

CFD modeling of bubbling fluidized beds using OpenFOAM®: Model validation and comparison of TVD differencing schemes



Yefei Liu, Olaf Hinrichsen*

Catalysis Research Center and Chemistry Department, Technische Universität München, Garching b. München D-85748, Germany

ARTICLE INFO

Article history:

Received 11 January 2014

Received in revised form 23 May 2014

Accepted 7 July 2014

Available online 15 July 2014

Keywords:

Fluidized bed

CFD

Two-fluid model

TVD scheme

OpenFOAM

ABSTRACT

The two-fluid model with kinetic theory of granular flow is implemented into the open source CFD package OpenFOAM®. The effect of total variation diminishing (TVD) convection schemes is investigated by simulating two bubbling fluidized beds. Five TVD schemes are employed to discretize the convection terms of phase velocity and solid volume fraction. Simulated results of the two test cases give reasonable agreement with the experimental data in the literature. For the discretization of the phase velocity convection terms, the five schemes give quite similar time-averaged radial profiles of particle axial velocity. The predicted bubbles in the bed with a central jet are not influenced by the different schemes. For the discretization of the solid volume fraction convection terms, the limitedLinear01, Sweby01 and vanLeer01 schemes give the converged and reasonable solutions, whereas the SuperBee01 and MUSCL01 schemes diverge the solutions. When using the faceLimited gradient scheme the convection scheme becomes more diffusive.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

The Eulerian–Eulerian two-fluid model is widely employed for computational fluid dynamics (CFD) simulations of the gas–solid flows, since it requires economical computational resources and enables large-scale reactor modeling. The two-fluid model treats gas and solid phases as fully interpenetrating continua, which typically involves two sets of averaged Navier–Stokes equations (Ding & Gidaspow, 1990; Enwald, Peirano, & Almstedt, 1996; Ishii, 1975). These governing equations are closed with the interfacial transfer models and the kinetic–frictional models for the description of granular phase (Jenkins & Savage, 1983; Johnson & Jackson, 1987; Lun, Savage, Jeffrey, & Chepur, 1984; Schaeffer, 1987; Srivastava & Sundaresan, 2003).

Numerical solution of two-fluid model equations with kinetic theory of granular flow (KTGF) has been achieved with some commercial software, in-house programs and open source codes, e.g., FLUENT®, CFX®, MFX and K-FIX. The commercial CFD packages FLUENT and CFX gain many practices in Eulerian-based modeling of gas–solid flows (Asegehegn, Schreiber, & Krauz, 2011; Chalermisinsuwan, Kuchonthara, & Piumsomboon, 2010; Hulme, Clavelle, van der Lee, & Kantzas, 2005; Lettieri, Cammarata, Micale, & Yates, 2003; Yang, Wang, Ge, & Li, 2003; Zhang, Pei, Brandon,

Chen, & Yang, 2012). FLUENT uses a phase-coupled SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) algorithm, while the inter-phase slip algorithm is adopted in CFX. The open source code MFX (Multiphase Flow with Interphase eXchange) developed at National Energy Technology Laboratory is also widely applied for fluidized bed simulations (Li, Grace, & Bi, 2010; Li, Dietiker, Zhang, & Shahnam, 2011; McKeen & Pugsley, 2003; Syamlal, Rogers, & O'Brien, 1993). MFX is developed with staggered grid arrangement and Fortran coding. The MFX manuals are well documented and freely available to researchers. The in-house program K-FIX and its alternate versions are employed by few researchers (Lu & Gidaspow, 2003; Lu et al., 2004). The K-FIX source codes are not available to the public. Recently, the open source CFD code OpenFOAM® (Open source Field Operation And Manipulation) (OpenCFD Ltd., 2012) receives much more attention from academia and industry. In OpenFOAM the original gas–solid flow solvers with two-fluid model are not yet fully developed and validated (Herzog, Schreiber, Egbers, & Krautz, 2012). To improve the solver accuracy and robustness, Passalacqua and Fox (2011) proposed an iterative solution procedure to solve the two-fluid model equations.

OpenFOAM is developed based on the fundamental ideas of object orientation, layered software design and equation mimicking. Many CFD solvers for specialized applications have been implemented in OpenFOAM. Compared with the business software products, OpenFOAM offers the possibility to have insight into the source codes and hence it is of great convenience for code customization. Instead of the staggered grids, OpenFOAM adopts the

* Corresponding author. Tel.: +49 89 289 13232; fax: +49 89 289 13513.
E-mail address: olaf.hinrichsen@ch.tum.de (O. Hinrichsen).

co-located grids. The unstructured polyhedral cells can be deployed to handle complex geometries. The use of the object-oriented C++ language further makes users implement new models readily. OpenFOAM is becoming a flexible alternative tool for CFD simulation. However, the gas–solid flow simulations with OpenFOAM are still quite few.

Gas–solid flow simulation in a fluidized bed depends not only on the fundamental two-fluid model theory but also on the numerical techniques, i.e., time step, solution algorithm and discretization scheme. First, it is important to achieve the complete implementation of the two-fluid model with KTGF. While a robust gas–solid solver was developed and tested by Passalacqua and Fox (2011), further experimental validations are still very necessary to confirm the applicability of OpenFOAM in simulating fluidized beds. Second, for the fluidized beds with convection-dominated flows, great care should be taken to the discretization schemes of the convection terms. Guenther and Syamlal (2001) compared the first-order upwind scheme with three second-order schemes (SMART, SuperBee and MINMOD) in the MFIX code. And the effect of convection scheme on the isolated bubble shape was investigated. Braun, Mineto, Navarro, Cabezas-Gómez, and Silva (2010) also simulated the bubble behaviors in a fluidized bed by using the MFIX code. The first-order upwind and SuperBee schemes were compared. Although various convection schemes are available in OpenFOAM, little attention is paid to the effect of convection scheme on the solution results of gas–solid flows in fluidized beds. It is of major importance to perform an in-depth comparison of convection schemes within the OpenFOAM framework. Such a study provides the guideline on the selection of convection schemes in the gas–solid flow modeling.

In this work a two-fluid solver is implemented into OpenFOAM. Experimental validations are performed by simulating two bubbling fluidized bed cases, i.e., one with uniform gas feed and the other with a central gas jet. The simulated results are compared against the experimental data from the literature. The effect of different convection schemes is investigated for the convection terms of phase velocities and solid volume fraction.

2. Mathematical models

The continuity equations of the gas and solid phases are written as

$$\frac{\partial(\alpha_g \rho_g)}{\partial t} + \nabla \cdot (\alpha_g \rho_g \mathbf{U}_g) = 0 \quad (1)$$

$$\frac{\partial(\alpha_s \rho_s)}{\partial t} + \nabla \cdot (\alpha_s \rho_s \mathbf{U}_s) = 0 \quad (2)$$

where ρ_g is the gas phase density, ρ_s is the solid phase density, \mathbf{U}_g is the gas phase velocity, \mathbf{U}_s is the solid phase velocity, α_g and α_s are the volumetric fractions of gas and solid phases, $\alpha_g = 1 - \alpha_s$.

The momentum equations of the gas and solid phases are given as

$$\begin{aligned} \frac{\partial(\alpha_g \rho_g \mathbf{U}_g)}{\partial t} + \nabla \cdot (\alpha_g \rho_g \mathbf{U}_g \mathbf{U}_g) = & -\alpha_g \nabla p + \nabla \cdot (\alpha_g \tau_g) \\ & + \alpha_g \rho_g \mathbf{g} + \beta(\mathbf{U}_s - \mathbf{U}_g) \end{aligned} \quad (3)$$

$$\begin{aligned} \frac{\partial(\alpha_s \rho_s \mathbf{U}_s)}{\partial t} + \nabla \cdot (\alpha_s \rho_s \mathbf{U}_s \mathbf{U}_s) = & -\alpha_s \nabla p - \nabla p_s + \nabla \cdot (\alpha_s \tau_s) \\ & + \alpha_s \rho_s \mathbf{g} + \beta(\mathbf{U}_g - \mathbf{U}_s) \end{aligned} \quad (4)$$

where p is the bed pressure, p_s is the solid phase pressure, τ_g and τ_s are the stress tensors of gas and solid phases, β is the interphase momentum transfer coefficient, and \mathbf{g} is the gravitational acceleration.

Table 1

The interphase momentum transfer coefficients.

(a) Syamlal model (Syamlal et al., 1993)	
$\beta = \frac{3}{4} C_D \frac{\alpha_g \alpha_s \rho_g}{V_r^2 d_p} \mathbf{U}_g - \mathbf{U}_s $	
$C_D = \left(0.63 + 4.8 \sqrt{\frac{V_r}{Re}} \right)^2$	
$V_r = 0.5[a - 0.06Re + \sqrt{(0.06Re)^2 + 0.12Re(2b - a) + a^2}]$	
$a = \alpha_g^{4.14}, \quad b = \begin{cases} 0.8\alpha_g^{1.28} & \alpha_g \leq 0.85 \\ \alpha_g^{2.65} & \alpha_g > 0.85 \end{cases}$	
$Re = \frac{\rho_g d_p \mathbf{U}_g - \mathbf{U}_s }{\mu_g}$	
(b) Gidaspow model (Gidaspow, 1994)	
$\beta = \begin{cases} \frac{3}{4} C_D \alpha_g \alpha_s \rho_g \frac{ \mathbf{U}_g - \mathbf{U}_s }{d_p} \alpha_g^{-2.65} & \alpha_s < 0.2 \\ 150 \frac{\mu_g \alpha_s^2}{\alpha_g d_p^2} + 1.75 \frac{\rho_g \alpha_s}{d_p} \mathbf{U}_g - \mathbf{U}_s & \alpha_s \geq 0.2 \end{cases}$	
$C_D = \begin{cases} \frac{24}{\alpha_g Re} [1 + 0.15(\alpha_g Re)^{0.687}] & \alpha_g Re < 1000 \\ 0.44 & \alpha_g Re \geq 1000 \end{cases}$	

The gas phase is assumed as a Newtonian fluid, and its stress tensor is defined using the Newtonian stress–strain relation as

$$\tau_g = \mu_g [\nabla \mathbf{U}_g + (\nabla \mathbf{U}_g)^T] - \frac{2}{3} \mu_g (\nabla \cdot \mathbf{U}_g) \mathbf{I} \quad (5)$$

where μ_g is the shear viscosity of gas phase and \mathbf{I} is the unit tensor. Similarly, the shear stress tensor of solid phase is expressed as

$$\tau_s = \mu_s [\nabla \mathbf{U}_s + (\nabla \mathbf{U}_s)^T] + \left(\lambda_s - \frac{2}{3} \mu_s \right) (\nabla \cdot \mathbf{U}_s) \mathbf{I} \quad (6)$$

where μ_s is the solid shear viscosity and λ_s is the solid bulk viscosity.

2.1. Interphase momentum transfer coefficient

In order to solve the phase momentum equations, the interphase momentum transfer coefficient β should be calculated with drag force functions. The majority of the reported drag models were empirically developed. No drag model is available for predicting a wide range of gas–solid flow types. In this work, the drag force function of Syamlal et al. (1993) is applied for the fluidized bed with uniform gas feed. This function was derived by converting the terminal velocity correlations in fluidized or settling beds. The fluidized bed with a central jet is simulated with the drag model proposed by Gidaspow (1994). The Gidaspow model combines the Ergun equations (Ergun, 1952) with the Wen and Yu model (Wen & Yu, 1966). The Ergun equations were derived from the packed-bed pressure drop data, whereas the Wen and Yu model was formulated based on the homogeneous expansion of fluidized beds.

The drag correlations of the Syamlal and Gidaspow models are given in Table 1. Fig. 1 shows the comparison between these models. At the high solid volume fractions, the values of β calculated with the Gidaspow model are larger than those calculated with the Syamlal model. Therefore, the Gidaspow model predicts stronger bed expansion and the predicted solid concentrations are smaller. Our preliminary study revealed that the fluidized bed with uniform gas feed is better simulated using the Syamlal model. However, the Gidaspow model is needed to obtain the satisfactory predictions on the fluidized bed with a central jet.

2.2. Kinetic theory of granular flow

The solid shear stress τ_s and solid pressure p_s in Eq. (4) are modeled by the kinetic theory of granular flow (Gidaspow, 1994). The fluctuation energy of solid phase, also known as granular

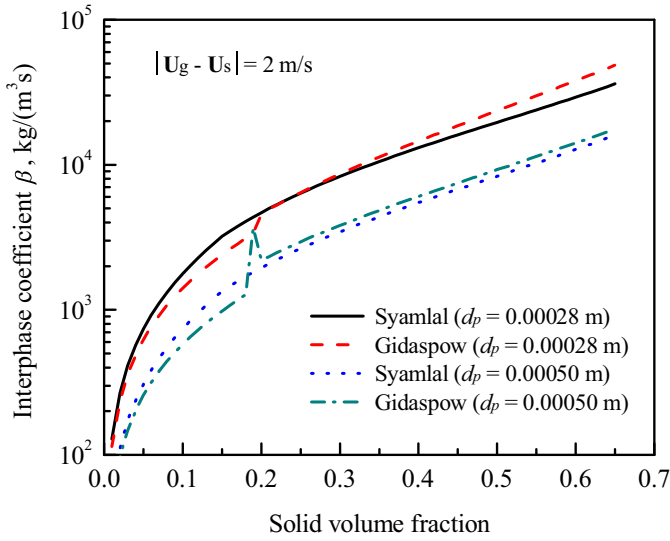


Fig. 1. Comparison of different interphase momentum transfer coefficients.

temperature, is obtained by solving its transport equation:

$$\frac{3}{2} \left[\frac{\partial}{\partial t} (\alpha_s \rho_s \Theta) + \nabla \cdot (\alpha_s \rho_s \mathbf{U}_s \Theta) \right] = (-p_s \mathbf{I} + \tau_s) : \nabla \mathbf{U}_s + \nabla \cdot (\kappa_s \nabla \Theta) - \gamma_s + J_{vis} + J_{slip} \quad (7)$$

where Θ is the granular temperature, κ_s is the conductivity of granular temperature, γ_s is the dissipation rate due to particle collisions, J_{vis} is the dissipation rate resulting from viscous damping, and J_{slip} is the production rate due to the slip between gas and particle. These terms are modeled following the work of Gidaspow (1994).

The solid bulk viscosity λ_s and the solid shear viscosity μ_s are calculated according to Gidaspow (1994):

$$\lambda_s = \frac{4}{3} \alpha_s \rho_s d_p g_0 (1 + e) \left(\frac{\Theta}{\pi} \right)^{1/2} \quad (8)$$

$$\mu_{s,KTGF} = \frac{4}{5} \alpha_s^2 \rho_s d_p g_0 (1 + e) \left(\frac{\Theta}{\pi} \right)^{1/2} + \frac{10 \rho_s d_p \sqrt{\Theta \pi}}{96 g_0 (1 + e)} \left[1 + \frac{4}{5} \alpha_s g_0 (1 + e) \right]^2 \quad (9)$$

where d_p is the particle diameter, e is the particle–particle restitution coefficient and g_0 is the radial distribution function. The solid phase pressure p_s is calculated following the work of Lun et al. (1984):

$$p_{s,KTGF} = \alpha_s \rho_s \Theta [1 + 2(1 + e) g_0 \alpha_s] \quad (10)$$

The expression of g_0 proposed by Sinclair and Jackson (1989) is used:

$$g_0 = \left[1 - \left(\frac{\alpha_s}{\alpha_{s,max}} \right)^{1/3} \right]^{-1} \quad (11)$$

where $\alpha_{s,max}$ is the particle packing limit.

2.3. Frictional stress models

In the regions where the particles are closely packed, the behavior of the granular flow is dominated by the frictional stresses and it is not adequately described only by KTGF. Similar to the solid kinetic stresses, the solid frictional stresses consist of the frictional shear stress and the frictional normal stress (i.e., solid frictional

pressure). When the solid volume fraction exceeds a critical value $\alpha_{s,min}$, the solid frictional stresses are usually added to the solid kinetic stresses calculated from KTGF (Johnson, Nott, & Jackson, 1990):

$$p_s = p_{s,KTGF} + p_{s,f} \quad (12)$$

$$\mu_s = \mu_{s,KTGF} + \mu_{s,f} \quad (13)$$

The solid frictional pressure $p_{s,f}$ proposed by Schaeffer (1987) and the frictional shear viscosity $\mu_{s,f}$ by Johnson et al. (1990) are employed as

$$p_{s,f} = Fr \frac{(\alpha_s - \alpha_{s,min})^n}{(\alpha_{s,max} - \alpha_s)^p} \quad (14)$$

$$\mu_{s,f} = \frac{p_{s,f} \sin \phi_{fr}}{2 \sqrt{I_{2D}}} \quad (15)$$

where $Fr = 0.05 \text{ N/m}^2$, $n = 2$, $p = 5$, ϕ_{fr} is the internal frictional angle, $\phi_{fr} = 28.5^\circ$, and I_{2D} represents the second invariant of the deviator of the strain rate tensor D_s :

$$I_{2D} = \frac{1}{6} [(D_{s11} - D_{s22})^2 + (D_{s22} - D_{s33})^2 + (D_{s33} - D_{s11})^2] + D_{s12}^2 + D_{s23}^2 + D_{s31}^2 \quad (16)$$

$$D_s = \frac{1}{2} [\nabla \mathbf{U}_s + (\nabla \mathbf{U}_s)^T] \quad (17)$$

3. Numerical implementation

3.1. Discretization of the phase momentum equations

The gas and solid phases are assumed to be incompressible. Eqs. (3) and (4) are re-written by dividing the density of each phase:

$$\begin{aligned} \frac{\partial(\alpha_g \mathbf{U}_g)}{\partial t} + \nabla \cdot (\alpha_g \mathbf{U}_g \mathbf{U}_g) - \nabla \cdot (\alpha_g \nu_g \nabla \mathbf{U}_g) + \nabla \cdot (\alpha_g R_g^C) \\ = -\frac{\alpha_g}{\rho_g} \nabla p + \alpha_g \mathbf{g} + \frac{\beta}{\rho_g} (\mathbf{U}_s - \mathbf{U}_g) \end{aligned} \quad (18)$$

$$\begin{aligned} \frac{\partial(\alpha_s \mathbf{U}_s)}{\partial t} + \nabla \cdot (\alpha_s \mathbf{U}_s \mathbf{U}_s) - \nabla \cdot (\alpha_s \nu_s \nabla \mathbf{U}_s) + \nabla \cdot (\alpha_s R_s^C) \\ = -\frac{\alpha_s}{\rho_s} \nabla p - \frac{1}{\rho_s} \nabla p_s + \alpha_s \mathbf{g} + \frac{\beta}{\rho_s} (\mathbf{U}_g - \mathbf{U}_s) \end{aligned} \quad (19)$$

where ν is the kinematic viscosity, $\nu = \mu/\rho$. R_g^C and R_s^C are expressed as follows:

$$R_g^C = -\nu_g (\nabla \mathbf{U}_g)^T + \frac{2}{3} \nu_g (\nabla \cdot \mathbf{U}_g) \mathbf{I}$$

$$R_s^C = -\nu_s (\nabla \mathbf{U}_s)^T - \left(\frac{\lambda_s}{\rho_s} - \frac{2}{3} \nu_s \right) (\nabla \cdot \mathbf{U}_s) \mathbf{I}$$

The difficulty is found in the solution of the phase momentum equations, since the solid phase is not necessarily present in the whole computational domain of a bubbling fluidized bed. As a result, the discretized momentum equations result in the singular system of linear algebraic equations. Some workers suggested re-writing the phase momentum equations in the non-conservative or semi-conservative forms to avoid the singularity (Oliveira & Issa, 2003; Park, Cho, Yoon, & Jeong, 2009). Passalacqua and Fox (2011) adopted the conservative forms of the phase momentum equations. In those computational cells with solid volume fraction lower than a specified minimum value, the discretized momentum equations were not solved.

The momentum conservative form is also adopted in our solver. As an example of the equation implementation in OpenFOAM, the transient term, convection term, diffusion term and divergence term in the l.h.s of Eq. (19) are discretized with the following codes:

```
fvVectorMatrix UaEqn
(
    fvm::ddt(alpha,Ua)
  +fvm::div(alpha*f*phia,Ua)
  -fvm::laplacian(alpha*nua,Ua)
  +fvc::div(alpha*Rca)
);
```

where α represents the solid volume fraction, α_{f} is the interpolated face value of solid volume fraction, and ϕ_{ia} is the particle velocity flux. The fvm operator returns the implicit discretization of the transient, convection and diffusion terms, while the divergence of $\alpha \mathbf{R}_{\text{ca}}$ is explicitly discretized with the fvc operator. It is seen that the different terms in the C++ implementation mimics their mathematical counterparts.

By discretizing the transient, convection, diffusion and divergence terms in Eqs. (18) and (19), the semi-discretized forms of the momentum equations are written as

$$A_g \mathbf{U}_g = \mathbf{H}_g - \frac{\alpha_g}{\rho_g} \nabla p + \alpha_g \mathbf{g} + \frac{\beta}{\rho_g} (\mathbf{U}_s - \mathbf{U}_g) \quad (20)$$

$$A_s \mathbf{U}_s = \mathbf{H}_s - \frac{\alpha_s}{\rho_s} \nabla p - \frac{1}{\rho_s} \nabla p_s + \alpha_s \mathbf{g} + \frac{\beta}{\rho_s} (\mathbf{U}_g - \mathbf{U}_s) \quad (21)$$

where A and \mathbf{H} are accessed by using the member function $\mathbf{A}()$ and $\mathbf{H}()$ of the class fvVectorMatrix , respectively. The parameter A gives the diagonal coefficients of the velocity matrix. \mathbf{H} represents the off-diagonal part of the velocity matrix and the source part.

The partial elimination algorithm proposed by Spalding (1980) is adopted to decouple the phase momentum equations. The different phase velocities in Eqs. (20) and (21) are collected:

$$\mathbf{U}_g = \xi_g \left(\mathbf{H}_g - \frac{\alpha_g}{\rho_g} \nabla p + \alpha_g \mathbf{g} \right) + \xi_g \frac{\beta}{\rho_g} \mathbf{U}_s \quad (22)$$

$$\mathbf{U}_s = \xi_s \left(\mathbf{H}_s - \frac{\alpha_s}{\rho_s} \nabla p - \frac{1}{\rho_s} \nabla p_s + \alpha_s \mathbf{g} \right) + \xi_s \frac{\beta}{\rho_s} \mathbf{U}_g \quad (23)$$

with

$$\xi_g = \frac{1}{A_g + \max(\delta, \alpha_s \alpha_g)(\beta^*/\rho_g)}, \quad \xi_s = \frac{1}{A_s + \max(\delta, \alpha_s \alpha_g)(\beta^*/\rho_s)}$$

where β^* is obtained by extracting $\alpha_s \alpha_g$ from β . In this study a small value ($\delta = 1 \times 10^{-4}$) is introduced into β to avoid the infinite values of ξ_g and ξ_s when $\alpha_s = 0$.

Substituting \mathbf{U}_s in Eq. (20) using Eq. (23) and \mathbf{U}_g in Eq. (21) using Eq. (22), we obtain the decoupled momentum equations:

$$\begin{aligned} \mathbf{U}_g = & \frac{\mathbf{H}_g}{A_g^*} + \frac{\xi_s \beta}{\rho_g A_g^*} \mathbf{H}_s - \left(\frac{\alpha_g}{\rho_g A_g^*} + \frac{\alpha_s \xi_s \beta}{\rho_g \rho_s A_g^*} \right) \nabla p \\ & - \frac{\xi_s \beta}{\rho_g \rho_s A_g^*} \nabla p_s + \left(\frac{\alpha_g}{A_g^*} + \frac{\alpha_s \xi_s \beta}{\rho_g A_g^*} \right) \mathbf{g} \end{aligned} \quad (24)$$

$$\begin{aligned} \mathbf{U}_s = & \frac{\mathbf{H}_s}{A_s^*} + \frac{\xi_g \beta}{\rho_s A_s^*} \mathbf{H}_g - \left(\frac{\alpha_s}{\rho_s A_s^*} + \frac{\alpha_g \xi_g \beta}{\rho_g \rho_s A_s^*} \right) \nabla p \\ & - \frac{1}{\rho_s A_s^*} \nabla p_s + \left(\frac{\alpha_s}{A_s^*} + \frac{\alpha_g \xi_g \beta}{\rho_s A_s^*} \right) \mathbf{g} \end{aligned} \quad (25)$$

with

$$A_g^* = A_g - \frac{\beta}{\rho_g} \left(\xi_s \frac{\beta}{\rho_s} - 1 \right), \quad A_s^* = A_s - \frac{\beta}{\rho_s} \left(\xi_g \frac{\beta}{\rho_g} - 1 \right)$$

3.2. Pressure equation and PISO algorithm

The PISO (Pressure Implicit with Split Operator) solution algorithm is adopted to handle the pressure–velocity coupling (Issa, 1985), where a pressure equation based on the volumetric continuity equation is solved to correct the predicted velocities. The velocity fluxes are obtained by calculating the dot-product of velocities in Eqs. (24) and (25) with the surface area vector \mathbf{S}_f :

$$\begin{aligned} \varphi_g = & \left(\frac{\mathbf{H}_g}{A_g^*} \right)_f \cdot \mathbf{S}_f + \left(\frac{\xi_s \beta}{A_g^* \rho_g} \mathbf{H}_s \right)_f \cdot \mathbf{S}_f - \left(\frac{\alpha_g}{A_g^* \rho_g} + \frac{\alpha_s \xi_s \beta}{A_g^* \rho_g \rho_s} \right) |\mathbf{S}_f| \nabla^\perp p \\ & - \left(\frac{\xi_s \beta}{A_g^* \rho_g \rho_s} \right) (\nabla p_s)_f \cdot \mathbf{S}_f + \left(\frac{\alpha_g}{A_g^*} + \frac{\alpha_s \xi_s \beta}{A_g^* \rho_g} \right) \mathbf{g} \cdot \mathbf{S}_f \end{aligned} \quad (26)$$

$$\begin{aligned} \varphi_s = & \left(\frac{\mathbf{H}_s}{A_s^*} \right)_f \cdot \mathbf{S}_f + \left(\frac{\xi_g \beta}{A_s^* \rho_s} \mathbf{H}_g \right)_f \cdot \mathbf{S}_f - \left(\frac{\alpha_s}{A_s^* \rho_s} + \frac{\alpha_g \xi_g \beta}{A_s^* \rho_g \rho_s} \right) |\mathbf{S}_f| \nabla^\perp p \\ & - \left(\frac{1}{A_s^* \rho_s} \right) (\nabla p_s)_f \cdot \mathbf{S}_f + \left(\frac{\alpha_s}{A_s^*} + \frac{\alpha_g \xi_g \beta}{A_s^* \rho_s} \right) \mathbf{g} \cdot \mathbf{S}_f \end{aligned} \quad (27)$$

Due to the volumetric conservation, the total volumetric flux is obtained as

$$\nabla \cdot \varphi = \nabla \cdot (\alpha_{g,f} \varphi_g + \alpha_{s,f} \varphi_s) = 0 \quad (28)$$

From Eq. (28), the pressure equation can be derived as

$$\begin{aligned} \nabla \cdot \left\{ \left[\alpha_{g,f} \left(\frac{\alpha_g}{A_g^* \rho_g} + \frac{\alpha_s \xi_s \beta}{A_g^* \rho_g \rho_s} \right) + \alpha_{s,f} \left(\frac{\alpha_s}{A_s^* \rho_s} + \frac{\alpha_g \xi_g \beta}{A_s^* \rho_g \rho_s} \right) \right] |\mathbf{S}_f| \nabla^\perp p \right\} \\ = \nabla \cdot \varphi^0 \end{aligned} \quad (29)$$

where φ^0 is the total volumetric flux without the contribution of the pressure gradient and it is expressed as

$$\varphi^0 = \varphi_g^0 + \varphi_s^0 \quad (30)$$

with

$$\begin{aligned} \varphi_g^0 = & \left(\frac{\mathbf{H}_g}{A_g^*} \right)_f \cdot \mathbf{S}_f + \left(\frac{\xi_s \beta}{A_g^* \rho_g} \mathbf{H}_s \right)_f \cdot \mathbf{S}_f - \left(\frac{\xi_s \beta}{A_g^* \rho_g \rho_s} \right) (\nabla p_s)_f \cdot \mathbf{S}_f \\ & + \left(\frac{\alpha_g}{A_g^*} + \frac{\alpha_s \xi_s \beta}{A_g^* \rho_g} \right) \mathbf{g} \cdot \mathbf{S}_f \end{aligned} \quad (31)$$

$$\begin{aligned} \varphi_s^0 = & \left(\frac{\mathbf{H}_s}{A_s^*} \right)_f \cdot \mathbf{S}_f + \left(\frac{\xi_g \beta}{A_s^* \rho_s} \mathbf{H}_g \right)_f \cdot \mathbf{S}_f - \left(\frac{1}{A_s^* \rho_s} \right) (\nabla p_s)_f \cdot \mathbf{S}_f \\ & + \left(\frac{\alpha_s}{A_s^*} + \frac{\alpha_g \xi_g \beta}{A_s^* \rho_s} \right) \mathbf{g} \cdot \mathbf{S}_f \end{aligned} \quad (32)$$

Once the pressure equation is solved, the velocity flux of each phase is corrected using the new pressure field:

$$\varphi_g = \varphi_g^0 - \left(\frac{\alpha_g}{A_g^* \rho_g} + \frac{\alpha_s \xi_s \beta}{A_g^* \rho_g \rho_s} \right) |\mathbf{S}_f| \nabla^\perp p^n \quad (33)$$

$$\varphi_s = \varphi_s^0 - \left(\frac{\alpha_s}{A_s^* \rho_s} + \frac{\alpha_g \xi_g \beta}{A_s^* \rho_g \rho_s} \right) |\mathbf{S}_f| \nabla^\perp p^n \quad (34)$$

Finally, the new velocity fields are reconstructed from the corrected face fluxes using the flux reconstruction method (Passalacqua & Fox, 2011).

3.3. Discretization of the solid phase continuity equation

The boundedness of solid phase fraction is the key issue in the discretization of solid phase continuity equation. The phase fraction

should be bounded between zero and one. Following the approach of Rusche (2002), the boundedness is achieved by re-formulating the continuity equation as

$$\frac{\partial \alpha_s}{\partial t} + \nabla \cdot (\mathbf{U} \alpha_s) + \nabla \cdot (\alpha_g \mathbf{U}_r \alpha_s) = 0 \quad (35)$$

where \mathbf{U} is the mixture velocity, $\mathbf{U} = \alpha_s \mathbf{U}_s + \alpha_g \mathbf{U}_g$, and \mathbf{U}_r is the relative velocity, $\mathbf{U}_r = \mathbf{U}_s - \mathbf{U}_g$.

Special care should be taken on the discretization of the solid phase continuity equation, since the solid volume fraction should not exceed its physical maximum value, i.e., the packing limit $\alpha_{s,\max}$. In MFIX, the effect of solid pressure is explicitly included in the solid phase continuity equation to obtain a solid volume fraction correction equation. This correction equation is solved using the local relaxation to avoid numerical instability (Syamlal, 1998). However, this explicit method is not satisfactory (Passalacqua & Fox, 2011). To make the solid volume fraction below the packing limit, we implicitly include the solid pressure gradient in the solid phase continuity equation. The solid pressure gradient is expressed as the product of the elastic stress modulus $G(\alpha_s)$ and the gradient of solid volume fraction (Bouillard, Lyczkowski, & Gidaspow, 1989):

$$\nabla p_s = G(\alpha_s) \nabla \alpha_s \quad (36)$$

$$G(\alpha_s) = \frac{\partial p_s}{\partial \alpha_s} \quad (37)$$

Finally, the solid phase continuity equation is re-formulated as

$$\frac{\partial \alpha_s}{\partial t} + \nabla \cdot (\mathbf{U}^* \alpha_s) + \nabla \cdot (\alpha_g \mathbf{U}_r^* \alpha_s) - \nabla \cdot \left[\alpha_s \left(\frac{1}{\rho_s A_s^*} \right) G(\alpha_s) \nabla \alpha_s \right] = 0 \quad (38)$$

with

$$\mathbf{U}^* = \mathbf{U} + \alpha_s \left(\frac{1}{\rho_s A_s^*} \right) G(\alpha_s) \nabla \alpha_s$$

$$\mathbf{U}_r^* = \mathbf{U}_r + \left(\frac{1}{\rho_s A_s^*} \right) G(\alpha_s) \nabla \alpha_s$$

Eq. (38) is discretized with the class `fvScalarMatrix` as

```
fvScalarMatrix alphaEqn
(
    fvm::ddt(alpha)
  + fvm::div(phi, alpha)
  + fvm::div(-fvc::flux(-phir, beta), alpha)
  - fvm::laplacian(alpha*f*ppMagf, alpha)
);
```

where `phi` is the flux of the mixture velocity, `phir` is the flux of the relative velocity, `ppMagf` is the face value of the term $G(\alpha_s)/(\rho_s A_s^*)$, `beta` is gas phase volume fraction. The discretization of the Laplacian term is provided in Appendix A.

4. TVD differencing schemes

The finite volume method divides the spatial domain into a number of contiguous control volumes or cells (Fig. 2). Dependent variables and other properties are stored at cell centers in OpenFOAM. For the face f connecting two cells, OpenFOAM designates an owner cell P and a neighbor cell N . The face area vector \mathbf{S}_f is a vector normal to the face pointing out of the owner cell, whose magnitude is that of the face area. The vector \mathbf{d} connects the centers of two adjacent cells.

4.1. Discretization of convection terms

The discretization of convection terms in fluid transport equations has proven to be one of the most troublesome issues. The

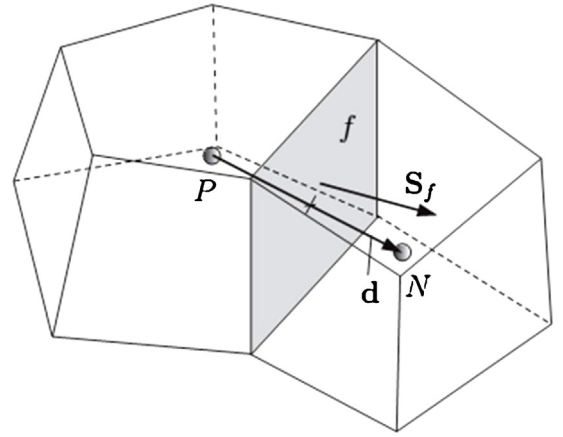


Fig. 2. Geometric parameters of the control volumes (OpenCFD Ltd., 2012).

general convection term is integrated over a control volume after applying the Gauss theorem and linearized as follows:

$$\int_V \nabla \cdot (\rho \mathbf{U} \phi) dV = \int_S d\mathbf{S} \cdot (\rho \mathbf{U} \phi) \approx \sum_f \mathbf{S}_f \cdot (\rho \mathbf{U})_f \phi_f = \sum_f F_f \phi_f \quad (39)$$

where ϕ is a dependent variable and F_f is the face flux, $F_f = \mathbf{S}_f \cdot (\rho \mathbf{U})_f$. The approximation of the face value ϕ_f from the cell-centered values is the key problem in the cell-centered finite volume discretization. Generally, the face value can be determined with the first-order upwind scheme and high-order convection schemes (linear and non-linear schemes). The central differencing scheme is an example of the high-order linear convection schemes (also called κ -schemes), while the flux limiter and normalized-variable approach are the two widely used non-linear high-order convection schemes (Waterson & Deconinck, 2007).

For the central differencing (CD) scheme, the face value ϕ_f is calculated as

$$\phi_{f,CD} = f_{CD} \phi_P + (1 - f_{CD}) \phi_N = f_{CD} (\phi_P - \phi_N) + \phi_N \quad (40)$$

where f_{CD} is a weighting factor, $f_{CD} = \overline{fN}/\overline{PN}$. While the central differencing scheme is second-order accurate, this scheme causes unphysical spatial oscillations in the convection-dominated flows and thus violating the boundedness.

To maintain the boundedness the upwind differencing (UD) scheme determines the face value ϕ_f from the direction of the flow:

$$\phi_{f,UD} = f_{UD} (\phi_P - \phi_N) + \phi_N \quad (41)$$

where the weighting factor f_{UD} equals to 1 if the flux $F_f > 0$ and 0 otherwise. The upwind scheme ensures the unconditional boundedness at the cost of accuracy by introducing numerical diffusion.

To preserve the boundedness with reasonable accuracy, OpenFOAM provides the blended schemes by combining the upwind and central differencing schemes. The face value ϕ_f in the blended schemes is formulated as

$$\phi_f = (1 - \omega) \phi_{f,UD} + \omega \phi_{f,CD} \quad (42)$$

where ω is a blending factor. Clearly, $\omega = 0$ gives the upwind scheme and $\omega = 1$ results in the central differencing.

The discretized convection term along with the discretization of the transient, diffusion, and source terms generates the following linear equation system:

$$a_P \phi_P = \sum_N a_N \phi_N + S_P \quad (43)$$

where S_P is the source term. Jasak (1996) pointed out that only the upwind scheme creates the diagonally equal matrix. The other

Table 2
The TVD flux limiters used in the present study.

Flux limiter	Formulation	References
limitedLinear	$\omega = \max \left[\min \left(\frac{2}{k} r, 1 \right), 0 \right] \quad 0 < k \leq 1$	OpenCFD Ltd. (2012)
Sweby	$\omega = \max[0, \min(kr, 1), \min(r, k)] \quad 1 \leq k \leq 2$	Sweby (1984)
SuperBee	$\omega = \max[0, \min(2r, 1), \min(r, 2)]$	Roe (1985)
vanLeer	$\omega = \frac{r+ r }{1+ r }$	van Leer (1974)
MUSCL	$\omega = \max \left[\min \left(2r, \frac{1}{2}r + \frac{1}{2}, 2 \right), 0 \right]$	van Leer (1979)

convection schemes would create negative matrix coefficients in Eq. (43). The negative coefficients violate the diagonal equality and result in the unbounded solutions. In order to improve the matrix quality, Khosla and Rubin (1974) proposed the deferred correction method to implement the high-order convection schemes. In this method, the part of the convection term corresponding to upwind differencing is implicitly built into the matrix, and the other part is added into the source term. At present, the deferred correction method is not available in the OpenFOAM package and is not used in this work.

4.2. TVD flux limiters in OpenFOAM

The blending factor ω in Eq. (42) can be evaluated in a face-by-face way based on total variation diminishing (TVD) (Harten, 1983; Sweby, 1984) or normalized variable diagram (NVD) (Gaskell & Lau, 1988; Leonard, 1988). The TVD flux limiting procedure creates a differencing scheme which is higher than first-order accurate without the spurious oscillations. The NVD approach stabilizes the central differencing scheme and ensures the solution boundedness by introducing a small amount of numerical diffusion (Jasak, 1996).

The TVD concept has proved to be the most promising method (Jasak, 1996). The performances of TVD schemes are preferentially evaluated in our gas–solid flow simulations. The blending factor ω (also called flux limiter in TVD method) requires the calculation of the variable r which represents the ratio of successive gradients of the interpolated variable. The TVD limiting scheme of the scalar fields calculates r as follows:

$$r = \begin{cases} 2 \frac{\mathbf{d} \cdot (\nabla \phi)_P}{\phi_N - \phi_P} - 1 & F_f > 0 \\ 2 \frac{\mathbf{d} \cdot (\nabla \phi)_N}{\phi_N - \phi_P} - 1 & F_f \leq 0 \end{cases} \quad (\text{scalar } \phi) \quad (44)$$

where $(\nabla \phi)_P$ is the full gradient calculated at the owner cell P . The discretization of gradient term is given in Appendix B.

When discretizing the vector fields, OpenFOAM adopts the V-scheme to maximize the limiting based on the “worst-case” direction where the steepest gradient exists (Greenshields, Weller, Gasparini, & Reese, 2009). The V-scheme is used by adding “V” to the name of a TVD scheme. The calculation of r for a vector field is expressed as

$$r = \begin{cases} 2 \frac{(\phi_N - \phi_P) \cdot (\mathbf{d} \cdot \nabla \phi_P)}{(\phi_N - \phi_P) \cdot (\phi_N - \phi_P)} - 1 & F_f > 0 \\ 2 \frac{(\phi_N - \phi_P) \cdot (\mathbf{d} \cdot \nabla \phi_N)}{(\phi_N - \phi_P) \cdot (\phi_N - \phi_P)} - 1 & F_f \leq 0 \end{cases} \quad (\text{vector } \phi) \quad (45)$$

To bound some scalar fields (e.g., α_s), the name of a scheme can be preceded by the word “limited” and followed with the lower bound and upper bound, respectively. This limited scheme uses the upwind differencing scheme ($\omega = 0$) when the value of ϕ in the cell is not between the lower bound and upper bound. For example, to bound the vanLeer scheme between -2 and 3 , it is specified as “limitedVanLeer -2.0 3.0”. For the scalar fields commonly bounded between 0 and 1 , the limited scheme is renamed as the 01-scheme and is used by adding “01” to the name of the scheme.

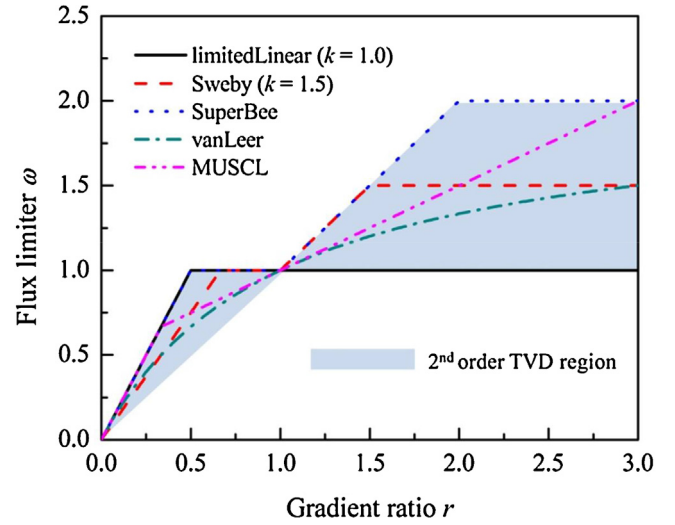


Fig. 3. Profiles of five TVD flux limiters in the Sweby diagram.

Five TVD flux limiters are formulated in Table 2 and also profiled in Fig. 3. The limitedLinear scheme is an OpenFOAM invention and it is the stabilized central differencing scheme with a limiter. For the limiterLinear and Sweby schemes, the coefficient k is needed. When $k = 0$, the limitedLinear and Sweby schemes become the central differencing scheme. For the smooth variation ($r = 1$), all the TVD schemes degenerate the central differencing scheme. The Sweby, SuperBee, vanLeer and MUSCL schemes can use some downwinding, whereas the limitedLinear scheme only combines the upwind and central differencing schemes.

5. Test case descriptions

5.1. Experimental conditions

The fluidized bed setup with uniform gas feed (Fig. 4a) is simulated which was experimentally studied by Taghipour, Ellis, and Wong (2005). The bed material is made of spherical glass beads with a mean diameter of 2.8×10^{-4} m and a density of 2500 kg/m^3 . The experiment started with a static bed height of 0.4 m and a solid volume fraction of 0.6 . The fluidized bed with a central jet (Fig. 4b) used by Gidaspow, Lin, and Seo (1983) is also simulated which is filled with the particles with a mean diameter of 5.0×10^{-4} m. The initial bed height is up to 0.2922 m . At the center of the bottom, a rectangular orifice of $0.0127 \text{ m} \times 0.0381 \text{ m}$ allows a central jet into the bed, while the rest of the bottom is uniformly fed by air at the minimum fluidization velocity of 0.282 m/s .

5.2. Simulation setup

The 2D computational domain is employed for the fluidized bed with uniform gas feed and is discretized with $11,200$ rectangular cells, corresponding to a grid width of 0.005 m . The grid size is the same as that adopted by Taghipour et al. (2005). The fluidized bed

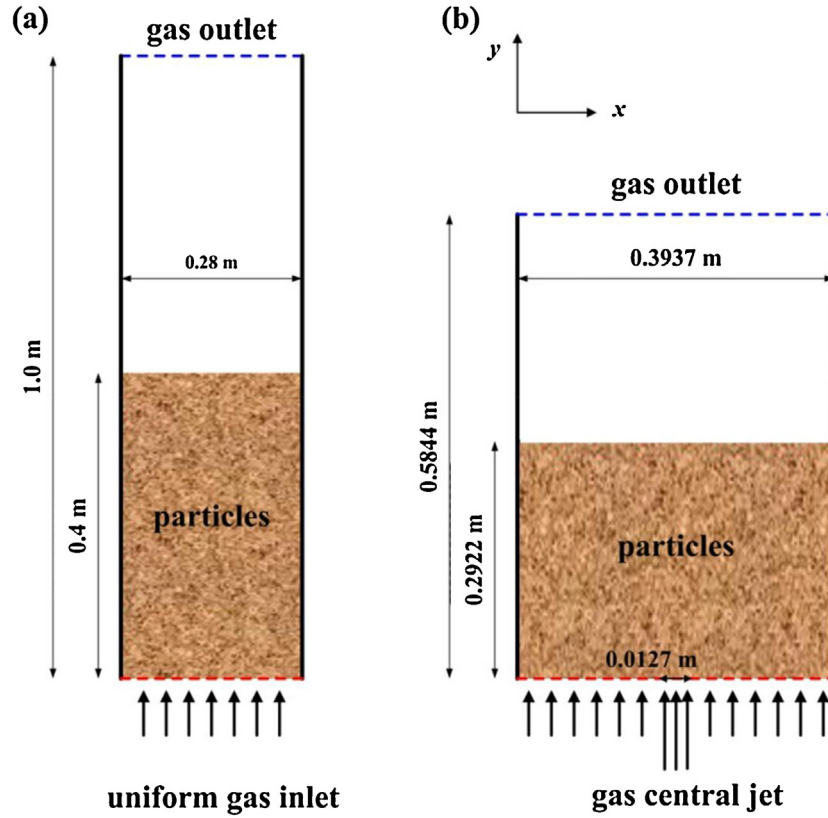


Fig. 4. Schematics of fluidized beds with (a) uniform gas feed and (b) a central jet.

Table 3

Parameters for the simulation of fluidized bed with uniform gas feed.

Parameter	Value
Bed width, m	0.28
Bed height, m	1.0
Bed depth, m	0.025
Initial bed height, m	0.4
Initial solid packing	0.6
Superficial gas velocity, m/s	0.38, 0.46
Min. fluidization velocity, m/s	0.065
Gas density, kg/m ³	1.2
Gas kinematic viscosity, m ² /s	1.4×10^{-5}
Particle diameter, m	2.8×10^{-4}
Particle density, kg/m ³	2500
Particle–particle restitution coefficient	0.8
Specularity coefficient	0.5
$\alpha_{s,max}$	0.65
$\alpha_{s,min}$	0.50

Table 4

Parameters for the simulation of fluidized bed with a central jet.

Parameter	Value
Bed width, m	0.3937
Bed height, m	0.5844
Bed depth, m	0.0381
Jet orifice width, m	0.0127
Initial bed height, m	0.2922
Voidage at min. fluidization	0.402
Jet inlet velocity, m/s	1.511, 3.550
Minimum fluidization velocity, m/s	0.282
Gas density, kg/m ³	1.2
Gas kinematic viscosity, m ² /s	1.4×10^{-5}
Particle diameter, m	5.0×10^{-4}
Particle density, kg/m ³	2610
Particle–particle restitution coefficient	0.8
Specularity coefficient	0.5
$\alpha_{s,max}$	0.65
$\alpha_{s,min}$	0.63

with a central jet is simulated using 2D domain with 124 cells in the x-direction and 108 cells in the y-direction. The parameters for the two fluidized bed cases are summed up in Tables 3 and 4. The transient simulations are conducted with an adaptive time step method. The time step is adapted with the Courant number which is defined in Eq. (46):

$$Co = \frac{\Delta t |\mathbf{U}_r|}{\Delta x} \quad (46)$$

where Δt is the time step, $|\mathbf{U}_r|$ is the magnitude of the relative velocity through the cell and Δx is the cell size in the direction of the relative velocity. The Courant number is not allowed to be larger than 0.1. The maximum value of time step is set to 1.0×10^{-4} s.

The inlets of fluidized beds are made impenetrable for the solid phase by setting the velocity of solid phase to be zero m/s. The uniform gas velocity is employed at the inlets. The pressure at

the outlet is specified as the atmospheric pressure. The no-slip boundary condition is applied at the wall for the gas phase velocity, while the partial slip boundary condition proposed by Johnson and Jackson (1987) is used for the solid phase in the two test cases. At the wall surfaces, the particle velocity is determined with the following correlation:

$$\nabla \tau_{s,w} = -\frac{\pi}{6} \frac{\alpha_s}{\alpha_{s,max}} \psi p_s g_0 \sqrt{3\Theta} \mathbf{U}_{s,w} \quad (47)$$

where ψ is the specularity coefficient. When the value of ψ is zero, the perfect smooth wall exists, on the contrary unity represents the no-slip wall boundary condition.

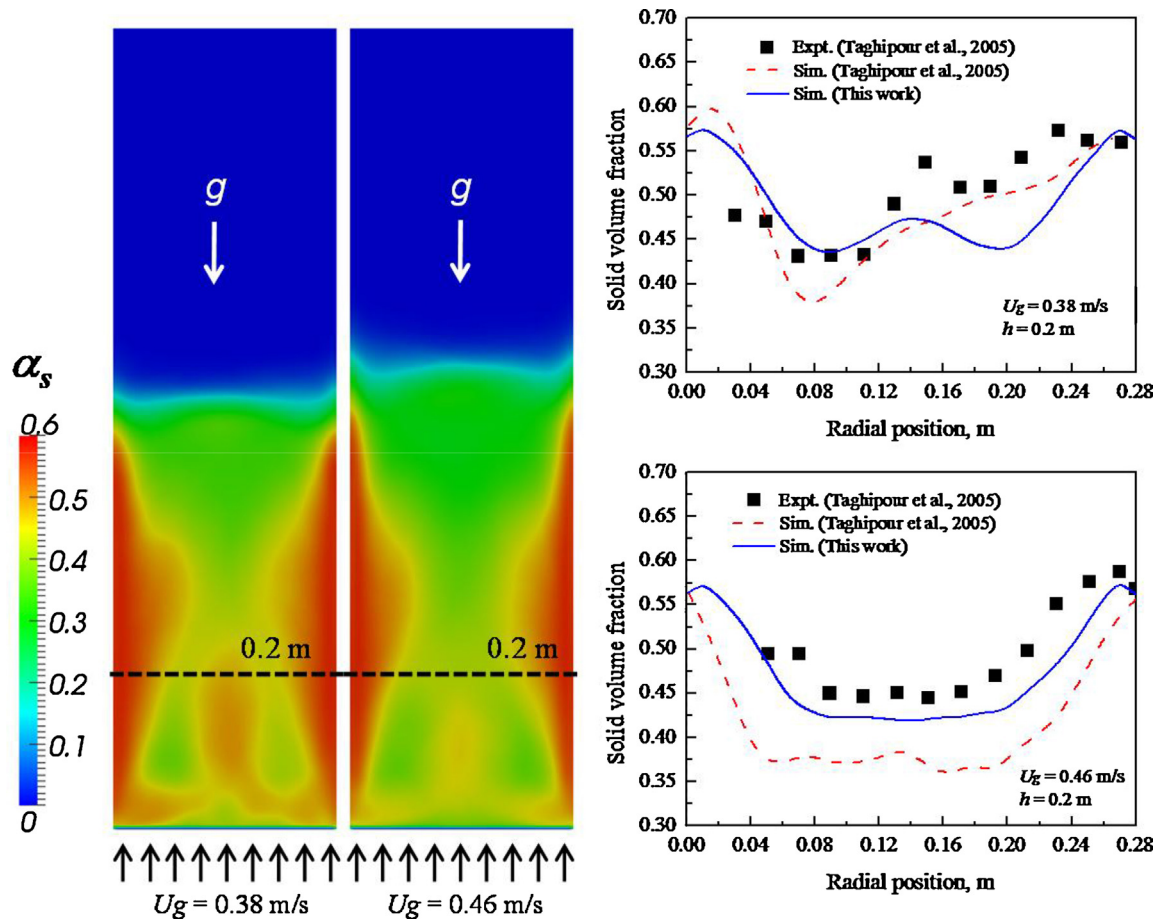


Fig. 5. Comparison of the simulated time-averaged solid volume fractions with experimental data at two superficial gas velocities.

6. Results and discussion

6.1. Model validations

6.1.1. Bubbling fluidized bed with uniform gas feed

Fig. 5 shows the comparison of the simulated profiles of solid volume fraction with the experimental results measured at two gas inlet velocities (0.38 and 0.46 m/s). The simulations are run for 60 s

of real time. The time-averaged results are computed over the last 55 s of the simulation. It is tested that the time period for averaging is enough to obtain the time-averaged results. Here, the limited-Linear, limitedLinearV and limitedLinear01 schemes are adopted for the convection terms in the granular temperature transport equation, the phase momentum equations and the solid phase continuity equation, respectively. The simulated results of solid volume fraction give reasonable agreement with the experimental data.

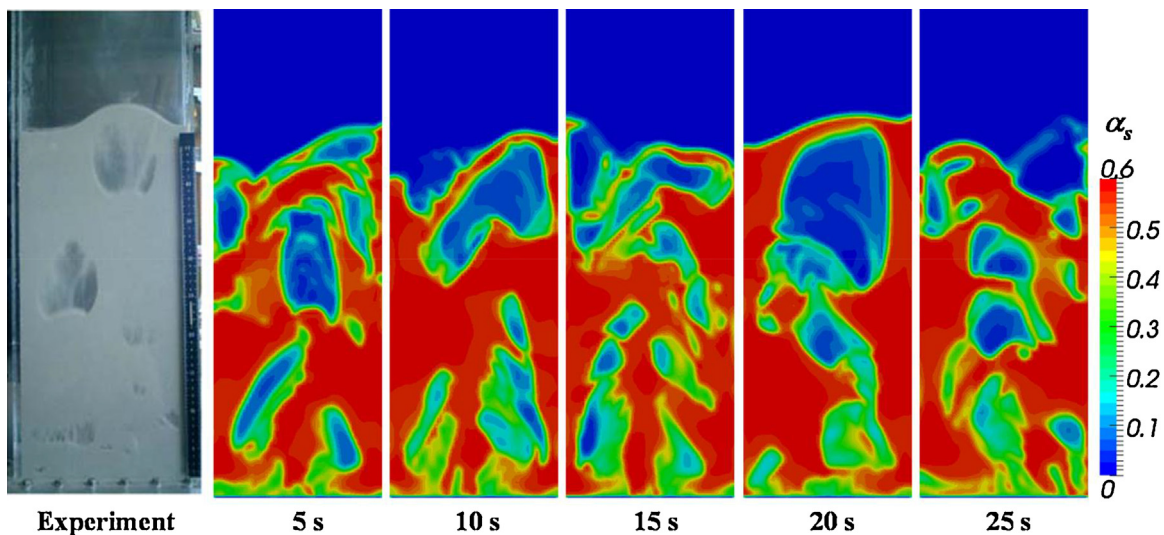


Fig. 6. Snapshots of experimental and simulated solid volume fraction distributions ($U_g = 0.46$ m/s).

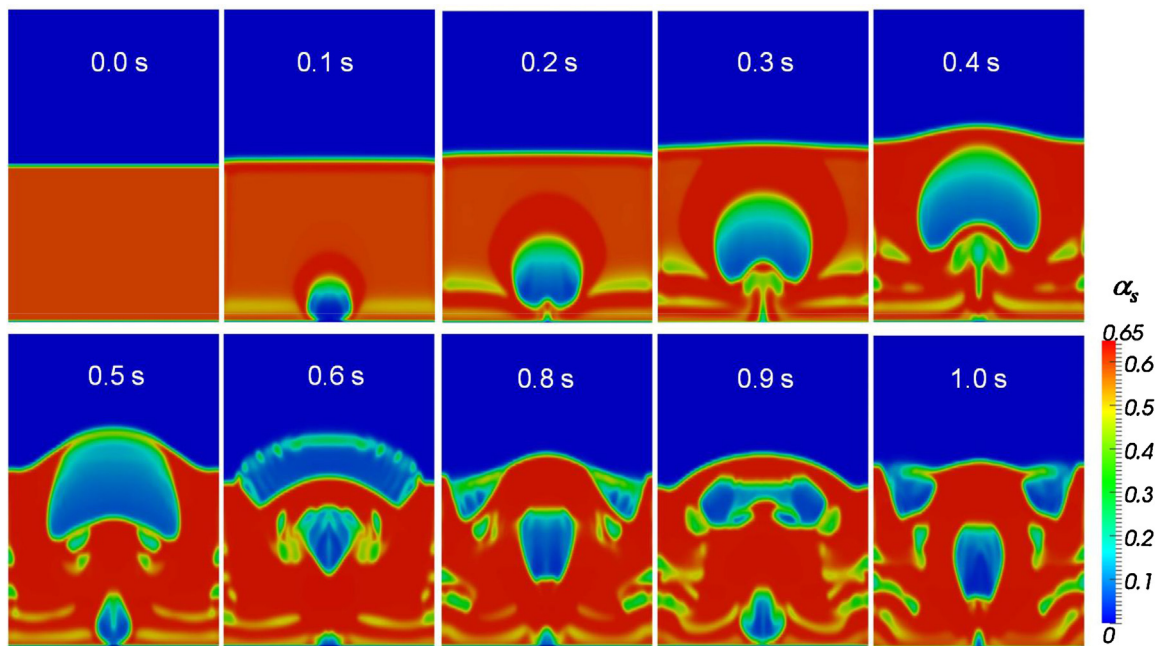


Fig. 7. Simulated contours of the fluidized bed with a jet velocity of 3.55 m/s.

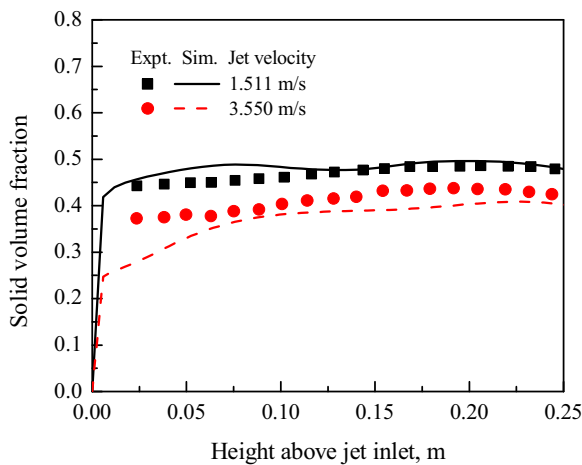


Fig. 8. Comparison of the simulated profiles of solid volume fraction with the experimental data from Gidaspow et al. (1983).

At the gas inlet velocity of 0.38 m/s, the dense particle flows are formed in the bed center region. While increasing gas inlet velocity to 0.46 m/s, the gas-particle flows are more developed which results in the flat radial profiles at the bed center.

As a comparison, the simulated results collected from Taghipour et al. (2005) are given in Fig. 5. For the inlet velocity of 0.46 m/s, the simulated solid concentrations in this work show better predictions than those from Taghipour et al. (2005). Moreover, the typical core-annular flow structures in the fluidized bed are properly captured where high particle concentrations are found in the near wall region and low concentrations in the bed center. In Fig. 6, the detailed bed hydrodynamics is shown by the contours of solid volume fraction and the snapshot from Taghipour et al. (2005). Both experimental and simulated results indicate that the small gas bubbles are mainly formed at the bed bottom, whereas the large bubbles exist at the upper part due to the bubble coalescence.

6.1.2. Bubbling fluidized bed with a central gas jet

The fluidized bed with a central gas jet is chosen to investigate the simulated bubble behaviors. Fig. 7 displays the evolution of the

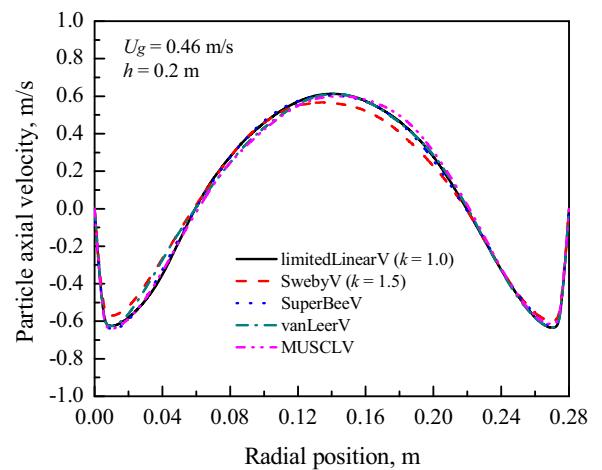


Fig. 9. Effect of discretization of the phase velocity convection terms on the time-averaged results of particle axial velocity.

first bubble injected by the central jet. Gradually, the first bubble is formed due to the more compact particle layer around it. The first bubble expands as it rises through the bed. The bubble shapes are quite similar to the ones simulated by Passalacqua and Marmo (2009). The typical rounded kidney shape of the first bubble is well captured at $t = 0.4$ s.

Fig. 8 gives the comparison of the simulated profiles of solid volume fraction with the experimental data from Gidaspow et al. (1983). The fine agreement between experiments and simulations is also achieved for two different jet velocities. In summary, our gas-solid solver is well capable of describing the solid particle concentrations and bubble behaviors in the different fluidized beds.

6.2. Discretization of phase velocity convection terms

For the velocity fields (U_s and U_g), the V-schemes of the five TVD flux limiters are employed for all phase velocity convection terms. The gradient terms are discretized with the non-limited Gauss linear scheme. For the fluidized bed with uniform gas feed,

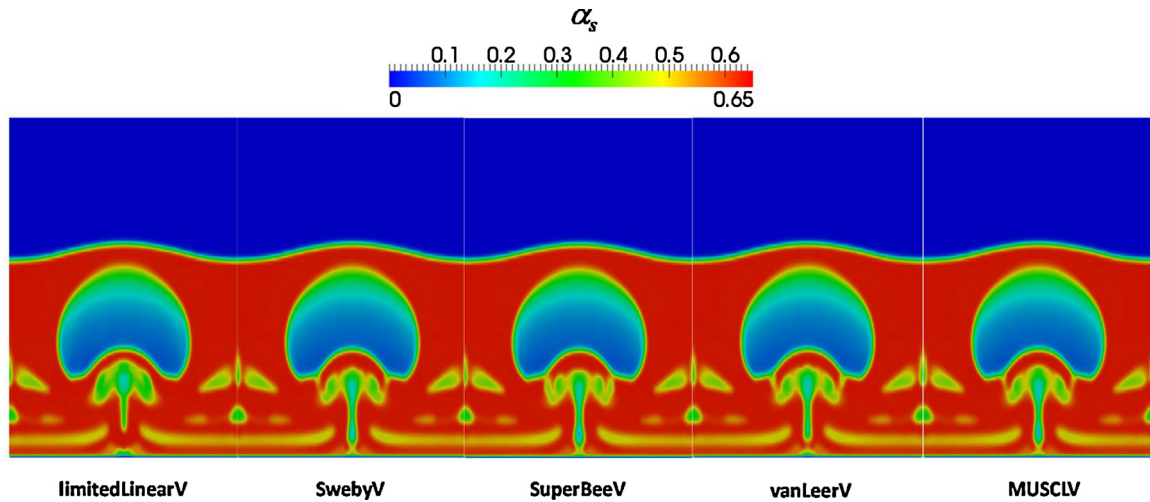


Fig. 10. The bubble shapes simulated with five TVD schemes for the phase velocity convection terms ($t = 0.4$ s).

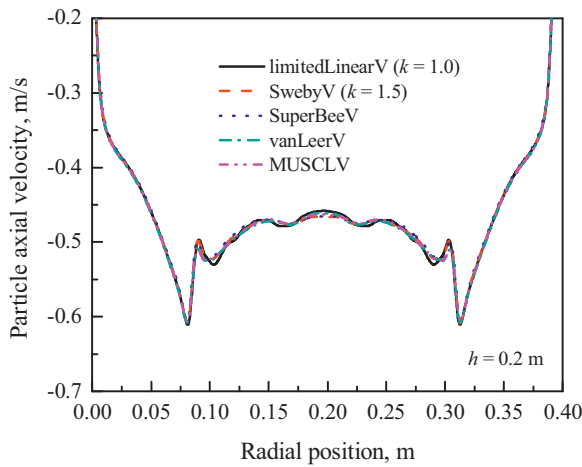


Fig. 11. Effect of discretization of the phase velocity convection terms on the instantaneous particle axial velocities ($t = 0.4$ s).

the simulated radial profiles of particle axial velocity are compared for the five schemes (Fig. 9). It is observed that the particles flow up in the bed center, while the downward flow exists in the near wall region. These particle flow patterns can be captured with the five schemes. Furthermore, the difference in the profiles of particle axial velocity is not pronounced among these schemes.

The effect of the discretization of phase velocity convection terms on the first bubble shape is studied in the fluidized bed with a central jet. Fig. 10 presents the comparison of the first bubbles simulated using the different schemes ($t = 0.4$ s). Clearly, the bubble shapes predicted with the five schemes are almost identical. Fig. 11 indicates the simulated instantaneous profiles of particle axial velocity at the vertical position of 0.2 m in the bed. Also, quite similar velocity profiles are observed among the five schemes. Since the five TVD schemes give quite similar results, it can be inferred that the gradient ratio r calculated with Eq. (45) is around 1 and the five TVD flux limiters become the central differencing scheme (see Fig. 3). From the above study, little care can be taken on the choice of the TVD convection schemes used for phase velocity convection terms.

6.3. Discretization of solid volume fraction convection terms

6.3.1. Comparison of the TVD schemes

The 01-schemes of the TVD flux limiters, i.e., limitedLinear01, Sweby01 and vanLeer01, are applied for the solid volume fraction

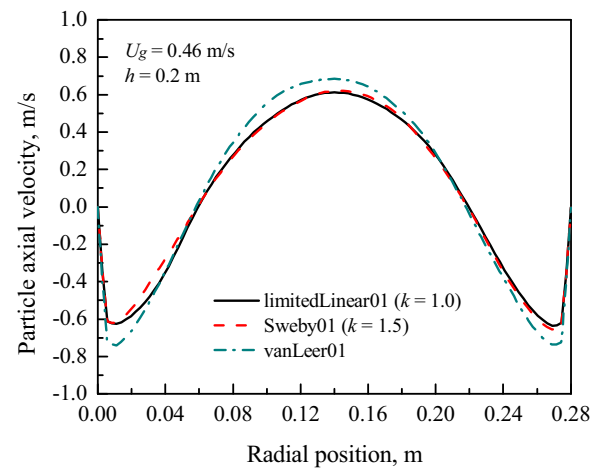


Fig. 12. Effect of discretization of the solid volume fraction convection terms on the time-averaged results of particle axial velocity.

convection terms. The phase velocity convection terms are discretized with the limitedLinearV scheme. The limitedLinear scheme is used for the granular temperature convection term. The gradient terms are discretized with the non-limited Gauss linear scheme. Fig. 12 shows the comparison of the time-averaged profiles of particle axial velocity simulated with the limitedLinear01, Sweby01 and vanLeer01 schemes. The three schemes have no significant differences in the simulated profiles. Fig. 13 presents the instantaneous contours of the first bubble predicted with the limitedLinear01, Sweby01 and vanLeer01 schemes at $t = 0.3$ s. Also, the bubble shapes predicted with the three schemes are quite similar to each other. However, the wakes following the first bubbles are different. Due to the gas jet with the high velocity, the strong convection takes place in the wakes. In such strong convective flows, the differences among the schemes would be more obvious.

It is found that the SuperBee01 and MUSCL01 schemes for solid volume fraction convection terms destroy the solution convergence. By using the SuperBee01 and MUSCL01 schemes, solid volume fraction cannot be effectively limited to be lower than the packing limit. When solid volume fraction reaches the packing limit, the radial distribution function g_0 in Eq. (11) becomes infinite which results in the solution divergence.

In Fig. 14 the face values of SuperBee01 scheme are compared with those of limitedLinear01 scheme. The one-dimensional variation of ϕ is used as an illustrative example. Here, the variable ϕ

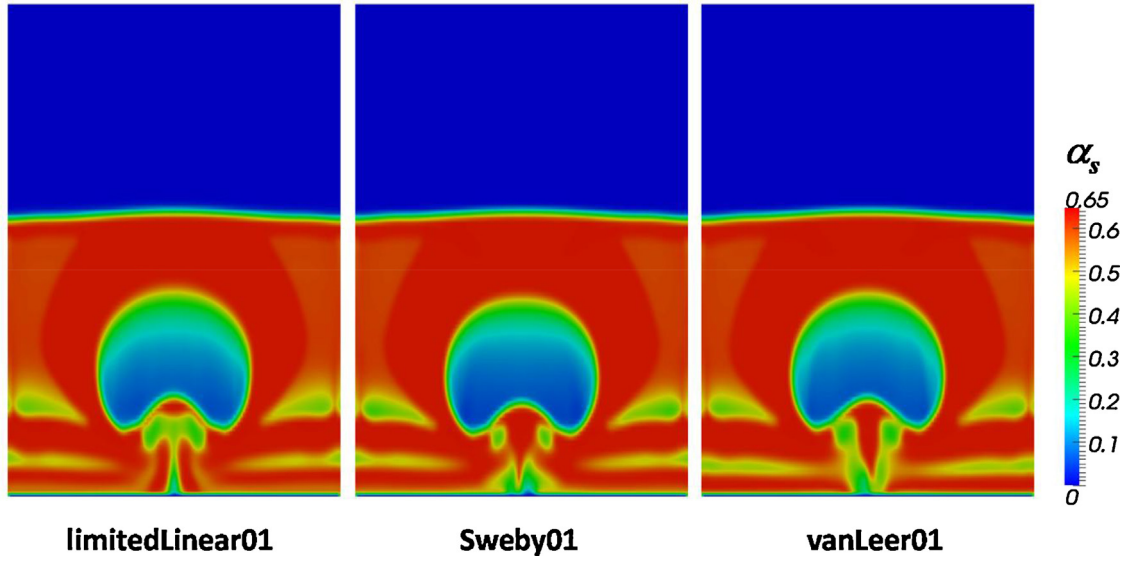


Fig. 13. The bubble shapes simulated with different schemes for the solid volume fraction convection terms ($t = 0.3$ s).

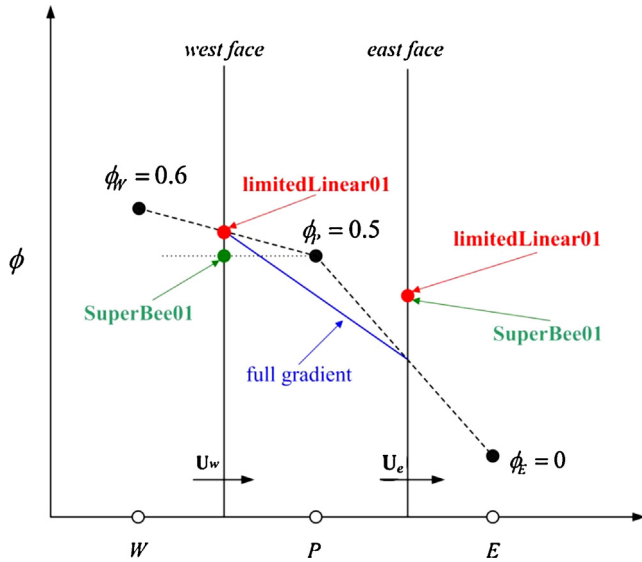


Fig. 14. An example for comparing the limitedLinear01 and SuperBee01 schemes.

represents the solid volume fraction. The cell centers are uniformly located in the grid. In this case, the gradient ratio r has a value of 5 at the west face. Therefore, at the west face, the SuperBee01 scheme becomes the downwind scheme, while the limitedLinear01 scheme uses the central differencing. At the east face, the gradient ratio r is 0.2 and the same face value is interpolated with the SuperBee01 and limitedLinear01 schemes:

$$\phi_e = 0.8\phi_p + 0.2\phi_E \quad (48)$$

The first-order Euler implicit scheme is employed to discretize the transient term of solid volume fraction:

$$\int_V \frac{\partial \phi}{\partial t} dV = \frac{\partial}{\partial t} \int_V \phi dV = V_P \frac{\phi_P - \phi_P^o}{\Delta t} \quad (49)$$

where V_P is the volume of cell P , ϕ^o is the value at the previous time step, and Δt is the time step. By discretizing the transient, convection and diffusion terms in Eq. (38), the discretized solid phase continuity equation is given in the following form:

$$a_P \phi_P = a_W \phi_W + a_E \phi_E + S_P \quad (50)$$

The matrix coefficients in Eq. (50) for the limitedLinear01 scheme are listed as

$$a_P = \frac{V_P}{\Delta t} + 0.5F_W + 0.8F_E + \frac{\Gamma_W |S_W|}{|d|} + \frac{\Gamma_E |S_E|}{|d|} \quad (51a)$$

$$a_W = \frac{\Gamma_W |S_W|}{|d|} - 0.5F_W \quad (51b)$$

$$a_E = \frac{\Gamma_E |S_E|}{|d|} - 0.2F_E \quad (51c)$$

$$S_P = \frac{V_P}{\Delta t} \phi_P^o \quad (51d)$$

where F_E and F_W are the flux at the east and west face, respectively, $F_E > 0$ and $F_W < 0$. Γ is the coefficient in the diffusion term. By using the SuperBee01 scheme, the matrix coefficients in Eq. (50) are given as

$$a_P = \frac{V_P}{\Delta t} + F_W + 0.8F_E + \frac{\Gamma_W |S_W|}{|d|} + \frac{\Gamma_E |S_E|}{|d|} \quad (52a)$$

$$a_W = \frac{\Gamma_W |S_W|}{|d|} \quad (52b)$$

$$a_E = \frac{\Gamma_E |S_E|}{|d|} - 0.2F_E \quad (52c)$$

$$S_P = \frac{V_P}{\Delta t} \phi_P^o \quad (52d)$$

Comparing Eq. (52a) with Eq. (51a), the matrix coefficient of ϕ_P obtained using the SuperBee01 scheme is more prone to the negative value, especially when using large time step. According to Jasak (1996) and Versteeg and Malalasekera (2007), the negative matrix coefficients would violate the solution boundedness and destroy numerical stability. From Eq. (38), the face flux F_E and F_W include the contribution from the solid pressure gradient, which could make the face flux much larger and diverge the solutions more readily. The same stability problem is also encountered by the MUSCL01 scheme. To further verify the above analysis, the SuperBee01 and MUSCL01 schemes are modified by reducing the influence of the downwind differencing. It is found that the stable solutions can be achieved with the two modified schemes. Fig. 15 shows the contours of the first bubble in the bed with a central jet.

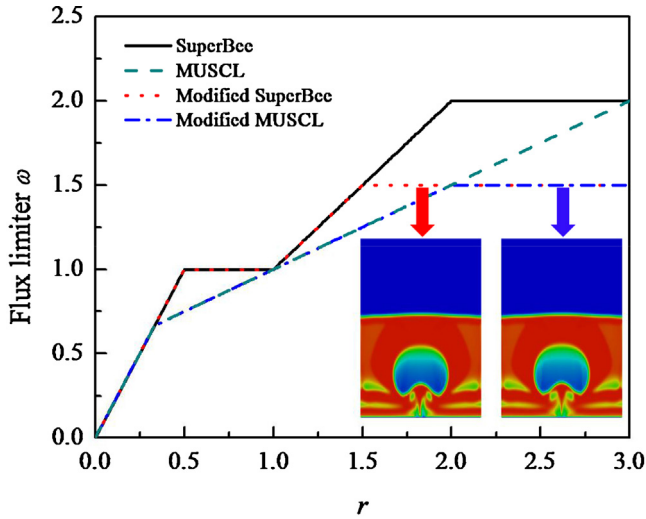


Fig. 15. Simulated results with the modified SuperBee01 and MUSCL01 schemes.

6.3.2. The influence of gradient discretization scheme

From Eqs. (44) and (45), the gradient discretization scheme plays an important role in the calculation of the gradient ratio r used in TVD convection schemes. By combining with the limitedLinear01 convection scheme, the non-limited, cellLimited, faceLimited and cellMDLimited versions of Gauss linear scheme are investigated for the gradient term of solid volume fraction. The gradient limiting method makes the pure Gauss linear scheme more stable (OpenCFD Ltd., 2012). The different gradient limiting methods are illustrated in Appendix B.

Fig. 16 displays the simulated bubbles with different gradient limiting methods. The non-limited, cellLimited and cellMDLimited methods produce the rounded kidney bubbles, whereas the unphysical pointed shape (Guenther & Syamlal, 2001) is predicted by the faceLimited method. The pointed bubble is also predicted with upwind scheme. A large amount of numerical diffusion is introduced by the upwind scheme. It is inferred the faceLimited method generates small values of the gradient ratio r , and makes the limitedLinear01 scheme more diffusive (see Fig. 3). Hence, the faceLimited scheme should not be applied for the gradient terms, although it provides good numerical stability. The cellLimited and cellMDLimited gradient schemes are recommended to achieve the simulations with higher accuracy and better stability.

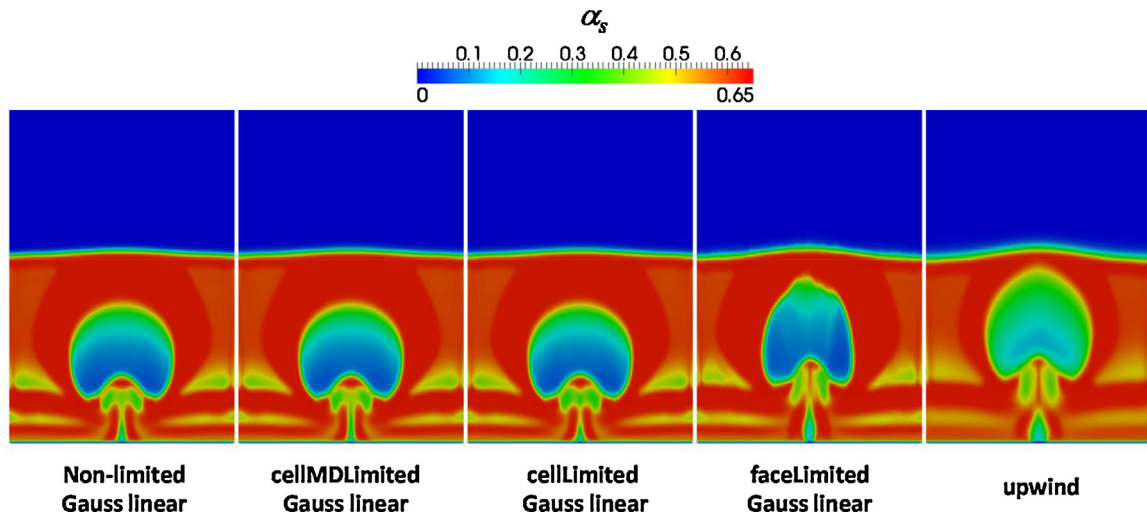


Fig. 16. The bubble shapes simulated with different schemes for the gradient term of solid volume fraction.

7. Conclusions

The gas–solid OpenFOAM solver based on two-fluid model is validated by simulating two different fluidized beds. The simulated results of solid volume fraction are in good agreement with the experimental data from the literature. The first bubble in the bed with a central jet is well predicted. The gas–solid solver is capable of predicting the gas–solid flows in the two fluidized beds.

For the discretization of the phase velocity convection terms, the five TVD schemes give very similar time-averaged results of particle axial velocity in the bed with uniform gas feed. In the bed with a central jet, the predicted first bubble shapes and particle velocity profiles are also quite similar among the five schemes. The five schemes have no stability problem in the discretization of phase velocity convection terms. Therefore, the choice of the TVD schemes needs little attention when discretizing the phase velocity convection terms.

When discretizing the solid volume fraction convection terms, the limitedLinear01, Sweby01 and vanLeer01 schemes give the stable solutions in the two test cases. The three schemes have no significant differences in the simulated results. However, in the discretized solid phase continuity equation, the SuperBee01 and MUSCL01 schemes generate the negative matrix coefficients and destroy the solution convergence. The faceLimited gradient scheme makes the convection scheme more diffusive. The cellLimited and cellMDLimited schemes have high accuracy and good stability.

Acknowledgements

Yefei Liu would like to acknowledge the financial support from China Scholarship Council (Grant no. 2011621027) and TUM Graduate School. Florian Habla is also acknowledged for his helpful discussion on this work.

Appendix A. Laplacian scheme

The Laplacian term is integrated over a control volume and discretized as follows:

$$\int_V \nabla \cdot (\Gamma \nabla \phi) dV = \int_S d\mathbf{S} \cdot (\Gamma \nabla \phi) \approx \sum_f \Gamma_f \mathbf{S}_f \cdot (\nabla \phi)_f \quad (\text{A.1})$$

The face gradient in Eq. (A.1) is implicitly discretized when the vector \mathbf{d} between the own cell center and the neighboring cell center is orthogonal to the face:

$$\mathbf{S}_f \cdot (\nabla \phi)_f = |\mathbf{S}_f| \frac{\phi_N - \phi_P}{|\mathbf{d}|} \quad (\text{A.2})$$

Appendix B. Gradient schemes

The gradient term is usually integrated over a control volume and discretized as

$$\int_V \nabla \phi dV = \int_S \mathbf{dS} \phi \approx \sum_f \mathbf{S}_f \phi_f \quad (\text{B.1})$$

For incompressible flows the face value ϕ_f is usually calculated using the linear interpolation (i.e., Gauss linear scheme in OpenFOAM). The cell-centered gradient (full gradient) at the cell P is calculated as

$$(\nabla \phi)_P = \frac{1}{V_P} \sum_f \mathbf{S}_f \phi_f \quad (\text{B.2})$$

where V_P is the volume of the cell P .

Gradient reconstruction may lead to the unbounded face value (Fig. B.1). The gradient limiting is necessary for the bounded variable. There are two types of gradient limiting available in OpenFOAM, i.e., cell limiting and face limiting. For the cell limited gradient, as shown in Fig. B.1, the neighbor cell with the maximum value of ϕ and the neighbor cell with the minimum value of ϕ are firstly found. Then, the cell limited gradient is calculated as

$$(\nabla \phi)_{P, \text{cellLimited}} = \begin{cases} \frac{H}{D} (\nabla \phi)_P & D > H \\ \frac{h}{d} (\nabla \phi)_P & h < d \end{cases} \quad (\text{B.3})$$

The parameters h and H are calculated as follows:

$$h = \phi_{\min} - \phi_P$$

$$H = \phi_{\max} - \phi_P$$

The parameters d and D are calculated based on the full gradient:

$$d = \mathbf{d}_{f,i} \cdot (\nabla \phi)_P$$

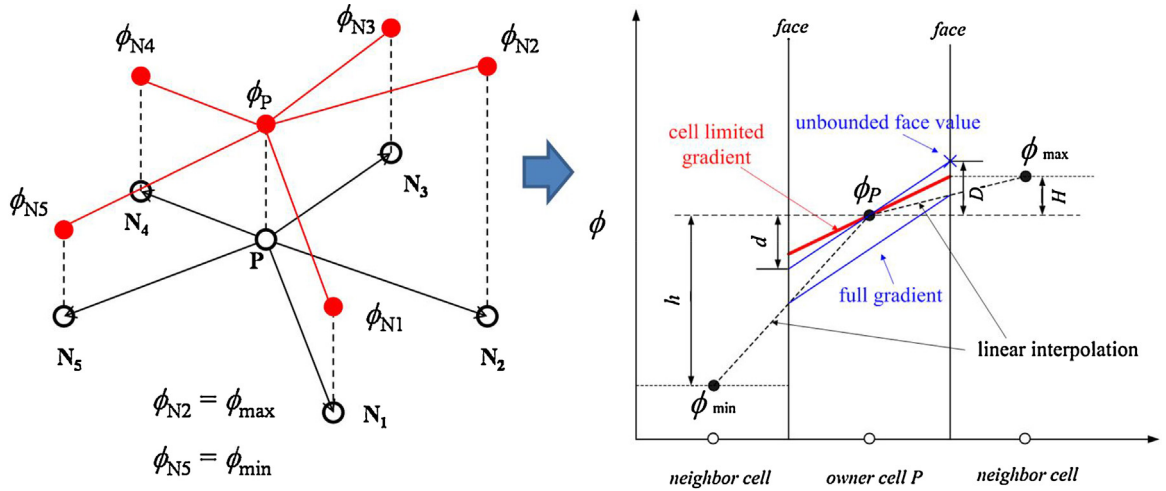


Fig. B.1. The cell limited gradient scheme in OpenFOAM.

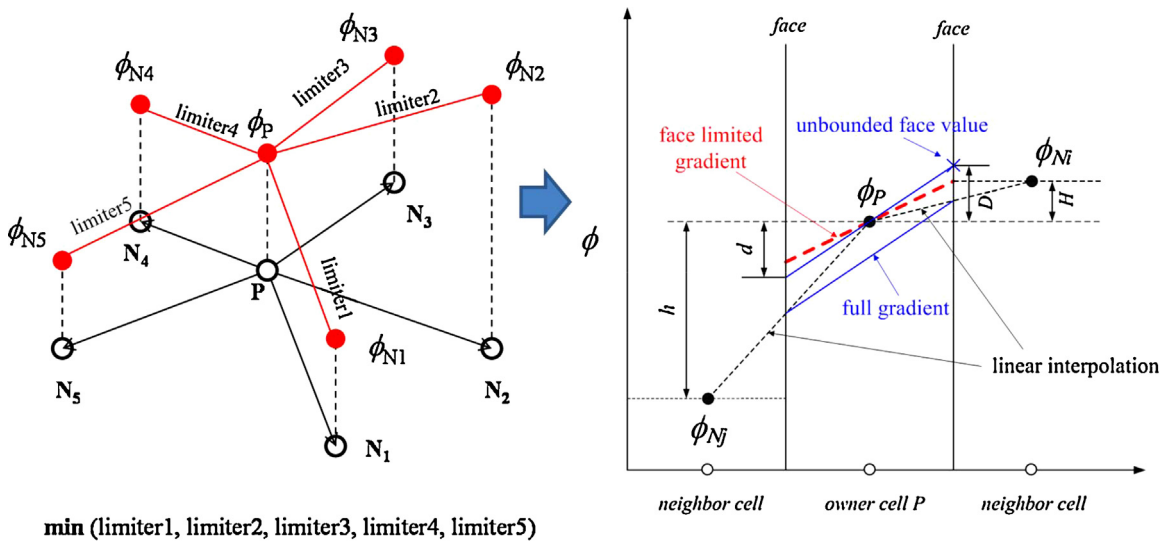


Fig. B.2. The face limited gradient scheme in OpenFOAM.

$$D = \mathbf{d}_{f,j} \cdot (\nabla \phi)_p$$

where $\mathbf{d}_{f,i}$ is the vector connecting the owner cell center and its bounding face center i . $\mathbf{d}_{f,j}$ is the vector connecting the owner cell center and its bounding face center j .

The multi-dimensional cell-limited gradient scheme (i.e., cellMDLimited scheme in OpenFOAM) applies the gradient limiter for each face and is expressed as

$$(\nabla \phi)_{p,\text{cellMDLimited}} = \begin{cases} (\nabla \phi)_p + \frac{\mathbf{d}_f(H-D)}{[\mathbf{d}_f \cdot (\nabla \phi)_p]^2} & D > H \\ (\nabla \phi)_p + \frac{\mathbf{d}_f(h-d)}{[\mathbf{d}_f \cdot (\nabla \phi)_p]^2} & h < d \end{cases} \quad (\text{B.4})$$

Instead of using the cell neighbors having the maximum and minimum values of ϕ , the face limited gradient is calculated by multiplying the minimum face limiter with the full gradient. For each face bounding the owner cell, each face limiter is calculated (Fig. B.2). By comparing each face limiter, the minimum face limiter is chosen to calculate the face limited gradient.

References

- Asegehegn TW, Schreiber M, Krauz HJ. Numerical simulation and experimental validation of bubble behavior in 2D gas–solid fluidized bed with immersed horizontal tubes. *Chemical Engineering Science* 2011;66:5410–27.
- Bouillard J, Lyczkowski R, Gidaspow D. Porosity distributions in a fluidized bed with immersed obstacle. *AIChE Journal* 1989;35:908–22.
- Braun MPS, Mineto AT, Navarro HA, Cabezas-Gómez L, Silva RC. The effect of numerical diffusion and the influence of computational grid over gas–solid two-phase flow in a bubbling fluidized bed. *Mathematical and Computer Modeling* 2010;52:1390–402.
- Chalermisinsuwan B, Kuchonthara P, Piumsomboon P. CFD modeling of tapered circulating fluidized bed reactor risers: Hydrodynamic descriptions and chemical reaction responses. *Chemical Engineering and Processing* 2010;49:1144–60.
- Ding JM, Gidaspow D. A bubbling fluidization model using kinetic theory of granular flow. *AIChE Journal* 1990;36:523–38.
- Enwald H, Peirano E, Almstedt AE. Eulerian two-phase flow theory applied to fluidization. *International Journal of Multiphase Flow* 1996;22:21–66.
- Ergun S. Fluid flow through packed columns. *Chemical Engineering Progress* 1952;48:89–95.
- Gaskell PH, Lau AKC. Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm. *International Journal for Numerical Methods in Fluids* 1988;8:617–41.
- Gidaspow D. Multiphase flow and fluidization: Continuum and kinetic theory descriptions. New York: Academic Press; 1994.
- Gidaspow D, Lin C, Seo YC. Fluidization in two-dimensional beds with a jet. 1. Experimental porosity distributions. *Industrial and Engineering Chemistry Fundamentals* 1983;22:187–93.
- Greenshields CJ, Weller HG, Gasparini L, Reese JM. Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows. *International Journal for Numerical Methods in Fluids* 2009;63:1–21.
- Guenther C, Syamlal M. The effect of numerical diffusion on simulation of isolated bubbles in a gas–solid fluidized bed. *Powder Technology* 2001;116:142–54.
- Harten A. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics* 1983;49:357–93.
- Herzog N, Schreiber M, Egbers C, Krautz HJ. A comparative study of different CFD-codes for numerical simulation of gas–solid fluidized bed hydrodynamics. *Computers and Chemical Engineering* 2012;39:41–6.
- Hulme I, Clavelle E, van der Lee L, Kantzas A. CFD modeling and validation of bubble properties for a bubbling fluidized bed. *Industrial and Engineering Chemistry Research* 2005;44:4254–66.
- Ishii M. Thermo-fluid dynamic theory of two-phase flow. Paris: Eyrolles; 1975.
- Issa R. Solution of the implicitly discretized fluid flow equations by operator-splitting. *Journal of Computational Physics* 1985;62:40–65.
- Jasak H. Error analysis and estimation for the finite volume method with applications to fluid flows [Ph.D. thesis]. London: Imperial College of Science, Technology and Medicine; 1996.
- Jenkins JT, Savage SB. A theory of the rapid flow of identical, smooth, nearly elastic spherical particles. *Journal of Fluid Mechanics* 1983;130:187–202.
- Johnson PC, Jackson R. Frictional–collisional constitutive relations for granular materials, with application to plane shearing. *Journal of Fluid Mechanics* 1987;176:67–93.
- Johnson PC, Nott P, Jackson R. Frictional–collisional equations of motion for particulate flows and their application to chutes. *Journal of Fluid Mechanics* 1990;210:501–35.
- Khosla PK, Rubin SG. A diagonally dominant second-order accurate implicit scheme. *Computers and Fluids* 1974;2:207–9.
- Leonard BP. Simple high-accuracy resolution program for convective modelling of discontinuities. *International Journal for Numerical Methods in Fluids* 1988;8:1291–318.
- Lettieri P, Cammarata L, Micale GDM, Yates J. CFD simulations of gas fluidized beds using alternative Eulerian–Eulerian modeling approaches. *International Journal of Chemical Reactor Engineering* 2003;1:1–21.
- Li TW, Grace J, Bi XT. Study of wall boundary condition in numerical simulations of bubbling fluidized beds. *Powder Technology* 2010;203:447–57.
- Li TW, Dietiker J-F, Zhang YM, Shahnaim M. Cartesian grid simulations of bubbling fluidized beds with a horizontal tube bundle. *Chemical Engineering Science* 2011;66:6220–31.
- Lu HL, Gidaspow D. Hydrodynamics of binary fluidization in a riser: CFD simulation using two granular temperatures. *Chemical Engineering Science* 2003;58:3777–92.
- Lu HL, He YR, Liu WT, Ding JM, Gidaspow D, Bouillard J. Computer simulations of gas–solid flow in spouted beds using kinetic–frictional stress model of granular flow. *Chemical Engineering Science* 2004;59:865–78.
- Lun CKK, Savage SB, Jeffrey DJ, Chepurin N. Kinetic theories for granular flow: Inelastic particles in Couette flow and slightly inelastic particles in a general flowfield. *Journal of Fluid Mechanics* 1984;140:223–56.
- McKeen T, Pugsley T. Simulation and experimental validation of a freely bubbling bed of FCC catalyst. *Powder Technology* 2003;129:139–52.
- Oliveira JP, Issa RI. Numerical aspects of an algorithm for Eulerian simulation of two-phase flows. *International Journal for Numerical Methods in Fluids* 2003;43:1177–98.
- OpenCFD Ltd. OpenFOAM 2.1.1 user's guide. United Kingdom: OpenCFD Ltd; 2012.
- Park IK, Cho HK, Yoon HY, Jeong JJ. Numerical effects of the semi-conservative form of momentum equations for multi-dimensional two-phase flows. *Nuclear Engineering and Design* 2009;239:2365–71.
- Passalacqua A, Fox RO. Implementation of an iterative solution procedure for multi-fluid gas–particle flow models on unstructured grids. *Powder Technology* 2011;213:174–87.
- Passalacqua A, Marmo L. A critical comparison of frictional stress models applied to the simulation of bubbling fluidized beds. *Chemical Engineering Science* 2009;160:2795–806.
- Roe PL. Some contributions to the modeling of discontinuities flow. *Lectures in Applied Mathematics* 1985;22:163–92.
- Rusche H. Computational fluid dynamics of dispersed two-phase flows at high phase fractions [Ph.D. thesis]. London: Imperial College of Science, Technology and Medicine; 2002.
- Schaeffer DG. Instability in the evolution equations describing incompressible granular flow. *Journal of Differential Equations* 1987;66:19–50.
- Sinclair JL, Jackson R. Gas–particle flow in a vertical pipe with particle–particle interactions. *AIChE Journal* 1989;35:1473–86.
- Spalding DB. Numerical computation of multi-phase fluid flow and heat transfer. In: Taylor C, editor. Recent advances in numerical methods in fluids. Swansea: Pineridge Press; 1980.
- Srivastava A, Sundaresan S. Analysis of a frictional–kinetic model for gas–particle flow. *Powder Technology* 2003;129:72–85.
- Sweby PK. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis* 1984;21:995–1011.
- Syamlal M. MFIX documentation: Numerical technique. Morgantown, West Virginia: EG&G Technical Services of West Virginia, Inc; 1998.
- Syamlal M, Rogers W, O'Brien TJ. MFIX documentation: Theory guide, technical note, DOE/METC-94/100, West Virginia: Morgantown Energy Technology Center Morgantown; 1993.
- Taghipour F, Ellis N, Wong C. Experimental and computational study of gas solid fluidized bed hydrodynamics. *Chemical Engineering Science* 2005;60:6857–67.
- van Leer B. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second order scheme. *Journal of Computational Physics* 1974;14:361–70.
- van Leer B. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *Journal of Computational Physics* 1979;32:101–36.
- Versteeg HK, Malalasekera W. An introduction to computational fluid dynamics. 2nd ed. England: Pearson Education Limited; 2007.
- Waterson NP, Deconinck H. Design principles for bounded higher-order convection schemes – a unified approach. *Journal of Computational Physics* 2007;224:182–207.
- Wen CY, Yu YH. Mechanics of fluidization. *Chemical Engineering Progress Symposium Series* 1966;62:100–11.
- Yang N, Wang W, Ge W, Li JH. CFD simulation of concurrent-up gas–solid flow in circulating fluidized beds with structure-dependent drag coefficient. *Chemical Engineering Journal* 2003;96:71–80.
- Zhang K, Pei P, Brandan S, Chen HG, Yang YP. CFD simulation of flow pattern and jet penetration depth in gas–fluidized beds with single and double jets. *Chemical Engineering Science* 2012;68:108–19.