International Journal of Heat and Fluid Flow 60 (2016) 31–46

Contents lists available at ScienceDirect

International Journal of Heat and Fluid Flow

journal homepage: www.elsevier.com/locate/ijheatfluidflow

Numerical simulation of heat transfer in particulate flows using a thermal immersed boundary lattice Boltzmann method

A. Eshghinejadfard*, D. Thévenin

Laboratory of Fluid Dynamics and Technical Flows, University of Magdeburg "Otto von Guericke", Universitätsplatz 2, 39106 Magdeburg, Germany

ARTICLE INFO

Article history:
Received 14 September 2015
Revised 24 March 2016
Accepted 19 April 2016
Available online 10 May 2016

Keywords:
Thermal lattice Boltzmann method
Immersed boundary method
Particulate flow
Heat transfer
Moving particle
Natural convection

ABSTRACT

In the current work the lattice Boltzmann method (LBM) is applied to investigate heat transfer phenomena in particulate flows. Different cases involving both two- and three-dimensional configurations are studied. For the fluid–particle interactions the direct-forcing and direct-heating immersed boundary (IB) method are applied to calculate the hydrodynamic force and energy exchange between the particle and the fluid, respectively. This Eulerian–Lagrangian approach captures the fluid flow around the particles with high accuracy. The Boussinesq approximation is applied to the coupling between flow and temperature fields. The energy equation is solved using a double-population model in the LBM framework. Numerical simulations reveal that this thermal IB-LBM can accurately predict the particle motion. A particularly interesting case involves particles with a variable temperature, where the competition between gravity and buoyancy induced by the temperature gradient can make particles sink or rise. It is observed that cold particles settle down faster than hot particles. Also, the thermal IB-LBM has been implemented for a collection of spherical particles. In this manner, the behavior of catalyst particles can be accurately predicted, as demonstrated in the last application, involving 60 particles interacting in an enclosure.

1. Introduction

Particulate flows are present in many natural and industrial applications such as filtration, pollution control, blood clogging, fluidized bed reactors, crystallization, or chemical reactors involving catalyst particles. For many years, the study and design of particle systems were confined to empirical and experimental research. During the past decades and as a result of increased computational capability and advanced modeling techniques, the numerical analysis of particulate flows has attracted the attention of many research groups. Three major numerical approaches can be identified in this regard. The first approach is the *two-fluid* model (as special case of the multi-fluid model) in which the properties of the particles are assumed to be continuous, like those of a pure fluid. Thus, conservation equations of mass, momentum and energy are developed through an averaging process and the constitutive relations for the solid phase are usually closed using the kinetic theory of granular flow. These equations are discretized at each computational node and solved through a procedure similar to that used for the fluid (Crowe et al., 2011; Chen and Wang, 2014). The interaction between the two phases is described by drag force correlations. This model is an Eulerian–Eulerian approach that does not properly model all details of particle–particle and particle–fluid interactions. The second model is the *discrete particle model* (DPM) and its extension, the discrete element model (DEM). In this model each particle is treated as a point force and its position is determined via Newton's equations of motion. This model is an Eulerian–Lagrangian approach and estimates the force acting on each particle by an empirical drag force. The third approach is the so called *direct numerical simulation* (DNS), which is the most accurate method. Here, the Eulerian grid is typically an order of magnitude smaller than the size of the particles, so that the fluid flow behavior between the particles is also computed. In this case, no correlation is required and both particle–particle and particle–fluid interactions can be modeled in a realistic way (Van der Hoef et al., 2008). However, collisions still require approximate models in general.

The DNS methods falls into two categories: boundary-fitted and non-boundary fitted. In the boundary-fitted techniques the generated computational grid fits the particle surface. This type of mesh is usually unstructured for complex surface geometries. This implies that mesh generation is computationally expensive and troublesome, especially when treating moving objects. The arbitrary Lagrangian–Eulerian (ALE) approach (Hu et al., 2001) locates in this group. This method loses its efficiency in case of 3D or in

* Corresponding author. Tel.: +49 3916712324.
E-mail address: amir.eshghinejadfard@ovgu.de (A. Eshghinejadfard).

http://dx.doi.org/10.1016/j.ijheatfluidflow.2016.04.002
S0142-727X(16)30112-6/

presence of numerous particles. Non-boundary fitted approaches are then easier to implement. The two main methods in this category are: the distributed Lagrange multiplier or fictitious domain method (DLM/FDM, see Glowinski et al., 1999) and the immersed boundary method (IBM). In this work, the latter approach is implemented. IBM was first introduced by Peskin (1977) in order to model the blood flow in the heart. In IBM, the fluid equations are discretized on a fixed Eulerian grid over the entire domain and the immersed boundary is discretized on a moving Lagrangian mesh. Among different approaches for the evaluation of the force term at each Lagrangian point in IBM the *direct forcing* method (Mohd-Yusof, 1997; Uhlmann, 2005) has obtained particular attention due to its ease of application.

In this paper, the lattice Boltzmann method (LBM) together with IBM is employed to simulate different particulate flows involving heat transfer. LBM originated from the concept of lattice gas cellular automata (LGCA). LGCA suffered from noise and it was not an easy task to extend it to three-dimensional (3D) cases. Therefore, the Boolean variable of LGCA was later replaced by a real-valued distribution function (McNamara and Zanetti, 1988) and the collision operator was replaced by a linear one (Higuera and Jimenez, 1989). Since then LBM has been widely applied to the simulation of different flow configurations including turbulent flows (Lammers et al., 2006), heat transfer (Mohamad and Kuzmin, 2010), nano-fluids (Nemati et al., 2010), blood flow (Krüger et al., 2013), to cite only a few. In LBM the fluid is considered as a set of fictitious particles. These particles that describe the fluid are allowed to move between lattice nodes or stay at rest. LBM has basically two steps: first, the collision of particles, which controls the relaxation toward equilibrium; and in the second step, the streaming of particles in which distribution functions are shifted to the neighboring lattice cells. The macroscopic flow properties such as density, velocity, and pressure can be retrieved from the collective behavior of the microscopic states of the particles including their location and velocity. Efficient parallelization, high computational efficiency and the absence of any elliptic Poisson equation for pressure made LBM increasingly popular. It has later been applied to the simulation of flow around particles using a bounce-back boundary condition (Ladd, 1994). Feng and Michaelides (2004) combined IBM and LBM and successfully simulated the sedimentation of a large number of particles in an enclosure. Finally, a direct-forcing based IB-LBM was proposed that overcame the drawbacks of the bounce-back method and was easier to implement (Feng and Michaelides, 2005; Dupuis et al., 2008).

The numerical simulation of particulate flows is already quite complex. This complexity increases even further when heat transfer must be taken into account. Although there has been increasing interest for corresponding studies in recent years, only few publications can be found. Gan et al. (2003) used the ALE-finite element method to model the motion of particles having heat transfer with surrounding fluid; however, this method was computationally expensive. Yu et al. (2006) extended the DLM/FDM to simulate two dimensional (2D) particulate flows with heat transfer. They considered particles of constant or varying temperature. Kim and Choi (2004) and Pacheco et al. (2005) applied IBM to the heat transfer between a fluid and stationary objects. Feng and Michaelides (2009) developed an IB finite-volume technique and applied this approach to particle-laden flows, where particles are moving. They verified their results by comparing them to those of Yu et al. (2006). They were also able to use this approach at $\rho_r C_r = 1$ (ρ_r: particle to fluid density ratio; C_r: particle to fluid specific heat ratio) and tackle the instability problems appearing around this threshold. However, they assumed a uniform temperature inside the particle, which is only valid for high thermal conductivities of the solid particle. Dan and Wachs (2010) used the DLM/FD method to model heat transfer problems in 3D with constant particle temperature. Kang and Hassan (2011) applied two types of thermal IB-LBM, a hybrid model and a simplified double-population method to simulate heat transfer between particles and the fluid. However, the simulation was limited to 2D and particles had a fixed temperature. Wachs (2011) studied the rising of 3D catalyst particles using a parallelized DNS-IB with a fictitious domain method. Deen et al. (2012) implemented DNS-IBM to study the heat transfer in both stationary beds and fluidized beds. Ström and Sasic (2013) used volume of fluid approach and modeled the motion of solid stationary and moving particles in the presence of heat transfer effects in both 2D and 3D domains. Weiwei (2014) developed a novel IBM under the framework of Navier–Stokes solver and applied it to non-isothermal flows in the presence of solid particles. Xia et al. (2015) modeled heat transfer from 3D moving spheres using a ghost-cell based IBM. Recently, Zhang et al. (2015) proposed a combined thermal LBM–IBM–DEM and simulated heat transfer between single and multiple particles with carrier fluid.

Therefore, the combination of thermal IBM with LBM is an interesting topic. In this work we aim to extend this approach to 3D cases in which spherical particles are moving while having heat exchange with surrounding fluid flow; the issue that has not been addressed by the above mentioned articles. The case of variable particle temperature will be studied as well; both in two- and three dimensional simulations. For this purpose, a force density and an energy density term are introduced into the LBM equations. These force and energy terms are evaluated through a direct-forcing and direct-heating IBM, respectively. The developed methodology has been implemented in the in-house LB software ALBORZ, described in Eshghinejadfard et al. (2016).

2. Model formulation

2.1. Thermal LBM

The LBM, which is a particle-based approach for solving fluid mechanics problems, was first proposed in the 1980s. It describes the evolution of a discretized particle distribution function, $f(\mathbf{x}, t, \xi)$, which represents the probability of finding a fluid particle with a certain velocity ξ at a certain location $\mathbf{x}$ and time t. In this method we do not need to solve directly the equations corresponding to macroscopic variables like velocity or pressure. Instead, the LBM operates at a mesoscopic level via the distribution functions f, which are simply summed up to obtain the macroscopic dynamics.

A discretization of the Boltzmann equation in time and space, and the conversion of the space of velocities $\{\xi\}$ into a finite set of velocities $\{\mathbf{c}_i\}$ within which the particles are allowed to move in the lattice, leads to the well-known lattice BGK model (Bhatnagar et al., 1954):

$$f_i(\mathbf{x}+\mathbf{c}_i\Delta t, t+\Delta t) - f_i(\mathbf{x},t) = -\frac{1}{\tau_f}\left[f_i(\mathbf{x},t) - f_i^{(eq)}(\mathbf{x},t)\right], \quad (1)$$

where f_i is the distribution function of particles moving with speed $\mathbf{c}_i$ and the right-hand side accounts for the single relaxation time (SRT) collision term. Here, τ_f is the dimensionless relaxation time and Δt is the time step. The equilibrium distribution function $f_i^{(eq)}(\mathbf{x},t)$ is obtained by using the Taylor series expansion of the Maxwell–Boltzmann distribution function with fluid density ρ_f and velocity $\mathbf{u}$ up to second order. It can be defined as:

$$f_i^{eq} = \omega_i \rho_f \left[1 + \frac{\mathbf{c}_i\cdot\mathbf{u}}{c_s^2} + \frac{(\mathbf{c}_i\cdot\mathbf{u})^2}{2c_s^4} - \frac{|\mathbf{u}|^2}{2c_s^2}\right], \quad (2)$$

where ω_i is the weight associated with the velocity $\mathbf{c}_i$, and the sound speed c_s is model-dependent and equal to $\Delta x/(\sqrt{3}\Delta t)$. In the present work, for the 2D and 3D flows, the nine-velocity (D2Q9) and nineteen-velocity models (D3Q19) are applied, respectively. In case of 2D simulations the D2Q9 model has the