International Journal of Heat and Mass Transfer 52 (2009) 777–786

Contents lists available at ScienceDirect

International Journal of Heat and Mass Transfer

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

Heat transfer in particulate flows with Direct Numerical Simulation (DNS)

Zhi-Gang Feng [a,*], Efstathios E. Michaelides [b]

[a] Department of Mechanical and Energy Engineering, University of North Texas, Denton TX 76207, USA
[b] Department of Mechanical Engineering, University of Texas at San Antonio, San Antonio, TX 78259, USA

ARTICLE INFO

Article history:
Received 6 December 2007
Received in revised form 2 July 2008
Available online 5 September 2008

Keywords:
Particulate flow
Solid and fluid interaction
Heat transfer
Immersed boundary method

ABSTRACT

A Direct Numerical Simulation (DNS) method has been developed to solve the heat transfer equations for the computation of thermal convection in particulate flows. This numerical method makes use of a finite difference method in combination with the Immersed Boundary (IB) method for treating the particulate phase. A regular Eulerian grid is used to solve the modified momentum and energy equations for the entire flow region simultaneously. In the region that is occupied by the solid particles, a second particle-based Lagrangian grid is used, which tracks particles, and a force density function or an energy density function is introduced to represent the momentum interaction or thermal interaction between particle and fluid. The numerical methods developed in this paper have been validated extensively by comparing the present simulation results with those obtained by others.

1. Introduction

Particulate flows take place in many natural and engineering processes, such as the sedimentation of sand particles in rivers; the evaporation and combustion of drops; the movement of blood cells in capillaries; the motion of catalyst particles in chemical reactors; and fluidization of solid fuels in reacting bed systems. In many of these cases, the multiphase flows not only involve the motion of particles and the momentum/mechanical interactions between particles and fluids, but also involve heat transfer between the two phases.

There are three common techniques that have been used extensively for the numerical simulations of particulate flows: one is the two-phase continuum model, which treats the solid and liquid phases as two fluids governed by separate momentum equations. The effect of fluid and particle interactions are mainly reflected in the apparent viscosity and the drag coefficient used in the governing equations. The second model is the discrete particle model. This model considers each particle as a point force and determines the position of particles by solving the Lagrangian equation of motion for every particle in the flow field. The force acting on each particle is approximated by an empirical drag force. The third approach, which has gained popularity in the recent years because of the improvement of computational power, is the Direct Numerical Simulation (DNS). This approach treats separately the solid particles and the fluid. The determination of the solid–fluid interaction is accomplished simultaneously via the solution of the Navier–Stokes equation for the fluid and the equations of motion for particles.

Several numerical schemes, which may be classified as DNS methods have been developed in the past decade: Feng et al. [1] studied the sedimentation of two-dimensional circular particles using the arbitrary Lagrangian–Eulerian (ALE) approach. They essentially solved the Navier–Stokes equations, subject to the no-slip boundary conditions on the surface of the particles. This method requires mesh-adapting, when the particles are moving, and it is not efficient in the simulations of systems with a large number of particles. A more efficient approach has been developed recently, which is based on the Immersed Boundary (IB) method and does not require re-meshing. The IB method was introduced by Peskin [2] to account for fluid–solid interactions. This method uses a fixed Cartesian mesh for the fluid, and a moving, Lagrangian grid for the particle. A force density function is usually defined in conjunction with this method to represent the effect of the particles on the fluid. Various schemes have been proposed to compute the force density functions [3–10]. Among them, the direct-forcing scheme introduced by Mohd-Yusof [11,12] has been a favorite choice of many, because of its simplicity [6,8–10] and ease in implementation. In parallel, Glowinski and his collaborators [13–16] developed a finite-element based fictitious domain approach for solving particulate flow problems. They have used Lagrangian multipliers to enforce the rigid body motion.

Other numerical approaches that do not require re-meshing has been proposed by Kalthoff et al. [17] and Zhang and Prosperetti [18,19]. Such methods incorporate an analytical Stokes flow solution for the region adjacent to the particle surface with some of the parameters determined by matching the outer flow conditions. The Lattice Boltzmann Method (LBM) was used by many since Ladd successfully applied it to particulate flows [20,21] using the

* Corresponding author. Tel.: +1 940 5652078; fax: +1 940 369 8675.
E-mail addresses: feng@unt.edu (Z.-G. Feng), stathis.michaelides@utsa.edu (E.E. Michaelides).

0017-9310/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijheatmasstransfer.2008.07.023

Nomenclature

A	radius of a spherical particle
Bi	Biot number
c	specific heat
f	force density
g	gravitational acceleration
Gr	Grashoff number
I_P	moment of inertia of a particle
k	thermal conductivity
L	length variable
n	normal outward vector
p	Pressure
Pe	Peclet number
Pr	Prandtl number
q	energy source
r	radius of a sphere
Re	Reynolds number
s, S	area
T	temperature
t	time variable
u, U	velocity
v, V	volume
x	position coordinate
y	position coordinate

Greek symbols

ρ	density
λ	energy density function
Ω	entire domain
β	the thermal expansion coefficient
$\sum S_i$	region occupied by the particles
μ	dynamic viscosity
σ	shear stress tensor
ω	angular velocity of the particle
Θ	dimensionless temperature
ε	pre-defined tolerance

Subscripts/superscripts

$'$	dimensionless variable
$\rightarrow$	a vector
0	value of ambient fluid
f	property related to fluid
k	value at the kth iteration
L	value at Lagrangian node
n	value at the nth time step
p	property related to particle
r	relative value
ref	reference value

"bounce-back" method. The LBM has gained popularity for its computational efficiency and easy implementation in numerical parallelization. Feng and Michaelides [6,8,9] combined the LBM and IB approach to overcome some of the drawbacks of the bounce-back scheme in treating boundary conditions at the surface of the particles.

DNS with heat transfer in multiphase systems has not received much attention yet. Among the limited studies reported in the literature, Gan et al. [22,23] studied the two-dimensional interaction of heat transfer between the particles and their surrounding fluid by using the ALE finite-element method. Yu et al. [24] employed the fictitious domain method to study two-dimensional particulate flow with heat convection. They resolved the heat interaction between the flow and particles for various temperature boundary conditions on the particles by introducing Lagrangian multipliers.

The IB method has been applied for the study of heat transfer problems involving stationary boundaries. Kim and Choi [25] proposed an IB method for the solution of two-dimensional heat flow problems with complex geometries. Also Pacheco et al. [26] presented an IB finite-volume method to study the heat transfer and fluid flow problems with non-staggered grids. However, there is no study available on the use of IB for the heat transfer in particulate flow. Though the fictitious domain method and immersed boundary method use similar concepts, the implementation details are quite different, and IB is more straightforward and easier to apply.

The objective of this paper is to develop a relatively simple numerical method that utilizes the IB technique for the solution of the thermal interaction between particles and fluid. The heat and mass transfer is an important aspect of particulate-flows. Industrial applications can be found in the safe handling of radioactive waste materials, the transport of mixtures in food processing and the fluidization of solid fuels in bioreactors. In many of these applications, solid phase has relatively much higher conductivity compared to fluid or gas phase; and we can assume particles to have uniform temperature (Biot number $Bi = 0$). In this paper, we will focus on such cases; the Bousinesq assumption is also adopted for the properties of the fluid. Two types of grids are used to solve the particle–fluid interactions: the first is a fixed Eulerian grid for the entire flow domain and the second is a moving Lagrangian grid for each particle. The modified momentum and energy equations are solved only on the Eulerian grid. The no-slip boundary condition on the particle surface and the rigid body motion of particles are enforced only in the Lagrangian grid. To account for the fluid–particle interactions, a force density and an energy density function are introduced into the momentum and energy equation, respectively. The temperature boundary condition is enforced by adding an energy density function into the energy equation. These density functions represent the total effects of the net momentum and energy exchanges between particles and fluid. A simple and very effective scheme suitable for the finite-difference method is also presented to resolve the issue of the instability that is caused by the motion of very light particles. This numerical method is implemented on staggered grids using an explicit finite difference approach. By applying the present method, we studied several two-dimensional particulate flow problems and obtained results that demonstrate the validity, accuracy and robustness of this method. More examples using the DNS method developed in this paper can also be found in a recent paper by the authors [27]. It must be pointed out that the IB and the fictitious domain method have a similar concept; however, compared to the fictitious domain method, the current IB based approach provides a much simple and effective way in solving heat and mass transfer in particulate flows with high solid–fluid thermal conductivity.

2. Problem description

We consider a particulate flow system composed of several circular rigid particles suspended in an incompressible Newtonian fluid, as shown in Fig. 1. The entire computational domain, Ω, is composed by the fluid region, L, and the solid particle region, $\sum S_i (S_1 + S_2$ in the figure). The domain is surrounded by a boundary, Γ. The boundary/surface of the ith particle S_i is denoted by ∂S_i. The particles are assumed to have uniform temperature, which implies that the Biot number is close to zero, a valid approximation

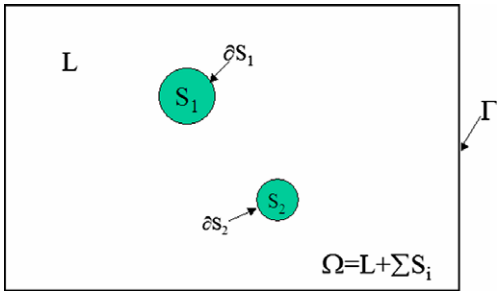

Fig. 1. Conceptual model of two circular particles suspended in a fluid.

for small, solid particles with high thermal conductivity. Temperature gradients in the fluid cause the variation of fluid properties. For simplicity, we employ the Boussinesq approximation for the fluid field and study the momentum and heat interactions between the particles and the fluid.

2.1. Momentum interaction between the fluid and the particles

The concept of the immersed boundary [2] enables one to describe the entire domain Ω, that is occupied by the fluid and solid particles, by using the following modified Navier–Stokes equations:

$$\rho_f \frac{\partial \vec{u}}{\partial t} + \rho_f \vec{u} \bullet \nabla \vec{u} = -\vec{\nabla} p + \mu_f \nabla^2 \vec{u} + \vec{f}, \quad \vec{x} \in \Omega \tag{1}$$

$$\vec{\nabla} \bullet \vec{u} = 0, \quad \vec{x} \in \Omega \tag{2}$$

where ρ_f and μ_f are the fluid density and dynamic viscosity of the fluid, respectively; $\boldsymbol{u}$ is the velocity field developed in the fluid; and p is the fluid pressure. It is evident that, compared to the usual form of the Navier–Stokes equation, Eq. (1) contains an additional term, $\boldsymbol{f}$, which is called "the force density field." This force density field is equal to zero in the regions occupied by the fluid. It is generally believed that the interior field of the solid particles plays an insignificant role on the motion of the particles *per se*. For example, Uhlmann [10] who conducted two-dimensional test computations of particle sedimentation concluded that "locating force points throughout the particle volume does not lead to significantly different results." Hence, for a pure solid–fluid momentum interaction problem, the force density may only be prescribed over the surface of the particles and may be computed by one of several schemes that were developed for this purpose, such as the spring model [5,6], or the direct-forcing scheme [8–10]. In the case of solid particles, it was proven that the direct-forcing scheme is more convenient than the spring model scheme because it does not introduce additional empirical parameters. Here we adopt the direct-forcing scheme for the computation of the force density.

Obviously, the momentum interaction between fluid and particles results in the motion of particles, which may be described as follows:

A. Translational motion:

$$\rho_p V_p \frac{\mathrm{d}\vec{U}_P}{\mathrm{d}t} = \rho_f \oint_{\partial S} \tilde{\sigma} \bullet \mathrm{d}\vec{s} + \int_S (\rho_p - \rho_f)\vec{g}\mathrm{d}v. \tag{3}$$

B. Rotational motion:

$$I_p \frac{\mathrm{d}\vec{\omega}_p}{\mathrm{d}t} = \rho_f \oint_{\partial S} (\vec{x} - \vec{x}_p) \times (\tilde{\sigma} \bullet \mathrm{d}\vec{s}), \tag{4}$$

where ρ_p, V_p, I_p, $\boldsymbol{U}_p$ and $\boldsymbol{\omega}_p$ are the particle's density, volume, moment of inertia, translational velocity, and angular velocity, respectively; $\boldsymbol{x}_p$ is the position vector of the center of mass of the particle; and $\boldsymbol{\sigma}$ is the fluid stress tensor.

The first term in the right-hand side of Eq. (3) is the interaction force between the particle and the surrounding fluid. The second term is the buoyancy force. The term in the right-hand side of Eq. (4) is the angular momentum impulse induced by the fluid–particle interaction. If the force density is prescribed for the entire region that represents the solid, one may apply the Cauchy principle to write the surface integrals as follows:

$$\rho_f \oint_{\partial S} \tilde{\sigma} \bullet \mathrm{d}\vec{s} = \rho_f \int_S \vec{f}\mathrm{d}v + \frac{\mathrm{d}}{\mathrm{d}t} \int_S \rho_f \vec{u}\mathrm{d}v, \tag{5}$$

and

$$\rho_f \oint_{\partial S} (\vec{x} - \vec{x}_p) \times (\tilde{\sigma} \bullet \mathrm{d}\vec{s}) = -\rho_f \int_{\partial S} (\vec{x} - \vec{x}_p) \times \vec{f}\mathrm{d}v + \rho_f \frac{\mathrm{d}}{\mathrm{d}t} \int_{\partial S} (\vec{x} - \vec{x}_p) \times \vec{u}\mathrm{d}v. \tag{6}$$

By using the rigid body motion conditions, it can be shown that the time derivatives of the volume integrals in the above two equations yield the following contributions of the internal fluid mass, as described by Glowinski [13] and Uhlmann [7]:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_S \rho_f \vec{u}\mathrm{d}v = \rho_f V_p \frac{\mathrm{d}U_p}{\mathrm{d}t}, \tag{7}$$

and

$$\rho_f \int_S (\vec{x} - \vec{x}_p) \times \vec{f}\mathrm{d}v = \rho_f \frac{I_p}{\rho_p} \frac{\mathrm{d}\vec{\varpi}_p}{\mathrm{d}t}. \tag{8}$$

Substituting these results into Eqs. (3) and (4) one obtains:

$$(\rho_p - \rho_f) V_P \frac{\mathrm{d}\vec{U}_P}{\mathrm{d}t} = \rho_f \int_S \vec{f}\mathrm{d}v + (\rho_p - \rho_f) V_p \vec{g}, \tag{9}$$

and

$$I_P \left(1 - \frac{\rho_p}{\rho_f}\right) \frac{\mathrm{d}\vec{\omega}_P}{\mathrm{d}t} = -\rho_f \int_{Vp} (\vec{x} - \vec{x}_p) \times \vec{f}\mathrm{d}\upsilon. \tag{10}$$

The force density may be determined using the direct forcing scheme to yield the following expression:

$$\vec{f} = \rho_f \frac{\partial \vec{u}}{\partial t} + \rho_f \vec{u} \bullet \nabla \vec{u} + \vec{\nabla} p - \mu_f \nabla^2 \vec{u}, \quad \vec{x} \in \sum_{S_i}, \tag{11}$$

where $\vec{u}$ is the velocity field in the solid region. Inside the particle, the motion of a rigid body condition determines the velocity field at any point $\vec{x}$:

$$\vec{u} = \vec{U}_P + \vec{\omega}_P \times (\vec{x} - \vec{x}_P). \tag{12}$$

2.2. Energy interactions between fluid and particles

We consider that particles have uniform temperatures. This is a valid assumption if the thermal conductivity ratio between the solid particles and the fluid is high, or equivalently, if the Biot number is very small: $Bi \ll 1$. An example in Section 4.3 will show that reasonable results could be obtained even for thermal conductivity ratio of 5. However, each particle's temperature is transient and is denoted by the function $T_p(t)$. This temperature is determined by the energy balance of each particle. At the surface of the particles, the fluid and particle temperatures are equal:

$$T_f(\vec{x}, t) = T_p(t), \quad \vec{x} \in \partial S. \tag{13}$$

Similar to the momentum interaction, which was described in Section 2.1, the temperature field of the entire region is governed by the following modified energy equation:

$$\rho_f c_f \frac{\partial T}{\partial t} + \rho_f c_f \vec{u} \bullet \nabla T = k_f \nabla^2 T + q_p + \lambda, \quad \vec{x} \in \Omega, \tag{14}$$

where c_f and k_f are the specific heat and thermal conductivity of fluid and q_p represents any energy sources inside the particle. The additional energy density term λ is added to enforce the temperature field in the region occupied by the particles. As in the case of the force balance equation, this term is zero in the field occupied by the fluid. In the solid region this term is computed using the following equation:

$$\lambda = \rho_f c_f \frac{\partial T}{\partial t} + \rho_f c_f \vec{u} \bullet \nabla T - k_f \nabla^2 T - q_p, \quad \vec{x} \in \sum S_i, \tag{15}$$

where T is the temperature of the particle. The numerical technique for enforcing the condition of $T = T_p(t)$ in the solid region will be discussed in a subsequent section.

The transient temperature of the particles, $T_p(t)$, may be determined by solving the following differential equation, which is obtained from the energy balance for the particle:

$$\rho_p V_p c_p \frac{\mathrm{d}T_p}{\mathrm{d}t} = \oint_{\partial S} k_f \vec{\nabla} T_f \bullet \vec{n}\, \mathrm{d}s + \int_S q_p \mathrm{d}v \tag{16}$$

where $\vec{n}$ is the outward normal vector. Similar to the treatment of equations of particle motion in (3) and (4), the energy balance Eq. (16) can be converted into the following:

$$(\rho_p c_p - \rho_f c_f) V_p \frac{\mathrm{d}T_p}{\mathrm{d}t} = \int_S q_p \mathrm{d}v. \tag{17}$$

For particles that have different temperatures than the fluid, the presence of this term will create a temperature gradient within the fluid, which would modify the fluid properties. This effect may be computed in a conceptually easy manner by using empirical equations for the properties of the fluid at every point of the grid that represents the fluid. However, this is a computationally intensive method. For relatively small temperature difference between the particles and fluid, the Boussinesq approximation has been often used successfully for the coupling of the energy and momentum equations. Hence, the momentum Eq. (1) may be rewritten as follows:

$$\rho_{f0} \frac{\partial \vec{u}}{\partial t} + \rho_{f0} \vec{u} \bullet \nabla \vec{u} = -\vec{\nabla} p + \mu_f \nabla^2 \vec{u} + \beta_f (T - T_{f0}) \vec{g} + \vec{f}, \quad \vec{x} \in \Omega \tag{18}$$

where the subscript 0 denotes values of the undisturbed fluid and β_f is the thermal expansion coefficient of the fluid.

2.3. Governing equations in dimensionless form

The numerical solution and the numerical algorithm is facilitated by the following dimensionless variables:

$$u' = \frac{u}{U_{ref}}, \quad l' = \frac{l}{L_{ref}}, \quad t' = \frac{t}{L_{ref}/U_{ref}}, p' = \frac{p}{\rho_{f0} U_{ref}^2}, \quad f' = \frac{f L_{ref}}{\rho_{f0} U_{ref}^2}, \tag{19}$$

for the momentum equation, and,

$$\Theta = \frac{T - T_{f0}}{T_{ref}}, \quad q' = \frac{q L_{ref}}{\rho_{f0} c_f T_{ref} U_{ref}}, \quad \lambda' = \frac{\lambda L_{ref}}{\rho_{f0} c_f T_{ref} U_{ref}} \tag{20}$$

for the energy equation. U_{ref}, L_{ref} and T_{ref} are the reference velocity, length-scale and temperature, respectively. The convective time-scale (L_{ref}/U_{ref}) is used as the reference time-scale for the problem. T_{f0} is the ambient fluid temperature, and in most cases, T_{ref} is chosen to be the initial temperature difference between the particles and the ambient fluid. We also define the following dimensionless parameters:

$$\rho_r = \frac{\rho_p}{\rho_{f0}}, \quad c_r = \frac{c_p}{c_f}, \tag{21}$$

$$Re = \frac{\rho_f u_{ref} l_{ref}}{\mu_f}, \quad Pr = \frac{\mu_f c_f}{k_f}, \quad Pe = Pr\ Re,$$

$$Gr = \frac{g \beta L_{ref}}{U_{ref}^2} (T_{p0} - T_{f0}) \left(\frac{\rho_f U_{ref} L_{ref}}{\mu_f} \right), \tag{22}$$

where Re is the Reynolds number; Pr is the Prandtl number; Pe is the Peclet number; and Gr is the Grashoff number.

In the sections that follow, the equations are simplified by omitting the superscript prime from all the variables. It must be remembered, however, that the variables in the equations that follow are dimensionless. Hence, the following set of dimensionless governing equations may be obtained:

A. For the velocity field:

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \bullet \vec{\nabla} \vec{u} = -\vec{\nabla} p + \frac{1}{Re} \nabla^2 \vec{u} + \frac{Gr}{Re^2} \vec{e}_g + \vec{f}, \quad \vec{x} \in \Omega, \tag{23}$$

where $\mathbf{e}_g$ is the unit vector in the direction of gravitational acceleration.

B. For the force density field:

$$\vec{f} = \frac{\partial \vec{u}}{\partial t} + \vec{u} \bullet \vec{\nabla} \vec{u} + \vec{\nabla} p - \frac{1}{Re} \nabla^2 \vec{u}, \quad \vec{x} \in \sum S_i. \tag{24}$$

C. Continuity equation:

$$\vec{\nabla} \bullet \vec{u} = 0, \quad \vec{x} \in \Omega. \tag{25}$$

D. For the velocity field inside the solid particle region:

$$\vec{u} = \vec{U}_P + \vec{\omega}_P \times (\vec{x} - \vec{x}_P). \tag{26}$$

E. For the temperature field:

$$\text{Energy equation}: \frac{\partial \Theta}{\partial t} + \vec{u} \bullet \vec{\nabla} \Theta = \frac{1}{Pe} \nabla^2 \Theta + q + \lambda, \quad \vec{x} \in \Omega \tag{27}$$

$$\text{Energy density function}: \lambda = \frac{\partial \Theta}{\partial t} + \vec{u} \bullet \vec{\nabla} \Theta - \frac{1}{Pe} \nabla^2 \Theta - q, \quad \vec{x} \in \sum S_i \tag{28}$$

F. For the motion of the particles:

$$(\rho_r - 1) V_p \frac{\mathrm{d}\vec{U}_p}{\mathrm{d}t} = \int_S \vec{f} \mathrm{d}v + (\rho_r - 1) V_p \frac{g L_{ref}}{U_{ref}^2} \vec{e}_g \tag{29}$$

$$\frac{I_p}{\rho_r} (\rho_r - 1) \frac{\mathrm{d}\vec{\omega}_p}{\mathrm{d}t} = -\int_S (\vec{x} - \vec{x}_p) \times \vec{f} \mathrm{d}v \tag{30}$$

In two dimensions we have the following closure expressions: $V_p = \pi r^2$, $I_p = \frac{1}{2} \rho_r \pi r^4$.

G. For the temperature of the particles:

$$(\rho_r c_r - 1) V_p \frac{\mathrm{d}\Theta_p}{\mathrm{d}t} = \int_S (\lambda + q_p) \mathrm{d}v. \tag{31}$$

The uniformity of temperature in the solid region is enforced by the condition:

$$\Theta = \Theta_P(t). \tag{32}$$

The initial conditions of the above set of equations are:

$$\vec{U}_P(t=0) = \vec{U}_{P0}; \vec{\omega}_P(t=0) = \omega_{P0}; \vec{u}(t=0) = \vec{u}_0; \Theta(t=0) = \Theta_0;$$
$$\Theta_p(t=0) = \Theta_{p0}. \tag{33}$$

It must be pointed out that, because of the mass-energy transfer analogy, the governing equations of the equivalent mass transfer problem are identical in form to those of the energy transfer problem. Hence, the numerical scheme that will be described in the next section may be used for the solution of the coupled mass transfer and particle motion problem without any significant modifications.

3. Numerical implementation

Since the velocity and temperature fields are coupled, the governing equations should be solved at each time step. The following steps are used in the solution procedure:

1. Compute the velocity in the solid region using Eq. (26).
2. Compute the force density term using Eq. (24).
3. Solve the modified Navier–Stokes Eq. (23) for the entire domain.
4. Compute the energy density term using Eq. (28).
5. Solve the temperature field of the entire computational domain using Eq. (27).
6. Solve the equations of motions (29) and (30) to obtain the translational and angular velocities of the particles.
7. Determinate the particle temperature using Eq. (31) if particle temperature changes with time.

The process outlined in steps 1–7 completes one time step of the computations. We have implemented this method to a simple two-dimensional problem with circular particles suspended in the flow and present the results in the following sections.

3.1. Solution of the velocity field

The procedure for solving the momentum equation with the direct-forcing scheme is well documented [8–10]. Most of the methods used in these references have implemented the no-slip boundary condition without enforcing the rigidity of solid particles. In the method described here, the rigidity of the solid particles is enforced at the surface and in the interior of the particles. A staggered grid is used as the Eulerian grid for the entire flow domain as shown in Fig. 2. In order to improve the computational accuracy, both the velocity components and the force density component are defined at the same locations, at the center of the faces of each cell. The pressure, temperature and energy density are defined at the center of the cell.

In order to solve the momentum and energy equations, the differential equations were discretized and solved by using a fully explicit finite difference scheme, though it can be readily extended to other implicit schemes. The procedure for the implementation of this explicit scheme is as follows:

1. Solve for the provisional velocity field, $\boldsymbol{u}^*$:

$$\frac{\vec{u}^* - \vec{u}^n}{\Delta t} + \vec{u}^n \bullet \vec{\nabla}\vec{u}^n = \frac{1}{Re}\nabla^2\vec{u}^n + \frac{Gr}{Re^2}\Theta^n\vec{e}_g + \vec{f}^n \tag{34}$$

2. Solve for the pressure field p,

$$\nabla^2 p^{n+1} = \frac{\vec{\nabla} \bullet \vec{u}^*}{\Delta t} \tag{35}$$

3. Update the velocity field:

$$\vec{u}^{n+1} = \vec{u}^* - \Delta t\vec{\nabla}p^{n+1}. \tag{36}$$

The force density term, $\boldsymbol{f}$, has been calculated using the direct-forcing method by enforcing the velocity in the solid regions to conform to the rigid body motion. A Lagrangian grid is used to enforce the rigid-body motion inside the region of the particles and the constraint of the equality of temperatures at the surface of the particles, as shown in Fig. 3. The number of cells used depends on the size of the particles. Fig. 3 shows a total of 76 cells used for a circular particle with radius equal to 5 times the distance δx (δx is the discretization space for Eulerian grid). The forcing term is computed at the midpoint of the cells, which are denoted by dots in Fig. 3. A similar Lagrangian grid has also been used by Uhlmann [29]. The usual practice for the choice of the size of the Lagrangian cells is to make the volume of a Lagrangian cell approximately equal to the volume of a regular Eulerian cell. When enforcing the temperature field for the solid region, one might only use the exterior cells since the interior temperature field is uniform and does not affect the fluid temperature.

The force density term is computed at each Lagrangian node as follows:

$$\vec{f}_L^n = \frac{\vec{U}_L^n - \vec{u}_L^n}{\Delta t} + \vec{u}_L^n \bullet \vec{\nabla}\vec{u}_L^n - \frac{1}{Re}\nabla^2\vec{u}_L^n + \vec{\nabla}p_L^n - \frac{Gr}{Re^2}\Theta_L^n\vec{e}_g. \tag{37}$$

The subscript L indicates the values at Lagrangian nodes. The values of velocity, pressure and temperature at these nodes are obtained by a bi-linear interpolation using the values of neighboring Eulerian nodes. U_L^n is the velocity of a Lagrangian node located at $\boldsymbol{x}$ and calculated by the expression: $U_L^n = \vec{U}_P^n + \vec{\omega}_P \times (\vec{x} - \vec{x}_P)$. While solving for the velocity field on the Eulerian grid using Eqs. (34) through (36), the force density is distributed from the Lagrangian nodes to the Eulerian nodes using the delta function, as described by Feng and Michaelides [8].

The fluid temperature field, Θ^{n+1}, is also solved explicitly,

$$\frac{\Theta^{n+1} - \Theta^n}{\Delta t} + \vec{u}^n \bullet \vec{\nabla}\Theta^n = \frac{1}{Pe}\nabla^2\Theta^n + q^n + \lambda_k^n. \tag{38}$$

To force the temperature in the solid region or solid boundary to be equal to the particle temperature Θ_P^n, the energy density is computed using the following expression on the Lagrangian nodes,

$$\lambda^n = \frac{\Theta_p^n - \Theta_L^n}{\Delta t} + \vec{u}^n \bullet \vec{\nabla}\Theta_L^n - \frac{1}{Pe}\nabla^2\Theta_L^n - q^n \tag{39}$$

where Θ_P is the particle temperature. The particle temperature is calculated by applying the energy balance equation,

$$(\rho_r c_r - 1)V_p\frac{\Theta_p^n - \Theta_p^{n-1}}{\delta t} = \int_S (\lambda + q_p)\mathrm{d}v \tag{40}$$

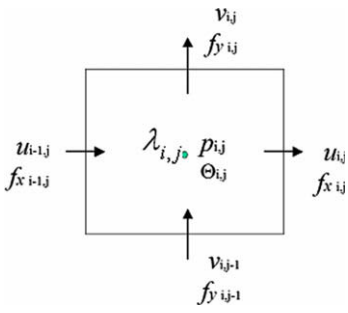

Fig. 2. The staggered Eulerian grid used for solving fluid velocity and temperature.

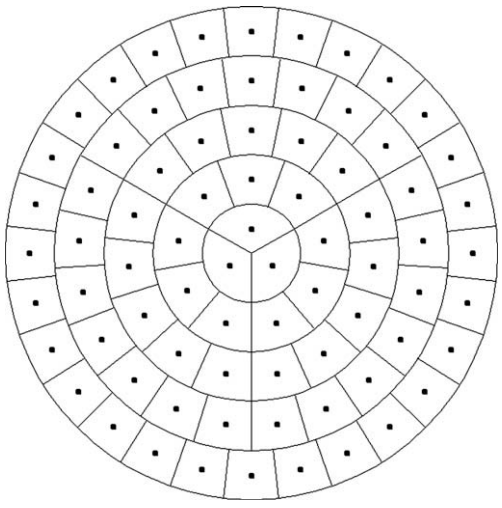

Fig. 3. Lagrangian grid for a circular cylinder of radius equal to 5 δx.

The temperature and velocity at each Lagrangian node are obtained by using a bi-linear interpolation with reference points its neighboring Eulerian nodes. The newly calculated energy density is then distributed into the Lagrangian nodes using a delta function.

3.2. Motion of particles

The equations of particulate motion are Eqs. (29) and (30), and may be discretized as follows:

$$(\rho_r - 1)V_r \frac{\vec{U}_p^{n+1} - \vec{U}_p^n}{\delta t} = \sum_L \vec{f}_L^n \delta V_L + (\rho_r - 1)V_P \frac{gL_{ref}}{U_{ref}^2}\vec{e}_g \tag{41}$$

$$I_P(\rho_r - 1)\frac{\vec{\omega}_p^{n+1} - \omega_p^n}{\delta t} = -\sum_L (\vec{x}_L - \vec{x}_p^n) \times \vec{f}_L^n \delta V_L. \tag{42}$$

The summation is carried out over all the Lagrangian nodes within the particulate region. However, we found from the implementation of this method that the solution becomes unstable when the particle–fluid density ratio is small. This has also been observed by other authors [10,20]. Uhlmann [7] found that with an implicit solver of the Navier–Stokes equations, he was able to achieve a stable solution at $\rho_r = 1.05$. To overcome this problem, one method is to commence with the dimensionless form of Eqs. (3) and (4) and to evaluate the interaction force and the angular impulse directly by performing the numerical integration over the particle boundary. We propose a simple, yet very effective, alternative approach to overcome the instability issue [9]. We discretized the Eqs. (29) and (30) by the following technique:

$$\rho_r V_p \frac{\vec{U}_p^{n+1} - \vec{U}_p^n}{\delta t} = \sum_L \vec{f}_L^n \delta V_L + V_p \vec{a}_p^n + (\rho_r - 1)V_P \frac{gL_{ref}}{U_{ref}^2}\vec{e}_g, \tag{43}$$

and

$$\rho_r I_P \frac{\vec{\omega}_p^{n+1} - \vec{\omega}_p^n}{\delta t} = -\sum_L (\vec{x}_L - \vec{x}_p^n) \times \vec{f}_L^n \delta V_L + I_P \vec{\alpha}_P^n, \tag{44}$$

where $\vec{a}_p$ and $\vec{\alpha}_p$ are the translational acceleration and angular acceleration, respectively. These quantities are evaluated explicitly:

$$\vec{a}_p^n = \frac{\vec{U}_p^n - \vec{U}_p^{n-1}}{\delta t}, \quad \vec{\alpha}_p^n = \frac{\vec{\omega}_p^n - \vec{\omega}_p^{n-1}}{\delta t}. \tag{45}$$

This approach works well even for light particles with $\rho_r = 1.00232$, as it will become apparent in the results of Section 4.2. In fact, it works for particles of the same density of fluid, as demonstrated in an example in Section 4.3. It must be pointed out that the instability issue does not arise in some fully implicit schemes, such as those formulations used in Refs. [14,16].

3.3. Temperature of particles

It is obvious that the particle energy balance Eq. (40) will be of a singular form when $\rho_r c_r$ equals to 1; and instability arises as $(1 - \rho_r c_r)$ becomes small. To resolve this issue, we discretize Eq. (40) as follows:

$$\rho_r V_p c_r \frac{\Theta_p^{n+1} - \Theta_p^n}{\delta t} = \sum_L q_L^n \delta V_L + V_p \left(\frac{d\Theta}{dt}\right)_p^n + \int_S q_p dv \tag{46}$$

and the time rate of particle temperature is evaluated by

$$\left(\frac{d\Theta}{dt}\right)_p^n = \frac{\Theta_p^n - \Theta_p^{n-1}}{\delta t}. \tag{47}$$

Such treatment allows us to simulate the extreme case of $\rho_r c_r = 1$ successfully, as shown in Section 4.3.

A collision algorithm is also necessary to prevent the particles from overlapping and from stopping the simulation for a system of interacting particles. In the present work, we adopted the same collision scheme as the one we have successfully used in our previous studies [8].

4. Numerical simulation results

We conducted a series of simulations to demonstrate the validity and robustness of the proposed numerical method. The first case we considered is the natural convection of a stationary heated circular cylinder, placed in a square duct; the second case is the sedimentation of a cold particle in an infinite channel; the third case is the motion of a catalyst particle in an enclosure in which the temperature of the particle varies with time; finally, the sedimentation of 56 particles in a fluid has been investigated.

4.1. A circular cylinder placed eccentrically in a square enclosure

The problem of natural convection of a circular cylinder placed eccentrically in a square enclosure has been studied extensively [24,26]. This fundamental problem has been investigated experimentally by many and numerically by Demirdzic et al. [28]. A schematic illustration of the problem is shown in Fig. 4, where a heated cylinder at rest is placed eccentrically in a square enclosure filled with a Newtonian fluid. The cylinder has a higher constant temperature at its boundary compared to the ambient temperature of the fluid and, as a result, natural convection occurs. We studied the heat convection from the cylinder at steady state and chose the length of the square sides to be the reference length. The parameters of the computation are as follows: the diameter of cylinder is $d = 0.4$, the eccentric distance is $\varepsilon = 0.1$, the dimensionless temperatures are 1 for the cylinder boundary and 0 for the ambient fluid, the adiabatic boundary conditions are used at the top and bottom walls, and constant (0) temperature is assumed on both the left and right side walls. The dimensionless flow numbers are: Reynolds number, $Re = 0.1$; Prandtl number $Pr = 10$; and Grashoff number $Gr = 100{,}000$.

Fig. 5 shows the local Nusselt numbers along one of the side walls computed by the present method and by Pacheco et al. [26] using a 200×200 grid. The simulation results of the present method have been obtained using a coarser 100×100 grid. It is apparent that the results agree very well with the results by Pacheco et al. [26]. The small discrepancy close to the wall ($x = 0.5$) is most likely due to the coarser grid used in this simulation.

The fluid velocity vector and isotherms of this problem are plotted in Fig. 6. One may observe two relatively large circulation regions on both the left and the right sides of the cavity that sweep in the upper area of the flow. These two circulation regions

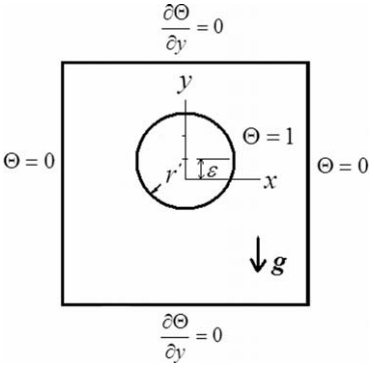

Fig. 4. Schematic diagram of a circular cylinder placed eccentrically in a square enclosure.

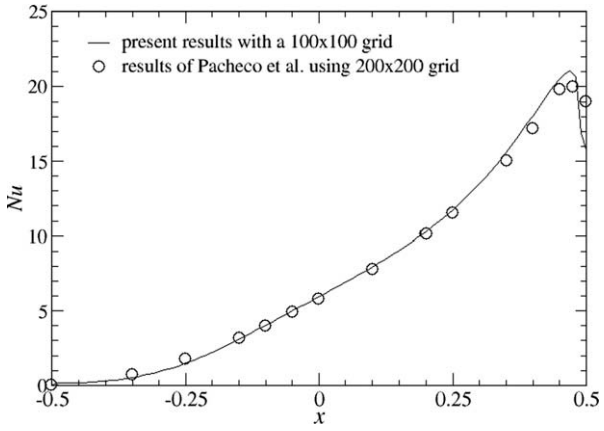

Fig. 5. Local Nusselt number along the side wall for $Re = 0.1$, $Pr = 10$ and $Gr = 10^5$.

dominate the flow motion and create a strong convection current that causes higher temperature to be built in the region near the top wall. In the region near the bottom wall, the convection is weak, signifying that cooling is mainly caused by conduction.

4.2. A particle with fixed temperature settling in an infinite channel

The problem of the sedimentation of a cold particle with constant temperature settling between two infinite vertical parallel planes was first studied by Gan et al. [22] using a boundary-fitted ALE method, and then was revisited by Yu et al. [24] as a validation problem for their fictitious domain method. These researchers found that the lateral equilibrium position of the particle depends strongly on the Grashoff number. The motion of the particle ranged from settling steadily along the centerline of the channel to regular oscillations about the centerline.

We chose the diameter of the particle as the reference length. The width of the channel is 4. The height of the computational domain is 32, but this domain is shifting while the particle is settling, thus simulating a flow in an infinite channel. We chose $U_{ref} = \sqrt{\pi a(\rho_r - 1)g}$ as the reference velocity. For the flow parameters we chose: $Re = 40.5$, $Pr = 0.7$, $\rho_r = 1.00232$, and $c_r = 1$. These are the same as used by Yu et al. [24]. The dimensionless temperature is equal to -1 on the particle boundary and initially 0 throughout the fluid domain. The particle is initially placed at one and half diameters off the centerline. An 80×480 grid is used for all the simulations; the time step is chosen to ensure that CFL number is less than 0.2.

Fig. 7 shows the lateral positions of the particle at $Gr = 0, 100, 564, 1000, 2000, 2500$ and 4500, respectively. It is observed that the particle experiences a weak oscillation around its centerline at $Gr = 564$ with an amplitude well below 5% of the particle diameter. However, at $Gr = 4500$, the particle encounters intensive vibrations with amplitudes about 1.4 times its diameter. The velocity components at $Gr = 4500$ shows the strong but regular oscillations. These results confirm the flow regimes that were reported by Yu et al. [24].

Table 1 lists the lateral equilibrium positions for $Gr = 1000$ and 2000, and the amplitude of lateral oscillation for $Gr = 4500$. These values are measured with respect to the centerline of the channel. Our results show almost perfect agreement compared to those obtained by Yu et al. [24].

The simulation results presented in this section not only show that our method works for constant particle temperature; they also demonstrate that the updating scheme for particle motion outlined in Eqs. (43) and (44) is stable even for a particle–fluid density ratio $\rho_r = 1.00232$.

4.3. Motion of a catalyst particle with freely varied temperature

The DNS method developed in this paper can be also used to solve particle–fluid heat transfer problems where the temperature of particles changes with time, provided that the particle–fluid

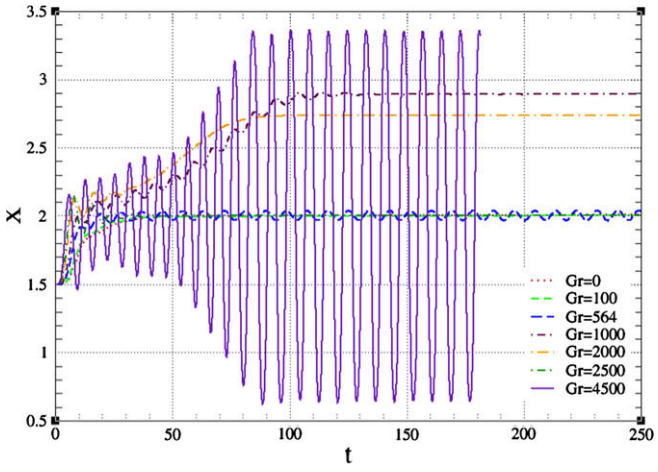

Fig. 7. The lateral position of a cold particle setting in an infinite channel at different Grashoff numbers.

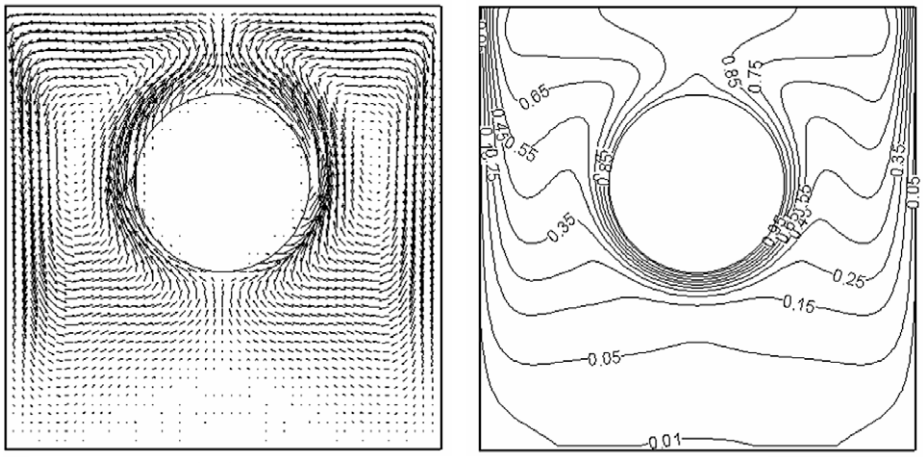

Fig. 6. Velocity vector map and isotherms for $Re = 0.1$, $Pr = 10$ and $Gr = 10^5$. (a) Velocity vector; (b) isotherms.

Table 1
Equilibrium position or amplitude of oscillation with respect to the centerline

Gr	Present	Yu et al. [24]
1000	1.90	1.89
2000	1.73	1.74
4500	1.35	1.33

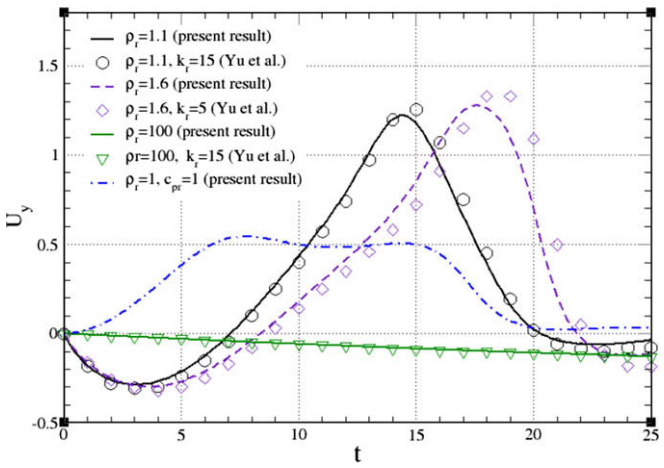

Fig. 8. Evolution of the vertical velocity of a catalyst particle moving in an enclosure.

thermal conductivity ratio is high. Yu et al. [24] studied the motion of a catalyst particle in an enclosure where the particle temperature is allowed to change both in time and space. They considered the following three cases:

a. $\rho_r = 1.1$, $c_r = 1$, $k_r = 15$;
b. $\rho_r = 1.6$, $c_r = 1$, $k_r = 5$;
c. $\rho_r = 100$, $c_r = 1$, $k_r = 15$.

It is expected that our DNS method should be able to produce a good approximation for cases (a) and (c) where $k_r = 15$. This motivates us to revisit these cases. In addition, we add another case (we name it as case d) where $\rho_r = 1$ and $c_r = 1$ to demonstrate that the particle motion and temperature updating scheme works for particles of the same density and the same specific heat as fluid. However, the reference velocity of $\sqrt{(\pi a(\rho_r - 1)g)}$ used for cases (a), (b) and (c) will yield zero for case (d), thus we choose a reference velocity of $\sqrt{(\pi a g)}$ instead. The enclosure is $8a$ wide and $16a$ high (a is the radius of particle). Initially the particle is located at the center of the enclosure; fluid and particle have zero velocity and temperature. The particle has a constant heat source of $q_p = 1$ over its entire region. We set $Re = 40$, $Pr = 0.7$ and $Gr = 1000$. A grid of 240×480 is used for all these cases.

We plotted the evolution of the particle vertical velocity in Fig. 8. It is seen that the particle in case (a) and (b) falls downward at the beginning due to the gravity force; however, as the particle temperature rises because of the release of heat source within it, so does the temperature-gradient induced buoyancy force; at certain point this buoyancy force is able to overcome the gravity force, and the particle begins to have upward acceleration at $t > 3$. At $t > 7$ for case (a) or $t > 8$ for case (b), the particle starts to reverse the direction and moves upward until it experiences the resistance caused by the presence of the upper wall. The particle in case (c) has a much higher particle–fluid density and the gravity force dominated the particle's downward motion. On the other hand, the particle in case (d) has the same density as fluid and its upward motion is controlled by the temperature-gradient induced buoyancy force. The results for cases (a), (b) and (c) taken form Yu et al. [24] are also plotted in Fig. 8 for comparisons.

It must be pointed out that the DNS method developed in this paper assumes the particle has a higher particle–fluid thermal conductivity and the particle temperature is assumed to be uniform. From Fig. 8, it shows a very good agreement for cases (a) and (c) where $k_r = 15$ between the results from our DNS method and those obtained by fictitious domain method. Even for case (b) where a moderate k_r is used ($k_r = 5$), we found that our DNS result approximates the one obtained by Yu et al. [24] reasonably well. In addition, the success of achieving stable results for case (c) and (d) signifies that the updating scheme described by Eqs. (43), (44) and (46) can be used for either very high density particle or particle of the same density and the same specific heat as fluid. Fig. 9 shows the temperature contour and velocity vector map at three different times for a catalyst particle moving in an enclosure.

We also conduct the grid size test and time step test using case (a) of $\rho_r = 1.1$; results for the vertical velocity vs. time are plotted in

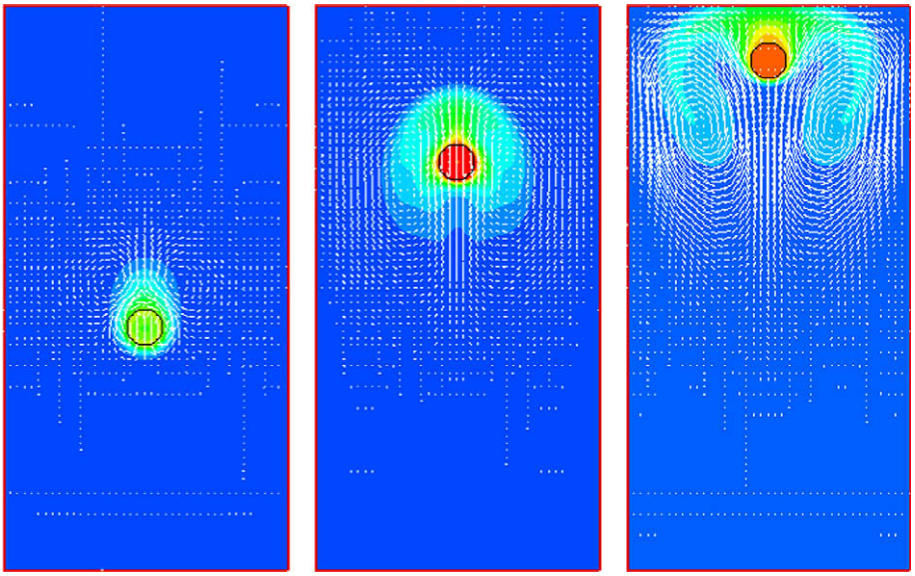

Fig. 9. Temperature contours and velocity vector for a catalyst particle of density 1.6 in an enclosure at $t = 5$, 15 and 20, respectively.

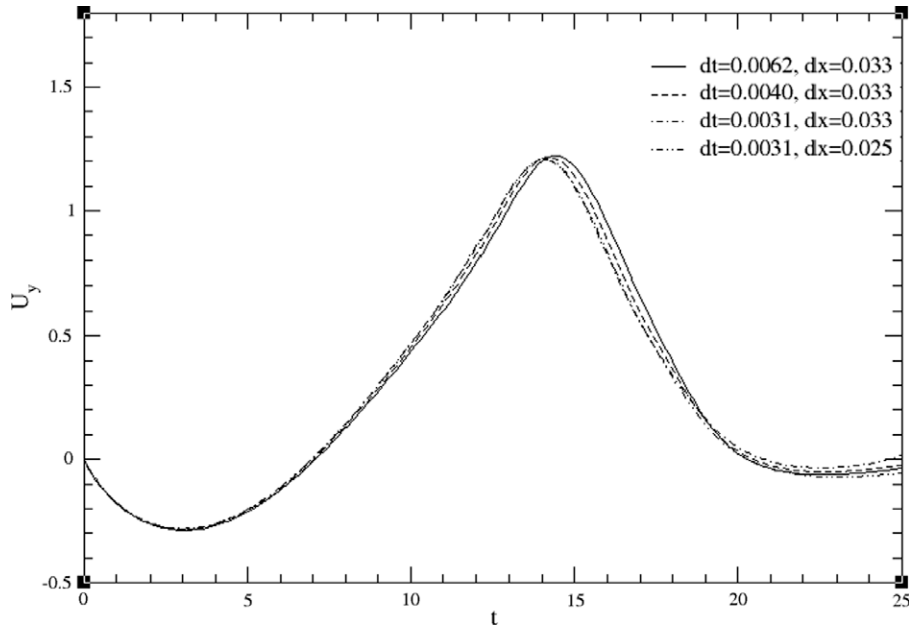

Fig. 10. Evolution of vertical velocity of a catalyst particle with $\rho_r = 1.1$ using different grid size and time step.

Fig. 10. It shows that the simulation results become insensitive as the time step $\delta t \leqslant 0.004$. It also indicates that the difference in results using $\delta x = 0.033$ (a grid of 240×480) and $\delta x = 0.025$ (a grid of 320×640) is insignificant when the same time step $\delta t = 0.0031$ is used.

4.4. Sedimentation of 56 interacting circular particles in an enclosure

We also considered the sedimentation of 56 heated circular particles in an enclosure. The particles are initially hot and, because of the energy exchange with the fluid, their temperature changes according to Eq. (46). The physical parameters are as follows: domain size $\Omega = [0,3] \times [0,6]$; particle radius $r = 0.08$; fluid viscosity $\nu_f = 0.002$; particle–fluid density and heat capacity ratios $\rho_r = 1.5$ and $c_r = 5.0$; $Pr = 1$; and $Gr = 200$. A 256×512 grid is used, which implies $r \approx 6.8$. Thus, the particle diameter is outlined by 13.6 grid points. The time step is $\delta t = 0.0005$.

Fig. 11 shows the temperature contours at three different stages of the settling process. Initially, the top region is heated up by the presence of the hot particles. As the heavier particles settle and move downwards, they cause a strong fluid motion, which may be observed in the velocity vector map of Fig. 12. This results in significant mixing in the fluid and higher uniformity of the temperature. As the particles settle, they also cool, with the cooling effect being more pronounced with the leading particles. It is apparent in the calculations and Fig. 11 that the particles following in the thermal wakes of others have higher temperature than those of the leading particles.

Fig. 12a depicts the velocity vectors of the entire flow domain at $t = 9$. In order to elucidate the effectiveness of our forcing scheme, we have chosen a small part of this figure with three particles and magnified the velocity field, as shown in Fig. 12b. It is observed that the rigid body motion of the particle interior region is enforced very well using the numerical technique presented in this study. The temperature contours, which are depicted in Fig. 12c show clearly the strong temperature gradients in the vicinity of the particles.

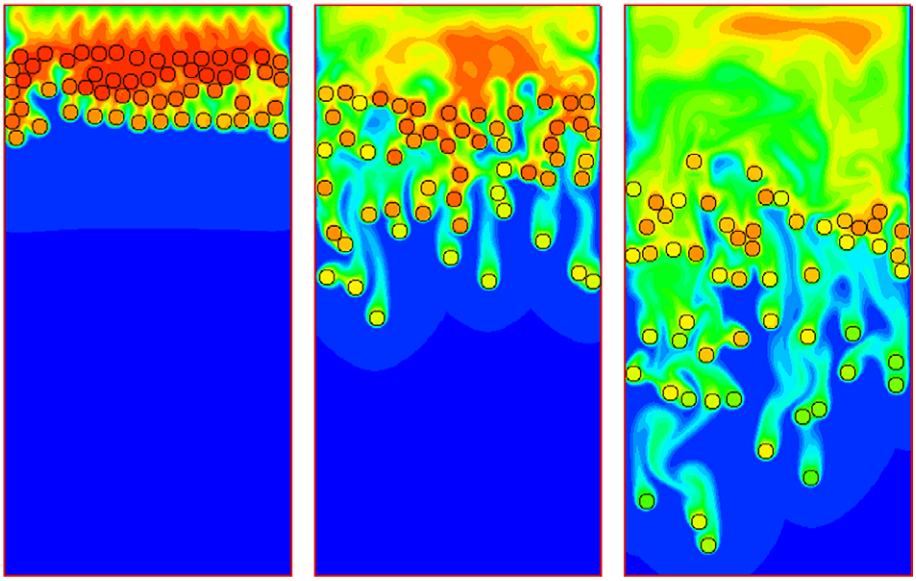

Fig. 11. Temperature contours of 56 circular particles at $t = 3$, 6 and 9.

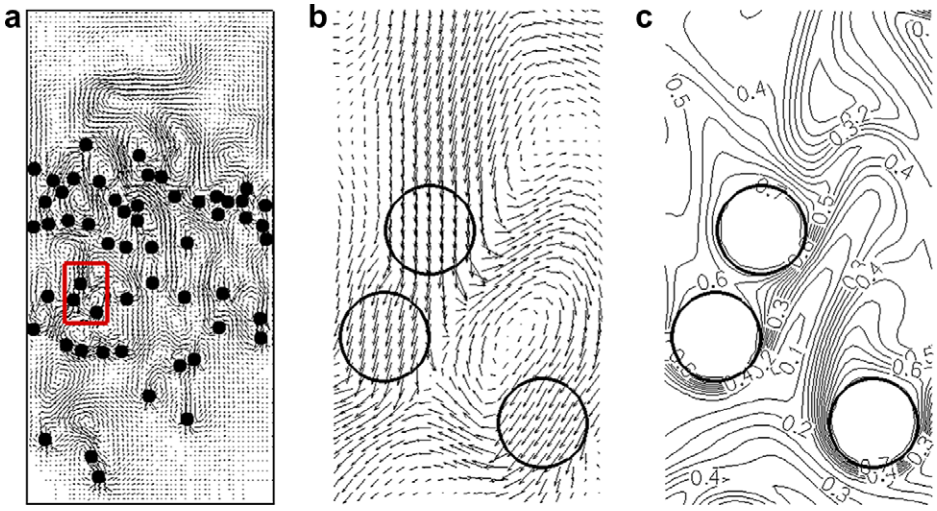

Fig. 12. Detailed velocity and temperature fields around particles at $t = 9$. The small box in part (a) indicates the area of the details in parts (b) and (c). (a) Velocity vectors; (b) detailed velocity field; (c) temperature contours.

5. Conclusions

A numerical method that is based on DNS and solves both the momentum and heat interactions in particulate flows has been developed. In order to enforce the rigid body motion of solid particles, a force density function has been placed in the entire region where the particles are present. This force density field has been calculated using a direct-forcing scheme. We used an analogous method and an energy density function to enforce the uniform temperature conditions in the regions that are occupied by particles of high thermal conductivity. The velocity and temperature of the particles are obtained by solving the set of ordinary differential equations, which represent the Lagrangian equation of motion and the energy equation of the particles. The numerical method developed was validated by comparisons with cases pertaining to the motion of particles with constant temperature as well as to the motion of particles with time-dependent temperature. Results obtained with a group of 56 particles that cool while settling, show how the local temperature field and buoyancy force affects the sedimentation process and the transfer of energy; they also show that this novel DNS method may be accurately used with particulate systems of a large number of particles, where the particulate motion and heat transfer occur simultaneously.

References

[1] J. Feng, H.H. Hu, D.D. Joseph, Direct simulation of initial value problems for the motion of solid bodies in a Newtonian fluid. Part 1. Sedimentation, J. Fluid Mech. 261 (1994) 95–134.
[2] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (1977) 220–252.
[3] A.L. Fogelson, C.S. Peskin, A fast numerical method for solving the three-dimensional Stokes equation in the presence of suspended particles, J. Comput. Phys. 79 (1988) 50–69.
[4] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip boundary with an external force field, J. Comput. Phys. 105 (1993) 354–366.
[5] K. Hofler, S. Schwarzer, Navier–Stokes simulation with constraint forces: finite-difference method for particle-laden flows and complex geometries, Phys. Rev. E 61 (2000) 7146–7160.
[6] Z.-G. Feng, E.E. Michaelides, The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems, J. Comput. 194 (2004) 602–628.
[7] M. Uhlmann, First experiments with the simulation of particulate flows, Technical Report No. 1020, CIEMAT, Madrid, Spain, ISSN 1135-9420, 2003.
[8] Z.-G. Feng, E.E. Michaelides, Proteus: a direct forcing method in the simulations of particulate flow, J. Comput. Phys. 202 (2005) 20–51.
[9] Z.-G. Feng, E.E. Michaelides, Robust treatment of no-slip boundary condition and velocity updating for the lattice-Boltzmann simulation of particulate flows, Comput. Fluids, in press. Also available online May 15:doi:10.1016/j.compfluid.2008.04.013.
[10] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.
[11] J. Mohd-Yusof, Combined immersed boundaries/B-splines methods for simulations of flows in complex geometries, Annual Research Briefs, Center for Turbulence Research, Stanford University, 1997.
[12] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulation, J. Comput. Phys. 161 (2000) 35–60.
[13] R. Glowinski, Finite element method for incompressible viscous flow, in: P.G. Ciarlet, J.L. Lions (Eds.), Handbook of Numerical Analysis, vol. IX, North-Holland, Amsterdam, 2008.
[14] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, J. Periaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2001) 363–426.
[15] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiphase Flow 25 (1999) 755–794.
[16] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, T.-W. Pan, A new formulation of the distributed Lagrange multiplier/_fictitious domain method for particulate flows, Int. J. Multiphase Flow 26 (2000) 1509–1524.
[17] W. Kalthoff, S. Schwarzer, H.J. Hermann, Algorithm for the simulation of particle suspensions with inertia effects, Phys. Rev. E 56 (1996) 2234–2242.
[18] Z. Zhang, A. Prosperetti, A method for particle simulation, J. Appl. Mech. 70 (2003) 64–74.
[19] Z. Zhang, A. Prosperetti, A second-order method for three-dimensional particle simulation, J. Comput. Phys. 210 (2005) 292–324.
[20] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part I. Theoretical foundation, J. Fluid Mech. 271 (1994) 285–310.
[21] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part II: numerical results, J. Fluid Mech. 271 (1994) 311–339.
[22] H. Gan, J.Z. Chang, H.H. Howard, Direct numerical simulation of the sedimentation of solid particles with thermal convection, J. Fluid Mech. 481 (2003) 385–411.
[23] H. Gan, J.Z. Chang, H.H. Howard, Simulation of the sedimentation of melting solid particles, Int. J. Multiphase Flow 29 (2003) 751–769.
[24] Z. Yu, X. Shao, A. Wachs, A fictitious domain method for particulate flows with heat transfer, J. Comput. Phys. 217 (2006) 424–452.
[25] J. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of heat transfer in complex geometries, Korean Soc. Mech. Eng. Int. J. 18 (2004) 1026–1035.
[26] J.R. Pacheco, A. Pacheco-Vega, T. Rodic, R.E. Peck, Numerical simulations of heat transfer and fluid problems using an immersed-boundary finite-volume method on non-staggered grids, Numer. Heat Transfer B 48 (2005) 1–24.
[27] Z.-G. Feng, E.E. Michaelides, Inclusion of heat transfer computations for particle laden flows, Phys. Fluids 20 (2008) 675–684.
[28] I. Demirdzic, Z. Lilek, M. Peric, Fluid flow and heat transfer test problems for non-orthogonal grids: Bench-Mark solutions, Int. J. Numer. Methods Fluids 15 (1992) 329–354.
[29] Uhlmann, New results on the simulation of particulate flows, Technical Report No. 1038, CIEMAT, Madrid, Spain, 2003. ISSN 1135-9420.