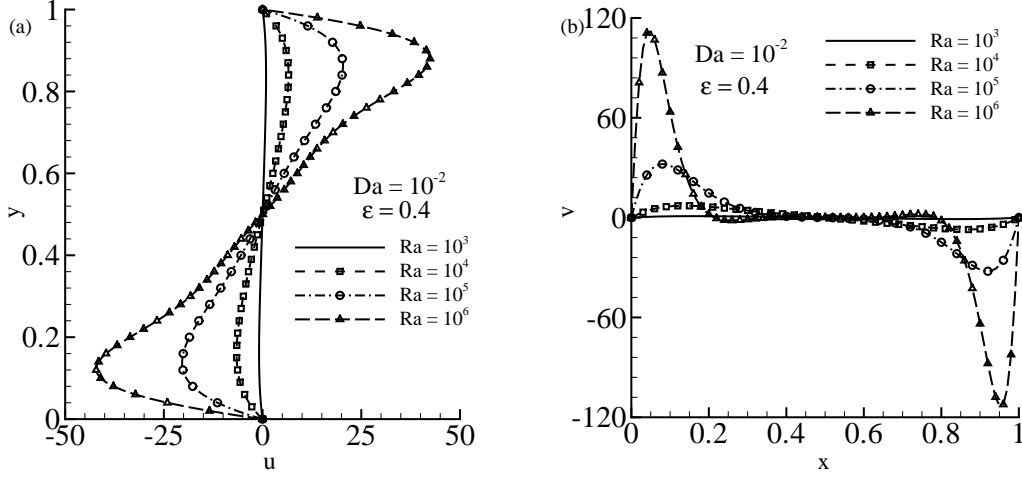


Figure 10: Velocity distribution (a) u -velocity and (b) v -velocity for the different Ra while, $Pr = 1.0$, $Da = 10^{-2}$ and $\epsilon = 0.4$.

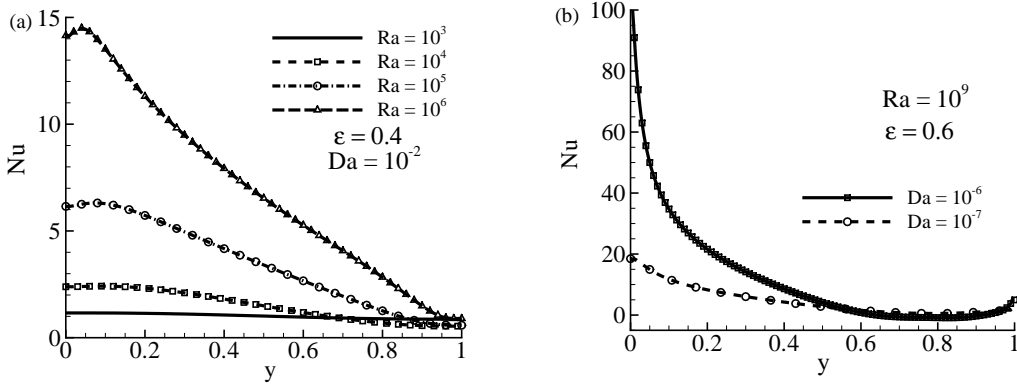


Figure 11: Local Nusselt number Nu for (a) different Ra while $Da = 10^{-2}$ and $\epsilon = 0.4$ (b) different Da while $Ra = 10^9$, $Pr = 1.0$ and $\epsilon = 0.6$.

The streamlines and isotherms for the low Rayleigh number, Ra ranges from 10^3 to 10^6 while $Da = 10^{-2}$ and $\epsilon = 0.6$ are depicted in the Figure 8(a)-(d). For $Da = 10^{-2}$ the fluid flow behaving as like clear fluid where conduction is dominated for $Ra = 10^3$ and convection is dominated for the higher Ra . Figure 9(a)-(d) show the streamlines and isotherms for the relative high Rayleigh number with Darcy ($Da \leq 10^{-6}$) and non-Darcy ($Da \geq 10^{-4}$) regime while $\epsilon = 0.4$. The pattern of the streamlines and isotherms are clearly different for the Darcy and non-Darcy region and they are qualitatively similar with the results of (Dixit and Babu [42]).

The effects of the Rayleigh number on the velocity distribution are depicted in Figure 11(a)-(b) while $Da = 10^{-2}$ and $\epsilon = 0.4$. It is seen that the velocity increase while increasing the Rayleigh numbers. For both velocity distribution, larger velocity occurs near the walls and the minimum velocity occurs at the center of the cavity where minimum values of stream function occur. The local rate of heat transfer in terms of the local Nusselt number Nu is illustrated in the Figure 11(a)-(b) for the different Ra while $Da = 10^{-2}$ and $\epsilon = 0.4$ and for the different Da while $Ra = 10^9$ and $\epsilon = 0.6$, respectively. The local Nusselt number increases for increasing the Rayleigh numbers but the local Nu decreases while decreasing the Darcy numbers.

In the Table 4, the average Nusselt number, \overline{Nu} , has been inserted for the different $Ra = 10^3$ to 10^{10} , Darcy number $Da = 10^{-2}$ to 10^{-7} and the porosity $\epsilon = 0.4$ and 0.6 . The results obtained by present MRT-LBM are compared with the results obtained by the finite element method

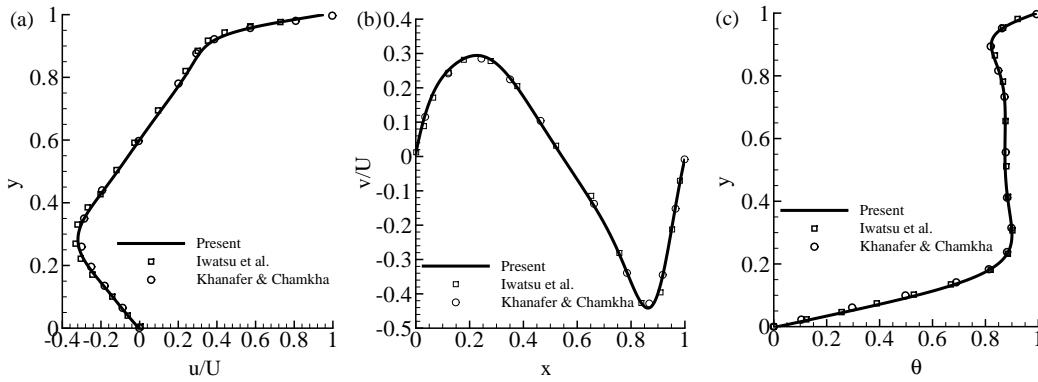


Figure 12: Comparison of the present mixed convection results with the available results of Iwatsu et al. [44], Khanafar and Chamkha [45] (a) u/U velocity (b) v/U velocity (c) temperature θ while $Re = 400$, $Gr = 100$, $Pr = 0.71$ and no porous media.

Table 5: Mixed convection: comparison of the average Nusselt number, \overline{Nu} with the available numerical results of Iwatsu et al. [44], Khanafar et al. [45, 46], Khalek [47], Tiwari and Das [48] and Kefayati et al. [49] for $Gr = 100$, and $Pr = 0.71$.

Re	Ri	lattice size	[44]	[45]	[46]	[47]	[48]	[49]	Present
100	0.01	256×256	1.94	2.01	2.02	1.985	2.10	2.09	2.077
400	0.00062	256×256	3.84	3.91	4.01	3.88	3.85	4.08	4.032
1000	0.0001	512×512	6.33	6.33	6.42	6.35	6.33	6.55	6.422

of Nithiarasu et al. [32], SRT-LBM of Guo and Zhao [3] and the SRT-LBM of Vishnampet et al. [7]. The comparison shows the agreement is quite excellent. A set of new results are presented for very high and very small Darcy numbers that are presented in the bottom the Table 4. For $\epsilon = 0.4$ and 0.6 the Average Nusselt number increases while the Rayleigh increases from $Ra = 10^8$ to 10^{10} keeping the Darcy number fixed at $Da = 10^{-7}$.

5.4. Mixed convection flow in a lid-driven square cavity

In the mixed convection case, the fluid velocity are non-dimensionalised by the lid velocity $U = 0.1$ and the temperature as like natural convection flow. Firstly, the code is validated while the heated lid is moving along x -direction in pure fluid case considering the $Gr = 100$ and the $Re = 100, 400$ and 1000 . Figure 12(a)-(c) shows the velocity and temperature distribution respectively while $Re = 400$, $Gr = 100$, $Da = 10^8$ and $\epsilon = 1$. These velocity and temperature distribution are compared with the results of Iwatsu et al. [44] and Khanafar et al. [45] that shows an excellent agreement. Another comparison has been made regarding the average Nusselt number, \overline{Nu} , that is shown in Table 5 for the three different Reynolds number, $Re = 100, 400$ and 1000 , and $Gr = 100$. The agreement of the average Nusselt number with the available results of Iwatsu et al. [44], Khanafar et al. [45, 46], Khalek [47], Tiwari and Das [48] and Kefayati et al. [49] is quite acceptable.

Figure 13(a)-(d) depict the streamlines (top) and isotherms (bottom) for the two different Re while $Pr = 0.71$, $Gr = 100$. For $Re = 400$ and 1000 there three vortex, primary, secondary and tertiary. The primary vortex spans in the whole cavity except the right and left the bottom corner. In the isotherms, it is seen the maximum temperature occurs near the top wall and the minimum at the bottom wall and qualitatively these results agree with results of Iwatsu et al. [44] and Khanafar et al. [45].

Table 6: Mixed convection case: the average Nusselt number, \overline{Nu} , with the Brinkman-Forchheimer model for $Pr = 1.0$.

Gr	Da	Re	lattice size	$\epsilon = 0.4$	$\epsilon = 0.6$
	10^{-1}	400	256×256	4.511	4.043
100		1000	512×512	7.255	6.569
	10^{-2}	400	256×256	3.265	2.898
		1000	512×512	5.253	4.334

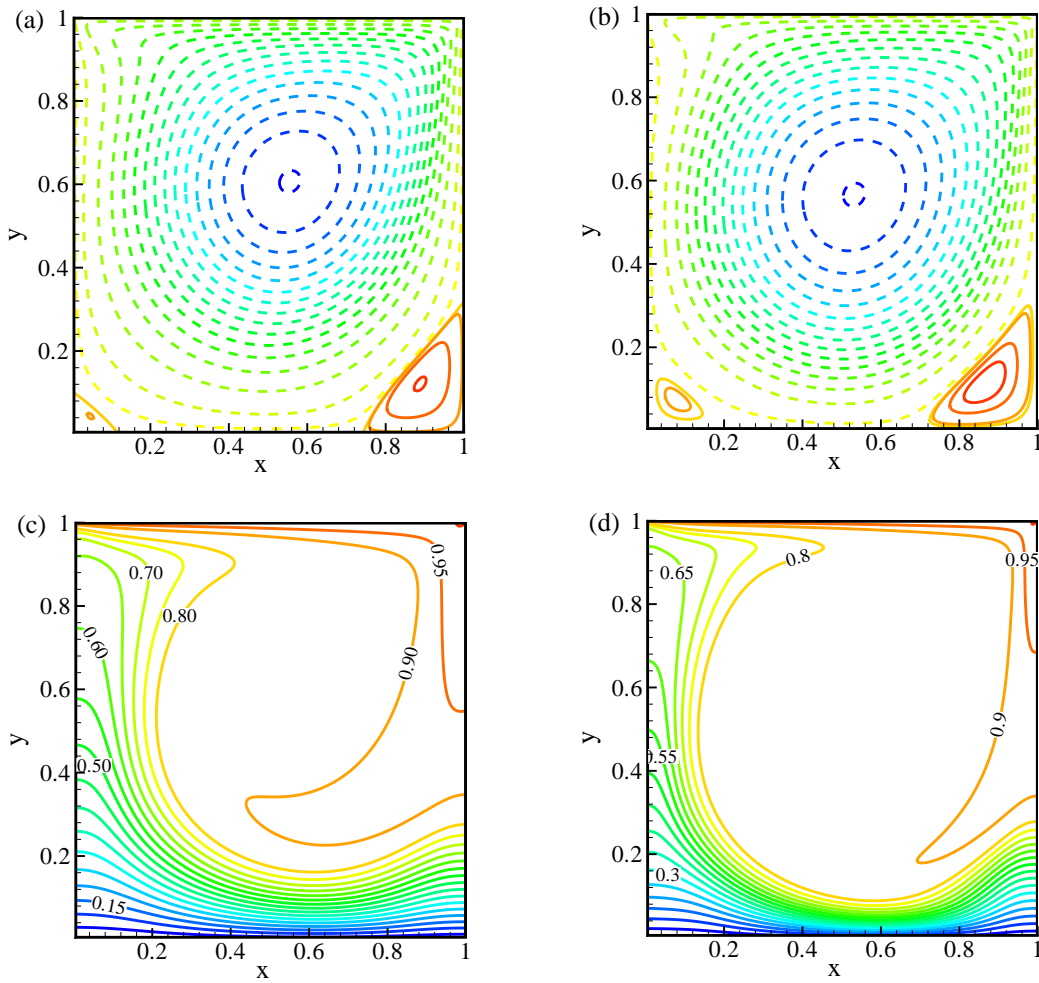


Figure 13: Streamlines (top):(a) $Re = 400$ and (b) $Re = 1000$ and isotherms (bottom): (c) $Re = 400$ and (d) $Re = 1000$ while $Pr = 0.71$ and no porous media.

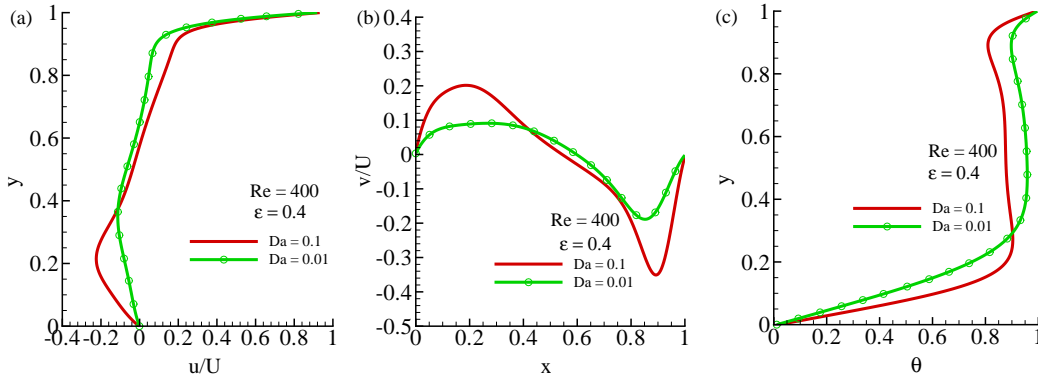


Figure 14: Velocity and temperature distribution for the different Da (a) u/U -velocity (b) v/U and $-$ velocity (c) temperature θ while $Re = 400$, $Gr = 100$ $Pr = 1$ and $\epsilon = 0.4$.

5.5. Mixed convection flow in lid-driven square cavity with porous media

The effects of the Darcy number, Da on the flow field and temperature distribution is illustrated in the Figure 14(a)-(c) while $Re = 400$ and $\epsilon = 0.4$. From the frame (a) and (b), it is observed that the u at $x = 0.5$ and v velocity at $y = 0.5$ decreasing for decreasing the Darcy number. It is obvious, since for the smaller values of Da , increase the porous matrix inside the cavity that retards the fluid motion. But the temperature, θ , at $x = 0.5$ enhanced for smaller values of Da . Figure 15(a)-(d) depict the streamlines (top) and isotherms (bottom) for the two different porosities $\epsilon = 0.4$ and 0.6 while $Gr = 100$ and $Re = 1000$. From the frames of the streamlines and isotherms, it is clearly seen that the effect of porosity in the porous media is significant. For changing the porosity $\epsilon = 0.4$ to $\epsilon = 0.6$ the distribution of the stream function and isotherms changes that would change the rate of heat transfer. For larger value of $\epsilon = 0.6$, the center of the primary vortex shifted to right top corner and increase the temperature of the fluid in side the cavity.

The average Nusselt number, \overline{Nu} , is inserted in the Table 5 for the different Reynolds number, Re , the Darcy number, Da and for $\epsilon = 0.4$ and 0.6 while $Gr = 100$. For the higher Re , the \overline{Nu} increases but for decreasing Da , reducing the average rate of heat transfer. The effects of the porosity are also significant in the rate of heat transfer. The average rate of heat transfer decreases for increasing the values of porosity ϵ that is opposite to the natural convection case.

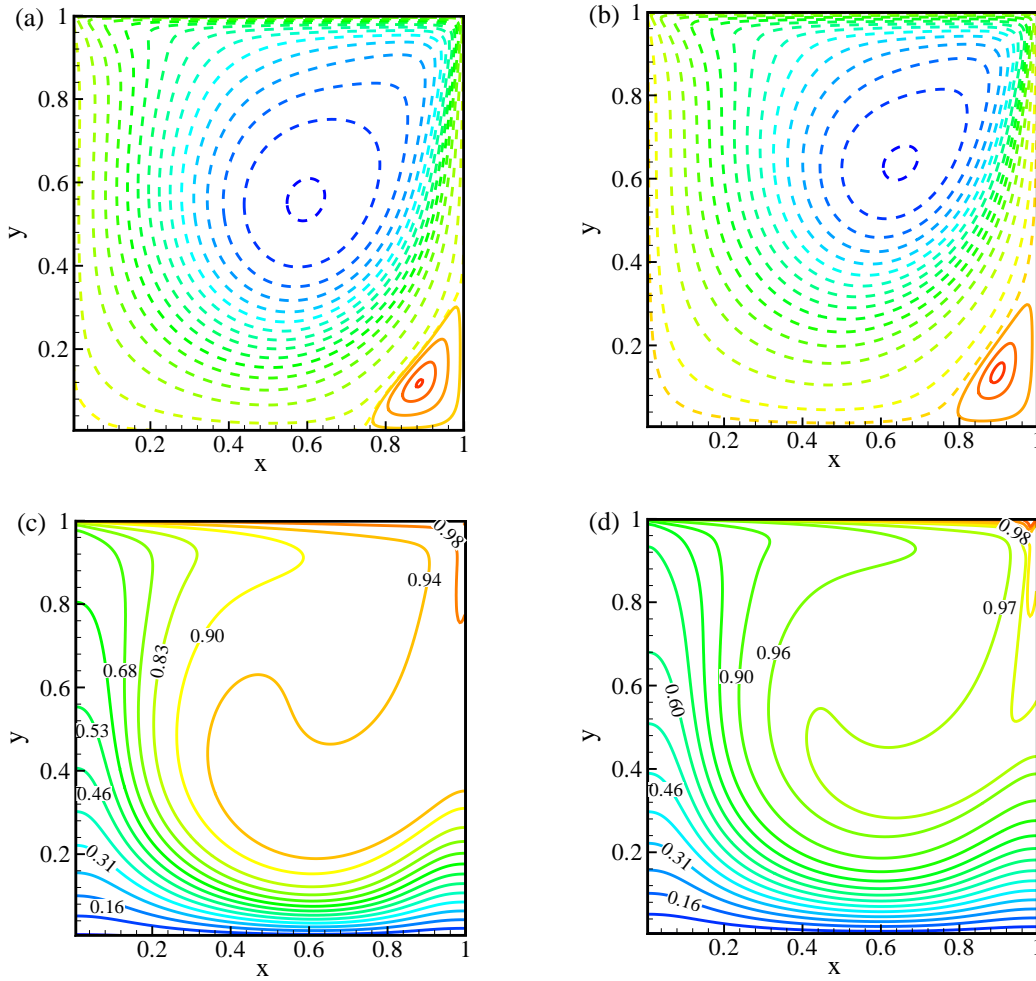


Figure 15: Streamlines (top):(a) $\epsilon = 0.4$ and (b) $\epsilon = 0.6$ and isotherms (bottom): (c) $\epsilon = 0.4$ and (d) $\epsilon = 0.6$ while $Re = 1000$, $Gr = 100$, $Pr = 1$ and $Da = 10^{-2}$.

5.6. GPU performance over CPU

Table 7 shows a comparison of the GPU parallel performance over a sequential CPU performance in terms of the simulation time per iteration step. The GPU performance is higher than CPU and the speed up is calculated as a ratio of CPU simulation time over GPU time. From this table, it is interesting to see that for 128×128 lattice size the simulation in GPU speeds up approximately 19 times than the CPU but for 2048×2048 it is 144 times faster. The performance of the GPU implementation strongly depends on the grid size. So, for larger grid problems the GPU performance is better than the lower grid size.

Achieving better numerical results using LBM requires larger computational grid size. This requirement is the crucial factor for the overall performance optimization. Since CPU has more latency in processor clock speed than GPU hardware, in the case of the smaller computational grid the performance of the numeric model using sequential programming in CPU surpasses the model implemented in GPU parallel computing environment. However, GPU has more throughput than CPU due to its many processor architectures which are very suitable for parallel task execution of numerical calculations. GPU hardware consists an array of scalar processors which are executed in a group predefined in terms of GPU hardware. During the invocation of the parallel code in CUDA computing environment, GPU hardware requires a minimum amount of grid size allocation in global memory bus to occupy all its processors for a synchronized task execution. Lower latency in device to host memory transfer is seen in smaller grid allocation which can potentially degrade the overall performance of the numerical analysis.

Table 7: MRT-LBM simulation time in CPU and GPU for the different mesh arrangements.

lattice size	Time (s)/step:	CPU	GPU	Speed up = t_{CPU}/t_{GPU}
128 × 128		0.00524	0.000275	19.05
256 × 256		0.04105	0.00073	56.23
512 × 512		0.18532	0.002566	72.22
1024 × 1024		0.29535	0.01004	88.25
2048 × 2048		5.76447	0.040113	143.71

This latency issue can be significantly evaded using larger grid allocation for hiding the memory issues by instantaneous data sharing among large amount of scalar processors. Moreover, the memory latency issue can be avoided with the assignment of the larger grid for the numerical calculation. Thus, the overall performance can be improved with gradually incrementing the grid size of the numerical model. Even, the performance speed up becomes substantially better in GPU than CPU programs whenever grid size is incremented. According to Lin et al. [27], for larger computational grids require more arithmetic operations, that would hide the memory latency and hence show a greater parallel performance.

6. Conclusions

In this paper, a double multiple-relaxation-time lattice Boltzmann method is proposed for the porous media with the Brinkman-Forchheimer extended Darcy model to simulate the natural and mixed convection flows in a square cavity. The numerical simulation has been done using the state-of-the-art GPU parallel computing via CUDA C platform. For the porous media, the Brinkman-Forchheimer model is directly used as the source term through the equilibrium distribution function. This approach is completely new that differs with the approach used in the single-relaxation-time lattice Boltzmann method. The numerical results for the natural convection case are computed for the wide range of Rayleigh numbers, $10^3 \leq Re \leq 10^{10}$, Darcy number $10^{-2} \leq Da \leq 10^{-7}$ and the porosity parameter $\epsilon = 0.4$ and 0.6 . In the mixed convectional case, the simulations are done for the Reynolds number $Re = 400, 100$ and Grashof number $Gr = 100$ with $10^{-1} \leq Da \leq 10^{-2}$.

For increasing the Ra in natural convection and Re in mixed convection the velocity increases but in both cases the velocity decreases while the Darcy number decreases. The average Nusselt number, \overline{Nu} , increases for increasing the porosity ϵ in natural convection but in mixed convection case opposite phenomena has happened.

The present results are compared with the available results computed by the finite difference, finite element method and single-relaxation time lattice Boltzmann method, and the comparison shows better agreement indeed. It is also well known that the MRT-LBM is superior to SRT-LBM in terms of the numerical stability. The forcing term implemented by nine discrete forcing moments that are added separately with the moment's space. The present proposed approach for the porous media with MRT-LBM can be used for other application in fluid flow and heat transfer simulation.

Using the CUDA C, the existing MRT-LBM FORTRAN 90 code has been re-written for GPU parallel computing that speeds up the simulation significantly. Precisely, in this Tesla k40 GPU it is 144 times faster than the core i7 CPU simulation with 2048×2048 lattice size. For the larger grids problem, GPU is more efficient than the smaller grids problems.

Acknowledgment

The first author gratefully acknowledges NVIDIA Corporation for granting the Tesla k40 GPU card and the PGI group for providing the University developer license of "PGI Accelerator Fortran/C/C++ compiler for a Workstation in Linux". This research is conducted by the financial support of North South University (NSU) "**Faculty Research Grant 2016-2017**".

References

- [1] Y. Ye, K. Li, Y. Wang, T. Deng, Parallel computation of entropic lattice Boltzmann method on hybrid CPU–GPU accelerated system, *Comput. Fluids* 110 (2015) 114–121.
- [2] E. Calore, A. Gabbana, J. Kraus, E. Pellegrini, S. Schifano, R. Tripicciono, Massively parallel lattice–Boltzmann codes on large GPU clusters, *Parall. Comput.* 58 (2016) 1–24.
- [3] Z. Guo, T. S. Zhao, A lattice Boltzmann model for convection heat transfer in porous media, *Numer. Heat Trans.: Part B* 47 (2) (2005) 157–177.
- [4] Z. Guo, T. S. Zhao, Lattice Boltzmann model for incompressible flows through porous media, *Phys. Rev. E* 66 (3) (2002) 036304.
- [5] T. Seta, E. Takegoshi, K. Okui, Lattice Boltzmann simulation of natural convection in porous media, *Math. Comput. Simul.* 72 (2) (2006) 195–200.
- [6] T. Seta, E. Takegoshi, K. Kitano, K. Okui, Thermal lattice Boltzmann model for incompressible flows through porous media, *J. Therm. Sci. Tech.* 1 (2) (2006) 90–100.
- [7] R. Vishnampet, A. Narasimhan, V. Babu, High Rayleigh number natural convection inside 2D porous enclosures using the lattice Boltzmann method, *ASME J. Heat Trans.* 133 (6) (2011) 062501.
- [8] X. Shan, H. Chen, Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation, *Phys. Rev. E* 49 (4) (1994) 2941.
- [9] A. J. C. Ladd, Sedimentation of homogeneous suspensions of non-Brownian spheres, *Phys. Fluids* 9 (3) (1997) 491–499.
- [10] K. N. Premnath, M. J. Pattison, Computation of MHD flows using the lattice Boltzmann method, MetaHeuristics, LLC, Santa Barbara, CA.
- [11] H. Nemati, M. Farhadi, K. Sedighi, H. R. Ashorynejad, E. Fattahi, Magnetic field effects on natural convection flow of nanofluid in a rectangular cavity using the lattice Boltzmann model, *Scientia Iranica* 19 (2) (2012) 303–310.
- [12] E. Fattahi, M. Farhadi, K. Sedighi, H. Nemati, Lattice Boltzmann simulation of natural convection heat transfer in nanofluids, *Int. J. Therm. Sci.* 52 (2012) 137–144.
- [13] R. Du, B. Shi, X. Chen, Multi-relaxation-time lattice Boltzmann model for incompressible flow, *Phys. Lett. A* 359 (6) (2006) 564–572.
- [14] P. Lallemand, L.-S. Luo, Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability, *Phys. Rev. E* 61 (6) (2000) 6546.
- [15] D. d’Humières, In *Rarefied Gas Dynamics: Theory and Simulations*, Prog. Astronaut. Aeronaut. Vol. 159, edited by B. D. Shizgal and D. P. Weaver, AIAA, Washington, D.C.

- [16] Z.-H. Chai, B.-C. Shi, Z. Lin, Simulating high Reynolds number flow in two-dimensional lid-driven cavity by multi-relaxation-time lattice Boltzmann method, *Chinese Physics* 15 (8) (2006) 1855.
- [17] X. Guo, C. Zhong, C. Zhuo, J. Cao, Multiple-relaxation-time lattice Boltzmann method for study of two-lid-driven cavity flow solution multiplicity, *Theor. Comput. Fluid Dyn.* 28 (2) (2014) 215–231.
- [18] M. A. Moussaoui, M. Jami, A. Mezhhab, H. Naji, M. Bouzidi, Multiple-relaxation-time lattice Boltzmann computation of channel flow past a square cylinder with an upstream control bi-partition, *Int. J. Num. Meth. Fluids* 64 (6) (2010) 591–608.
- [19] R. Du, W. Liu, A new multiple-relaxation-time lattice Boltzmann method for natural convection, *J. Sci. Comput.* 56 (1) (2013) 122–130.
- [20] M. Amine Moussaoui, M. Jami, A. Mezhhab, H. Naji, Computation of heat transfer and fluid flow in an obstructed channel using lattice Boltzmann method, *Engn. Comput.* 27 (1) (2010) 106–116.
- [21] Y. Guo, R. Bennacer, S. Shen, D. Ameziani, M. Bouzidi, Simulation of mixed convection in slender rectangular cavity with lattice Boltzmann method, *Int. J. Numer. Meth. Fluids* 20 (1) (2010) 130–148.
- [22] A. Mezhhab, M. A. Moussaoui, M. Jami, H. Naji, M. Bouzidi, Double MRT thermal lattice Boltzmann method for simulating convective flows, *Phys. Lett. A* 374 (34) (2010) 3499–3507.
- [23] F. Chen, A. Xu, G. Zhang, Y. Li, Multiple-relaxation-time lattice Boltzmann model for compressible fluids, *Phys. Lett. A* 375 (21) (2011) 2129–2139.
- [24] J. Wang, D. Wang, P. Lallemand, L.-S. Luo, Lattice Boltzmann simulations of thermal convective flows in two dimensions, *Comput. Math. Appl.* 65 (2) (2013) 262–286.
- [25] B. Trouette, Lattice Boltzmann simulations of a time-dependent natural convection problem, *Comput. Math. Appl.* 66 (8) (2013) 1360–1371.
- [26] T. Zhang, D. Che, Double MRT thermal lattice Boltzmann simulation for MHD natural convection of nanofluids in an inclined cavity with four square heat sources, *Int. J. Heat Mass Trans.* 94 (2016) 87–100.
- [27] L.-S. Lin, H.-W. Chang, C.-A. Lin, Multi relaxation time lattice Boltzmann simulations of transition in deep 2D lid driven cavity using GPU, *Comput. Fluids* 80 (2013) 381–387.
- [28] A. Xu, L. Shi, T. Zhao, Accelerated lattice Boltzmann simulation using GPU and OpenACC with data management, *Int. J. Heat Mass Trans.* 109 (2017) 577–588.
- [29] Q. Ren, C. L. Chan, Natural convection with an array of solid obstacles in an enclosure by lattice Boltzmann method on a CUDA computation platform, *Int. J. Heat Mass Trans.* 93 (2016) 273–285.
- [30] Q. Ren, C. L. Chan, Numerical study of double-diffusive convection in a vertical cavity with Soret and Dufour effects by lattice Boltzmann method on GPU, *Int. J. Heat Mass Trans.* 93 (2016) 538–553.

- [31] Q. Ren, C. L. Chan, GPU accelerated numerical study of PCM melting process in an enclosure with internal fins using lattice Boltzmann method, *Int. J. Heat Mass Trans.* 100 (2016) 522–535.
- [32] P. Nithiarasu, K. N. Seetharamu, T. Sundararajan, Natural convective heat transfer in a fluid saturated variable porosity medium, *Int. J. Heat Mass Trans.* 40 (16) (1997) 3955–3967.
- [33] S. Ergun, Fluid flow through packed columns, *Chem. Engn. Prog.* 48 (1952) 89–94.
- [34] Z. Guo, C. Shu, Lattice Boltzmann method and its application in engineering, Vol. 3, World Scientific, 2013.
- [35] NVIDIA, Compute Unified Device Architecture Programming Guide, version 7.5, 20 March 2017, CUDA Developer Zone, NVIDIA, <http://docs.nvidia.com/cuda/cuda-c-programming-guide/axzz4eZNceJtc>.
- [36] C. Obrecht, F. Kuznik, B. Tourancheau, J.-J. Roux, A new approach to the lattice Boltzmann method for graphics processing units, *Comput. Math. Appl.* 61 (12) (2011) 3628–3638.
- [37] N. Delbosc, J. L. Summers, A. Khan, N. Kapur, C. J. Noakes, Optimized implementation of the Lattice Boltzmann Method on a graphics processing unit towards real-time fluid simulation, *Comput. Math. Appl.* 67 (2) (2014) 462–475.
- [38] U. G. Ghia, K. N. Ghia, C. T. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *J. Comp. Phys.* 48 (3) (1982) 387–411.
- [39] E. Erturk, T. C. Corke, C. Gökçöl, Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers, *Int. J. Numer. Meth. Fluids* 48 (7) (2005) 747–774.
- [40] G. de Vahl Davis, Natural convection of air in a square cavity: a bench mark numerical solution, *Int. J. Numer. Meth. Fluids* 3 (3) (1983) 249–264.
- [41] S. B. Sathe, T. W. Tong, M. A. Faruque, Experimental study of natural convection in a partially porous enclosure, *J. Thermophysics* 1 (1987) 260–267.
- [42] H. N. Dixit, V. Babu, Simulation of high Rayleigh number natural convection in a square cavity using the lattice Boltzmann method, *Int. J. Heat Mass Trans.* 49 (3) (2006) 727–739.
- [43] P. Le Quéré, Accurate solutions to the square thermally driven cavity at high Rayleigh number, *Comput. Fluids* 20 (1) (1991) 29–41.
- [44] R. Iwatsu, J. M. Hyun, K. Kuwahara, Mixed convection in a driven cavity with a stable vertical temperature gradient, *Int. J. Heat Mass Trans.* 36 (6) (1993) 1601–1608.
- [45] K. M. Khanafer, A. J. Chamkha, Mixed convection flow in a lid-driven enclosure filled with a fluid-saturated porous medium, *Int. J. Heat Mass Trans.* 42 (13) (1999) 2465–2481.
- [46] K. M. Khanafer, A. M. Al-Amiri, I. Pop, Numerical simulation of unsteady mixed convection in a driven cavity using an externally excited sliding lid, *Eur. J. Mech. B/Fluids* 26 (5) (2007) 669–687.
- [47] M. Abdelkhalek, Mixed convection in a square cavity by a perturbation technique, *Comput. Mater. Sci.* 42 (2) (2008) 212–219.

- [48] R. K. Tiwari, M. K. Das, Heat transfer augmentation in a two-sided lid-driven differentially heated square cavity utilizing nanofluids, *Int. J. Heat Mass Trans.* 50 (9) (2007) 2002–2018.
- [49] G. Kefayati, S. Hosseinizadeh, M. Gorji, H. Sajjadi, Lattice Boltzmann simulation of natural convection in an open enclosure subjugated to water/copper nanofluid, *Int. J. Therm. Sci.* 52 (2012) 91–101.