The Effect of Numerical Diffusion on Isolated Bubbles in a Gas-Solid Fluidized Bed

C. Guenther *†
Department of Energy,
National Energy Technology Laboratory, Morgantown, WV 26507

M. Syamlal
Fluent Incorporated, Morgantown, WV 26507

Abstract

A two-fluid hydrodynamic model of a two-dimensional gas-solids fluidized bed is used to calculate the shape of an isolated bubble. The calculations are done by applying four different discretization schemes to the convection term. The bubble shape calculated by using the first order upwind scheme is unphysical and pointed, whereas the bubble shape calculated by the second order schemes SMART, Superbee, and MINMOD is physically realistic and rounded. Grid independence of the solutions is established. Superbee and SMART are found to give the most accurate results for a given grid resolution. Higher order discretization schemes are implemented with a downwind factor(DWF) method and with a deferred correction(DC) method. The DC method is significantly faster than the DWF method.

Keywords: Fluidization; convection; diffusion; bubbles; deferred correction

1 Introduction

Gas-solids fluidized beds are used in many industrial applications to drive chemical or combustion/gasification processes. To identify process modifications which will improve the reactor performance, industry increasingly relies upon numerical simulations of fluidized bed reactors. A distinct feature of gas-solids fluidized beds is the bubbles (regions with very little or no solids), which significantly affect reactor performance by maintaining uniform bed temperature on the one hand, while allowing reactant bypassing on the other hand. So it is essential to understand their characteristics and transient behavior.

The idea of describing fluidized beds with two-fluid hydrodynamic models have existed since the early 60’s: Davidson [1], Jackson [2], Murray [3], Collins [4], Anderson and Jackson [5], and Stewart [6]. The equations set forth by these researchers are very difficult to solve, and numerical solutions that predict bubbles came much later: GidasPow et al.[7], GidasPow...
In each of these studies, predicted bubble shapes, observed from contour plots of void fraction, produced pointed and elongated bubbles rather than the typical rounded kidney shape bubbles found experimentally and used in the theoretical analysis of Collins [4] and Stewart [6]. These studies raised the question whether the two-fluid model is incomplete in some manner as to result in the prediction of unphysical bubble shapes or whether the numerical technique used was inadequate to solve the equations correctly. Recent numerical studies have shown that the problem is perhaps with the numerical technique. Anderson et al. [14] used a Galerkin finite element scheme in the discretization process and Glasser et al. [15] used a pseudospectral Fourier method. Both of these investigations showed how small perturbations of the uniform state grow into structures resembling rounded bubbles. Syamlal [16] conducted several preliminary 2D calculations with first order upwinding (FOU) and a second order accurate Superbee scheme. The bubble shapes calculated by FOU were pointed while Superbee predicted a rounded bubble shape. Although other studies conducted with second order methods have reported rounded bubbles, they did not establish numerical diffusion as the cause of the pointed bubble-shape. For example, Sanyal [17] compared the results of a 3D simulation with a Superbee scheme and a 2D simulation that used FOU. The bubble shape in 3D was clearly rounded. Christie et al. [18] solved a reduced set of equations (with an algebraic equation for the solids momentum equation) using an operator splitting method. With FOU method they did not get the typical pointed-bubble; they got an inverted kidney-shaped bubble. But they were able to calculate bubbles with a rounded shape using Superbee. None of these studies established the grid independence of the Superbee calculations, which is essential to ensure that the rounded-shape is indeed the true solution of the equations. These studies indicate that the unphysical bubble shape is a numerical artifact rather than a fundamental deficiency in the two-fluid theory. Another body of evidence to support this conjecture comes from Lagrangian particle tracking methods, which are not affected by numerical diffusion. Gera and Tsuji [19] showed that the predicted bubble shape in a gas-solids fluidized bed is rounded and Sokolichin et al. [20] showed that the predicted gas-hold up profile is rounded. In both the papers simulations using a hydrodynamic model and FOU discretization resulted in a pointed-bubble. (Nevertheless, two-fluid models are preferred for dense gas-solids flow problems such as in fluidized beds since they eliminate the large computational cost associated with Lagrangian particle tracking.)

The primary objective of this paper is to demonstrate that two-fluid models can be solved using a finite volume method to calculate physically realistic bubble shapes. We show in this paper that the pointed bubble shape is indeed a numerical artifact resulting from the use of FOU method for discretizing convection terms. With grid independent solutions we show that second order methods such as Superbee, Smart, and MINMOD produce physically realistic rounded shape. A second objective of this paper is to compare two methods for implementing higher-order discretization schemes into two-fluid codes: downwind factor (DWF) and deferred correction (DC) methods. We show that the DC method for gas-solids computations is significantly faster than DWF method.

This paper uses a two-fluid model, in which the gas and the solids are considered to be interpenetrating continua and their flow is calculated by a set of mass and momentum balance equations. An additional field variable void fraction is needed in such a formulation to account for the fraction of the volume occupied by gas. The stresses in the solids phase is determined from kinetic theory of granular flow (Jenkins and Savage [21], Lun et al. [22], Gidaspow [8]) when the void fraction is greater than 0.42 (i.e., particles are not touching each other) and a frictional flow theory (Schaeffer [23]) for void fraction less than 0.42 (i.e., particles are in enduring contact). Further details about the model can be found in Syamlal et al. [24].

To preserve the physical basis of the conservation equations the finite volume method is used. This method divides the physical domain into discrete three-dimensional control volumes or cells
and then formally integrates the governing equations over them. This step ensures global conservation of mass, momentum, and energy independent of the grid size (Patankar [25]). A partial elimination method is used for handling the strong coupling between the gas and solids momentum equations. The numerical technique implemented in the hydrodynamic model MFIX is used for the studies reported here. Further details about the implementation can be found in Syamlal [26].

To determine the effect of numerical diffusion on the shape of a bubble, different low and high order interpolation schemes are used for the discretization of convection terms. In convection dominated flows, such as in gas-solids fluidized beds, care must be taken when discretizing the convection terms. First order methods, like FOU, tend to be highly diffusive whereas higher order methods tend to produce numerical oscillations near sharp gradients. Despite the diffusive behavior of low order methods they still continue to be popular due their boundedness, stability, and fast convergence rates for implicit methods. A common way of eliminating spurious oscillations in the solution while maintaining a high order of accuracy is by applying a limiter to the fluxes obtained through higher order discretization, as described in detail later.

In this study we use a universal limiter described by Leonard and Mokhtari [27]. This bounds discretized quantities calculated for control volume faces and prevents oscillations without introducing artificial diffusion. Constraints placed by the universal limiter can be represented on a normalized variable diagram (NVD) given by Leonard [28]. A NVD is helpful to assess the boundedness and diffusive behavior of the different discretization schemes. Higher order discretization schemes use information from a wider stencil than the 7-point stencil (in 3D) used by the linear equation solver in MFIX. We use two techniques for generating linear equations based on a 7-point stencil from higher order discretization formulas. The first technique is the downwind factor (DWF) method proposed by Leonard and Mokhtari [27]. Quantities defined at control volume faces are calculated in terms of their DWF’s and adjacent upstream and downstream nodal values. This keeps a compact structure to the resulting system of discretized equations with the contribution of a wider stencil and the non-oscillatory universal limiter constraints implicitly contained in the DWF. The expectation was that this technique would guarantee non-oscillatory results with the stability and convergence properties of FOU. Unfortunately, under certain conditions DWF method leads to instabilities because of the emphasis placed on downstream nodal values rather than upstream nodal values (Darwish and Moukalled [29]). The DWF method destroys the diagonally dominant structure of the coefficient matrix and slows convergence rates or in some cases leads to divergence. The second method we use is the deferred correction (DC) method of Rubin and Khosla [30]. DC keeps the algebraic structure of low order methods by the addition of a source term which contains the difference between the low and high order methods. This somewhat older technique is no longer preferred in single phase calculations because the source term places an undue burden on convergence. The performance of this technique in gas-solids flow calculations has not been reported. We find that DC is an effective method for gas-solids calculations. The main reason for this is that gas-solids calculations are inherently transient and need fairly small time steps. So the magnitude of the source term created by DC method is usually small in gas-solids flow calculations and does not adversely affect convergence. Because DC uses a low order method (FOU) in the formation of the algebraic system a diagonal dominant system is generated. This increased stability afforded by the DC method significantly improves computational speed in comparison to DWF method while achieving the same degree of accuracy.
2 Hydrodynamic Model

The code used to solve the standard two-fluid model is MFIX (Multiphase Flow with Interphase eXchanges) which was developed at the Department of Energy’s National Energy Technology Laboratory (Syamlal et al. [24]). Under isothermal conditions, two phases (gas and solids), and no reactions the continuity and momentum balance equations are given below.

**Gas-phase continuity**

\[
\frac{\partial}{\partial t} (\varepsilon_g \rho_g) + \nabla \cdot (\varepsilon_g \rho_g \mathbf{v}_g) = 0
\]

**Solids-phase continuity**

\[
\frac{\partial}{\partial t} (\varepsilon_s \rho_s) + \nabla \cdot (\varepsilon_s \rho_s \mathbf{v}_s) = 0
\]

**Gas-phase momentum**

\[
\frac{\partial}{\partial t} (\varepsilon_g \rho_g \mathbf{v}_g) + \nabla \cdot (\varepsilon_g \rho_g \mathbf{v}_g \mathbf{v}_g) = \nabla \cdot \mathbf{P}_g + \nabla \cdot \mathbf{\tau}_g + F_{gs}(\mathbf{v}_s - \mathbf{v}_g) + \varepsilon_g \rho_g \mathbf{g}
\]

**Solids-phase momentum**

\[
\frac{\partial}{\partial t} (\varepsilon_s \rho_s \mathbf{v}_s) + \nabla \cdot (\varepsilon_s \rho_s \mathbf{v}_s \mathbf{v}_s) = \nabla \cdot \mathbf{P}_s + \nabla \cdot \mathbf{\tau}_s - F_{gs}(\mathbf{v}_s - \mathbf{v}_g) + \varepsilon_s \rho_s \mathbf{g}
\]

Expressions for the gas-phase stress \( \mathbf{\tau}_g \), gas-solids drag \( F_{gs} \), granular stress \( \mathbf{\tau}_s \), and solids pressure \( P_s \) are needed to close the system. Constitutive models for these variables can be found in Appendix A and are also given in Syamlal et al. [24] and Syamlal [26]. The eight dependent variables in 3D: void fraction \( \varepsilon_g \) (the solids fraction \( \varepsilon_s = 1 - \varepsilon_g \)), pressure \( P_g \), and six velocity components are found by numerically solving the coupled non-linear partial differential equations (1)-(4).

Because we are only concerned about the discretization of convection terms in this paper, we will not go into the details of the numerical technique used in MFIX. To demonstrate the discretization technique we will use the general property balance equation given by

\[
\frac{\partial}{\partial t} (\varepsilon_m \rho_m \phi) + \frac{\partial}{\partial x_i} (\varepsilon_m \rho_m v_m \phi) = \frac{\partial}{\partial x_i} \left( \Gamma \frac{\partial \phi}{\partial x_i} \right) + R_\phi.
\]

Equation (5) is the transport equation for a scalar property \( \phi \) of the gas phase \( m = g \) or the solids phase \( m = s \). The terms in (5) represent rate of change, convection and diffusion effects, and a source term \( R_\phi \) for the quantity \( \phi \). In multi-phase flows the source term accounts for interphase mass and momentum transfer. Since the relevant features of (1)-(4) are contained in the transport equation, (5) will be used in the next section to introduce the discretization techniques in MFIX.

3 Numerical Analysis

3.1 Finite Volume Method

The finite volume method begins by formally integrating (5) over an arbitrary control volume shown in two-dimensions in Figure 1. Standard compass point notation is used surrounding the point \( P \ i.e., \) nodal points are denoted with uppercase letters \( E,W,N,S,T,B \) and cell faces are given by lower case letters \( e,w,n,s,t,b \).
The terms in (6) are given by the following expressions:

**Transient term**

\[
\int_{CV} \frac{\partial}{\partial t} (\varepsilon m \rho m \phi) dV + \int_{CV} \frac{\partial}{\partial x_i} (\varepsilon m \rho m v_{mi} \phi) dV = \int_{CV} \frac{\partial}{\partial x_i} \left( \Gamma_\phi \frac{\partial \phi}{\partial x_i} \right) dV + \int_{CV} R_\phi dV.
\]  

(6)

where the superscript 0 indicates a value at the previous time level.

**Convection term**

\[
\int_{CV} \frac{\partial}{\partial x_i} (\varepsilon m \rho m v_{mi} \phi) dV \approx (\varepsilon m \rho m \phi)_e (u_m)_e A_e - (\varepsilon m \rho m \phi)_w (u_m)_w A_w + (\varepsilon m \rho m \phi)_s (v_m)_s A_s - (\varepsilon m \rho m \phi)_n (v_m)_n A_n + (\varepsilon m \rho m \phi)_t (w_m)_t A_t - (\varepsilon m \rho m \phi)_b (w_m)_b A_b
\]

(8)

where \(A\) is the cross-sectional area of the control volume and \(t, b\) are the top and bottom of a cell face.

**Diffusion term**

\[
\int_{CV} \frac{\partial}{\partial x_i} \left( \Gamma_\phi \frac{\partial \phi}{\partial x_i} \right) dV \approx \left( \Gamma_\phi \frac{\partial \phi}{\partial x} \right)_e A_e - \left( \Gamma_\phi \frac{\partial \phi}{\partial x} \right)_w A_w + \left( \Gamma_\phi \frac{\partial \phi}{\partial y} \right)_n A_n - \left( \Gamma_\phi \frac{\partial \phi}{\partial y} \right)_s A_s + \left( \Gamma_\phi \frac{\partial \phi}{\partial z} \right)_t A_t - \left( \Gamma_\phi \frac{\partial \phi}{\partial z} \right)_b A_b
\]

(9)
Equations (8) and (9) represent the convective and diffusive fluxes across the faces of a control volume. The diffusive flux is approximated by simple interpolation scheme which is easily shown to be accurate to second order. For example, the diffusive flux at the east face is given by

$$\left( \Gamma \frac{\partial \phi}{\partial x} \right)_e = \left( \Gamma \phi \right)_e \frac{\phi_p - \phi_p}{\Delta x_e} + O(\Delta x^2).$$  \hspace{1cm} (10)$$

The discretization of the convection term is a crucial task and is the main topic of this paper. The discretization of the convection term is equivalent to determining the value of $\phi$ at the control volume faces, for example $\phi_e$ in equation (8).

3.2 Universal Limiter

Evaluation of the convection term requires an expression for $\phi$ at each face of a cell. The widely used method in gas-solids flow calculations is FOU, which although stable is only first order accurate and, hence, highly diffusive. Higher-order discretization techniques, on the other hand, produce unphysical oscillations in solution that can slow down or destroy convergence. A remedy for the numerical oscillations is to apply a flux limiter. Leonard and Mokhtari [27] introduced a universal limiter that can be expressed as a function of a normalized value of $\phi$ defined as

$$\tilde{\phi} = \frac{\phi - \phi_U}{\phi_D - \phi_U}$$  \hspace{1cm} (11)$$

where the subscripts $U$ and $D$ represent upstream and downstream nodes based on flow direction (Figure 2), and $C$ is a node between $U$ and $D$.

![Figure 2: Notation For Node Location Based on Flow Direction.](image)

By definition $\tilde{\phi}_U = 0$, $\tilde{\phi}_D = 1$. Also, the local distribution of $\phi$ is monotonic when $0 \leq \tilde{\phi}_C \leq 1$. Under monotonic conditions the following constraints are placed on $\tilde{\phi}_j$:

1. $\tilde{\phi}_C \leq \tilde{\phi}_j \leq 1$ for $0 \leq \tilde{\phi}_C \leq 1$, this includes the special case when $\phi_C = \phi_D$, in which case $\tilde{\phi}_j = \phi_C = \phi_D$, i.e., $\tilde{\phi}_j = 1$ for $\tilde{\phi}_C = 1$.

2. If $\phi_C = \phi_U$, we want $\tilde{\phi}_j = \phi_C = \phi_U$, i.e., $\tilde{\phi}_j = 0$ for $\tilde{\phi}_C = 0$. 

6
The above constraints yield the triangular shaded area for the allowable values of \( \bar{\phi}_f \) in Figure 3. Note that near \( \bar{\phi}_C \rightarrow 0 \) the boundary is defined by \( \bar{\phi}_f = \frac{\bar{\phi}_C}{1-\bar{\phi}_C} \) for \( 0 \leq \bar{\phi}_C \leq a \) so that, \( \bar{\phi}_f = 0 \) for \( \bar{\phi}_C = 0 \) is uniquely enforced. For nonmonotonic behavior (\( \bar{\phi}_C < 0 \) or \( \bar{\phi}_C > 1 \)), the universal limiter does not impose any constraints other than that the curve of \( \bar{\phi}_f \) must pass through the points \((0,0)\), and \((1,1)\) with positive finite slope. The value of \( \phi_f \) calculated by any higher order

\[ (0.5, 0.75) \]

Figure 3: Normalized Variable Diagram.

scheme should be constrained to pass through the shaded region to prevent numerical oscillation. Furthermore, Leonard [31] showed the necessary and sufficient condition for a scheme to be second order is for its functional representation of \( \phi_f \) to pass through the point \((0.5,0.75)\). Third order can be achieved if the functional passes through the point with a slope of 0.75. Methods of order higher than two can not be represented as a single curve on the normalized variable diagram.

### 3.3 Downwind Factors

The universal limiter can be expressed in terms of downwind weighting factors (Leonard and Mokhtari [27]) to generate an implicit compact stencil suitable for tridiagonal (or septadiagonal in 3D) solution methods. The DWF is defined as

\[
DWF = \frac{\phi_f - \phi_C}{\phi_D - \phi_C} = \frac{\bar{\phi}_f - \bar{\phi}_C}{1 - \bar{\phi}_C},
\]

Leonard and Mokhtari [27] describes a procedure for applying the limiter to any arbitrary higher order scheme. In practice, explicit formulas for the DWF’s can be obtained by evaluating \( \frac{\phi_f - \phi_C}{\phi_D - \phi_C} \), where \( \phi_f \) is replaced by its interpolated value (Syamlal [26]). The DWF can then be written as a function of the quantity \( \bar{\phi}_C \). Table 1 gives the downwind factors for the schemes used in this study as a function of \( \theta \) where \( \theta \equiv \frac{\bar{\phi}_C}{1 - \bar{\phi}_C} \).
The value of \( \phi \) at the east face, for example, is then written as

\[
\phi_e = \xi_e \phi_E + \hat{\xi}_e \phi_P
\]

where \( \hat{\xi}_e = 1 - \xi_e \). Using the above expressions for the term \( (e_m \rho_m \phi)_E \) in the right hand side of (8) produces the following expression for the convective term

\[
\left[ \xi_e (e_m \rho_m \phi)_E + \hat{\xi}_e (e_m \rho_m \phi)_P \right] (u_m)_e A_e
- \left[ \xi_w (e_m \rho_m \phi)_P + \hat{\xi}_w (e_m \rho_m \phi)_E \right] (u_m)_w A_w
+ \left[ \xi_n (e_m \rho_m \phi)_N + \hat{\xi}_n (e_m \rho_m \phi)_E \right] (v_m)_n A_n
- \left[ \xi_s (e_m \rho_m \phi)_S + \hat{\xi}_s (e_m \rho_m \phi)_E \right] (v_m)_s A_s
+ \left[ \xi_t (e_m \rho_m \phi)_T + \hat{\xi}_t (e_m \rho_m \phi)_E \right] (w_m)_t A_t
- \left[ \xi_b (e_m \rho_m \phi)_B + \hat{\xi}_b (e_m \rho_m \phi)_E \right] (w_m)_b A_b.
\]

This expression along with the expressions for the transient, diffusion, and source terms can be transformed after some algebraic manipulation into the following linear system

\[
a_p \phi_P = \sum_{nb} a_{nb} \phi_{nb} + S_p
\]

where \( S_p \) is the contributions from the discretization of the transient term and the volume integral of the linearized source term, the subscript \( nb \) represents E, W, N, S, T, and B. In the process of finding the coefficients in equation (16), the continuity equation was discretized using DWF and multiplied by \( \phi_P \). This equation was then subtracted from the discretization of (5) resulting in the final form given by (16). The reason for this manipulation is to prevent small mass imbalances that occur during the iteration process from inducing large fluctuations in the values of \( \phi \) (Patankar [25]).

Despite the compact stencil in (16), generated by the use of DWF, Darwish and Moukalled [29] showed that the large emphasis being placed on the downstream values rather than upstream values can lead to instabilities which can slow convergence and produce unphysical results. To show the source of this instability they considered the case of steady, pure convection in one-dimension, which in the present notation the one-dimensional discretization of (5) on a uniform mesh for steady, pure convection and no sources is given by

\[
a_p \phi_P = a_E \phi_E + a_W \phi_W
\]
where

\[
\begin{align*}
  a_p &= -\xi_p(\varepsilon \rho)_{e} u_e + \xi_w(\varepsilon \rho)_{w} u_w, \\
  a_E &= -\xi_p(\varepsilon \rho)_{e} u_e, \\
  a_W &= \xi_w(\varepsilon \rho)_{w} u_w.
\end{align*}
\]

(18)

(19)

(20)

Under these flow conditions \( a_p \) can approach zero i.e. \( a_p + a_w \approx 0 \). This destroys any hope of a diagonally dominant coefficient matrix, and results in increased computational times and even divergence. Flow conditions in fluidized beds are very chaotic and because of this such pathological conditions are rarely encountered. However, our comparison of DWF method with the DC method described next will show that DWF does reduce the speed of convergence.

### 3.4 Deferred Correction

As an alternative to DWF method the deferred correction (DC) method of Rubin and Khosla [30] was implemented into MFIX. Deferred correction uses the desirable stability properties of a low order method, in this paper FOU, to form the algebraic system given by (16). To capture the resolution of a higher order method an additional source term is added

\[
a_p \phi = \sum_{ab} a_{ab} \phi_{ab} + S_p + S_{DC}.
\]

(21)

The additional source term is given by

\[
S_{DC} = C_f (\phi^{LO} - \phi^{HO})
\]

(22)

where the convection factor at a cell face is given by

\[
C_f = (\varepsilon m \rho m) f(v_f) A_f.
\]

(23)

Equation (21) is derived by adding and subtracting FOU convection terms to the higher order convection terms. The higher order convection terms and the subtracted out FOU convection terms are combined to form the source term \( S_{DC} \). The added FOU convection term appears implicitly in the coefficients \( a_p \) and \( a_{ab} \). Since the added and subtracted FOU convection term cancel out, at convergence there is no net contribution to equation (21) from \( \phi^{LO} \). Therefore, at convergence the discretization used in equation (21) is purely higher order.

### 3.5 Verification

To demonstrate the accuracy of the Superbee scheme and the advantage of using DC over using DWF we consider the oblique convection of a gas species, \( \phi_1 \), across a square domain as shown in Figure 4. The mass fraction of species-one is zero along the south boundary and one along the west boundary. A constant velocity in the diagonal direction is specified everywhere; i.e., the velocity components \( u=v \). A uniform mesh is used. Initially species-one mass fraction is set to zero everywhere in the domain, and the simulation is run for seven seconds, by which time the solution reaches a steady-state. The steady-state, analytical solution for species-one mass fraction is zero in the lower (south-east) triangle and one in the upper (north-west) triangle. Figure 5 shows that Superbee using the DC method accurately captures the analytical solution with a sharp transition across the diagonal without any oscillations. Identical results were also obtained using Superbee with the DWF method. Figure 6 shows that the transition across the diagonal given by FOU is more diffused and that the numerical diffusion increases near the north-east corner.
The CPU requirements for different mesh resolution and velocities are given in Tables 2 and 3 which compare the schemes: Superbee (Sweby [32]), SMART (Gaskell and Lau [33]), and MINMOD (Harten [34]) implemented through the use of the DC and DWF methods. All simulations in this paper were performed on a desktop machine with dual pentium II 450 MHz processors. These results show significant computational savings when using DC to implement higher order methods especially when flow rates are increased or grids are refined. Furthermore, the numerical output with DC and with DWF produced identical results in each of the cases considered. The dramatic improvements shown in Tables 2 and 3 should not be expected in all types of transport problems. When $|\phi^{LO} - \phi^{HO}|$ becomes large, DC can actually slow convergence rates; e.g., steady state calculations. On the other hand, transient problems with strong convective forces and/or chaotic behavior, such as, fluidized beds require small time steps. This keeps $|\phi^{LO} - \phi^{HO}|$ sufficiently small enough not to adversely affect convergence rates.
Figure 5: Distribution of Species-One Mass Fraction Using Superbee with DC.

Figure 6: Distribution of Species-One Mass Fraction Using FOU.
4 Numerical Results

Results given in this section represent the numerical solution of (1)-(4) using the techniques presented in preceding section. The limited second-order schemes were used for all the convection terms appearing in equations (1)-(4). The value of different variables at the control volume faces is calculated by substituting \( \varepsilon, \rho \) and \( \varepsilon, \rho \) for \( \varepsilon, \rho \) in equation (14). Similarly, \( \varepsilon, \rho \) and \( \varepsilon, \rho \) are substituted for \( \varepsilon, \rho \) in an obvious extension of equation (14) for staggered grids. Details on the treatment of the individual terms found in (1)-(4) and the algebraic manipulation required to produce a discretized system of equations of the form given by (16) can be found in Syamlal [26].

4.1 2-D Bubbling Fluidized Bed

The computational region is 39.37 cm wide by 58.44 cm high. The initial bed height is 29.22 cm with a initial void fraction, \( \varepsilon_g \), of .44. The particles have a uniform diameter of 500 \( \mu \)m and a density of 2660 kg/m\(^3\). Air enters the bottom of the bed with a velocity of 23.4 cm/s maintaining the bed at minimum fluidization. A jet of width 1.5 cm is centrally located at the bottom of the bed. Initially, air enters through the jet with a flow rate identical to the minimum fluidization velocity. At \( t=0.13s \), the velocity at the jet is increased to 520 cm/s, which is sufficient to form a bubble or void in the bed. The physical parameters given above are taken from Kuipers et al. [10] and correspond to a classical fluidization experiment described by Gidaspow [8]. The full two-dimensional bed was simulated without assuming symmetry. No-slip condition was imposed on the two side walls.

Figure 7 shows the results of FOU with 124 computational cells in the \( x \)-direction and 108 cells in the \( y \)-direction. The bubble is given by a contour plot of the void fraction, in this paper the value is \( \varepsilon_g = 0.7 \). The solution is given at times \( t=0.36s, t=0.59s \), and just prior to the bubble reaching the surface \( t=0.66s \). The shape of the bubble as it detaches and rises through the bed agrees closely with the results found in earlier investigations \( e.g. \) Gidaspow [8] and Bouillard et al. [9].
Figure 7: Bubble Formation and Rise Using First Order Upwinding.

Note the unphysical pointed shape of the bubble. In Figures 8, 9, and 10 the results are obtained from the second order schemes: MINMOD, SMART, and Superbee using the same mesh size and contour concentration level. The results are given at the same times except for the time prior to the bubble reaching the surface. In each of these cases a rounded bubble forms at the jet which detaches and travels up through the bed.

Figure 8: Bubble Formation and Rise Using MINMOD.
4.2 Effect of Numerical Diffusion

The different shapes given in Figures 7-10 are attributed to the different amounts of numerical diffusion introduced by the discretization of the convection terms. It is somewhat counterintuitive to expect that the scheme with the largest amount of numerical diffusion (i.e., FOU) produces the most pointed bubbles. In this section we explain why this is so by using the example of gas species transport during a jet penetration. We will discuss later how this example is related to the bubble-shape problem. Figure 11 shows the problem specification: Species-1 is injected into a two dimensional domain initially occupied by Species-2. The flow is rectilinear from west to east at a velocity of 1 cm/s. At the left boundary Species-2 enters at a velocity of 1 cm/s everywhere except at the central jet. Species-1 is injected at a velocity of 40 cm/s through the central jet. We solve a single-phase flow model that includes the gas continuity, (inviscid) momentum, and species mass balance equations. There are 100 computational cells in the x-direction and 20 cells in the y-direction. Figure 12 shows the Species-1 mass fraction contours for 0.7 at a certain time. By this time Species-1 has penetrated partway through the domain. At the center of the jet FOU predicts a pointed shape as opposed to the flat shape given by Superbee. At the center the profile given by Superbee agrees better with the flat (plug-flow) profile expected from an analytical solution. (Away from the center, however, the profiles given by both schemes deviate considerably from an analytical solution because of numerical diffusion.)
The reason for the pointed shape can be deduced from the velocity profiles at the vertical plane shown in Figure 12. Even though we solved an inviscid problem, the velocity profile is parabolic because of numerical diffusion in the momentum equation. It is well known that a diffusion term arises when the convection terms are discretized. The less diffusive Superbee scheme gives a flatter profile than FOU. The steeper parabolic velocity profile given by FOU produces larger convection at the center of the jet. Thus Species-1 is swept farther downstream at the center, and the mass fraction contour becomes pointed. Conversely, the flatter velocity profile predicted by Superbee results in a flatter mass fraction contour. Furthermore, we found that the pointed shape of the mass fraction contour was determined by the discretization scheme used for solving the momentum equations and not by the discretization scheme used for solving the species mass balance. From this example we conclude that the numerical diffusion in the momentum equation caused the mass fraction profile to be pointed.

The bubble motion in a fluidized bed is much more complicated than the example we considered. Nevertheless, the top surface of the bubble is a discontinuity in the void fraction analogous to the discontinuity in the species mass fraction. The analogous velocity in this case is the solids velocity. This leads to the conjecture that the pointed shape is caused by the numerical diffusion in the solids momentum equation. We were then able to verify that the pointed or rounded bubble shape was completely determined by the discretization scheme used in the solids momentum equations. We found that when Superbee was used for discretizing the solids momentum equations and FOU was used for discretizing all other equations (gas and solids continuity and gas momentum) the result was a rounded bubble. When this was reversed (FOU for solids momentum equation and Superbee for all other equations) the result was a pointed bubble. Also, the
solids velocity profile near the top of the bubble given by FOU predicts larger solids convection at the center, and the void fraction contours become pointed at the center. So we conclude that the numerical diffusion in the solids momentum equation is the cause of the pointed bubble shape.

Although it is formally of second order, MINMOD also gives a pointed bubble shape (Figure 8). This result is not surprising since MINMOD can be shown to be more diffusive than either Superbee or SMART. Superbee is the least diffusive of the schemes we have considered and is known to maintain sharp discontinuities [37]. In the present study Superbee gave the least pointed bubble shape for a given grid resolution and is used for further comparisons.

The effects of numerical diffusion is not just limited to the bubble-shape results given above. Figure 13 compares the bubble profile of FOU and Superbee at t=0.59s with 124x108 cells and gas velocity streamlines superimposed. Figure 14 shows the same bubble profiles with the mass flux of the gas phase at the surface of the bubble. These results show Superbee can predict instabilities on the surface of the bubble caused by solids raining down through the top of the bubble. These instabilities produce irregularities in the profile of a bubble and can significantly affect bed dynamics. Not only does the comparisons given in Figures 13 and 14 show very different qualitative results, but surface instabilities like the one predicted by Superbee often cause the bubble to split into smaller bubbles. These bubbles tend to coalesce into each other further complicating the dynamics of the bed. Similar phenomena have been observed experimentally: see for example Figure 11 of Kuipers et al. [10] and Figures 7.10 and 7.11 of Gidaspow [8].

4.3 Grid Independence

To validate the results of this investigation experiments were performed comparing FOU and Superbee on three increasingly finer grids. In these experiments the velocity of the jet (520 cm/s) was turned on immediately. A bubble or void quickly forms at the base of the bed and is allowed to form until t=0.1s. Comparisons were done at this time level because of the difficulty in comparing results at later times due to the chaotic nature of fluidized beds and the instability of the bubble surface. Figures 15 and 16 show the contour plot of the void fraction ($\varepsilon_g = 0.7$) predicted by FOU and Superbee. Figure 15 shows FOU continues to predict a pointed shape bubble due to the slow convergence of first order methods. However, as the grid is refined there is a gradual increase in the horizontal diameter of the bubble. The results using Superbee are

![Image](image-url)
given in Figure 16. These results clearly show a grid independent solution has been reached as the grid is refined from 124x108 to 248x216 cells and establishes the coarser mesh of 124x108 computational cells as sufficient resolution to provide accurate results. The fine mesh results of Figure 15 indicates that FOU might be able to predict rounded bubbles provided extremely fine grids are used. Figure 17 considers the same experiments as above, but on a very fine grid (496x432 cells). A rounded bubble is now clearly evident establishing the fact that FOU does predict rounded bubbles provided there is sufficient grid resolution to reduce numerical diffusion. Unfortunately, this is not practical in all but the most simple simulations. Figure 17 required 102.7 hours of computational time while the results using Superbee and the DC method on 62x54 computational cells required only 1.2 hours to reach \( t=0.1 \)s. Therefore, FOU requires nearly 86 times the cpu time for Superbee to produce a solution of comparable accuracy. These results demonstrate the need to use higher order methods like Superbee to provide a computationally affordable method capable of calculating grid-independent solutions.
4.4 Computational Times

Table 4 compares the computational time required by DWF method and DC method for the results given by Figures 8, 9, and 10 with two different jet velocities (520 cm/s and 1000 cm/s). The cpu times are given in seconds and are recorded between the times t=0.79s and t=1.0s. These times were selected because initially the jet is operating at minimum fluidization velocity, its effect on the bed is very small, and very little difference is observed between DWF and DC methods. At later times the entire bed is in a highly transient state and the faster computational times due to DC are easily observed.

<table>
<thead>
<tr>
<th>124x108 cells</th>
<th>CPU Time (s) and DWF/DC ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>jet 520 cm/s</td>
</tr>
<tr>
<td>Superbee</td>
<td>6743/6063=1.1</td>
</tr>
<tr>
<td>SMART</td>
<td>6463/5383=1.2</td>
</tr>
<tr>
<td>MINMOD</td>
<td>6252/4950=1.3</td>
</tr>
</tbody>
</table>

Table 4: Relative speed up from the use of DC.

These results show that DC method can significantly improve computational times when higher order methods are used in the discretization of the convective terms. In the case of the jet operating at 1000 cm/s a converged solution could not be reached using the DWF method while
the DC method under the same operating conditions had no such limitations. Finally, initial contour plots of the void fraction predicted by these two methods were indistinguishable, but comparisons of contours at later times are difficult because of surface instabilities.

5 Conclusions

We have investigated sensitivity of the numerical solution of gas-solids flow equations to the order of accuracy of the discretized convection terms. We used the bubbles formed in a fluidized bed as a metric for comparing discretization schemes. The bubbles predicted with a first order upwind scheme had an unphysical pointed shape. Higher order methods, especially SMART and Superbee, produced physically realistic rounded bubbles. With fine grid simulations the grid independence of the solution is demonstrated. Based on this we conclude that the unphysical bubble shape predicted by FOU is indeed a numerical artifact, caused by the numerical diffusion introduced by the discretization of convection terms in the solids momentum equation. These unphysical results were effectively removed by using higher order discretization schemes such as SMART and Superbee to discretize convection terms in the solids momentum equation. Very fine grid FOU simulations also give a rounded bubble shape, but are prohibitively expensive.

Higher order methods predicted instabilities on the surface of a bubble which was not evident using FOU. We also found that the detachment time and rise velocity of bubbles predicted by FOU scheme are different from that for SMART and Superbee schemes. In a follow on study we plan to make detailed comparisons with experimental data to validate these results.

Two methods for incorporating higher-order methods into computational fluid dynamic codes, while maintaining a compact stencil for the linear equation solvers were investigated: the downwind factor method that stores higher order stencil information in the coefficients (left-hand side) of the linear equation set and the deferred correction method that includes the higher order stencil information explicitly as a source term (right-hand side). In gas-solids flow computations we find that the use of DC method improves the computational speed by an average of 30%. DC also allowed computations for certain conditions under which the DWF method failed.
A

Constitutive Models

Gas-phase stress:

\[ \ddot{\tau}_g = 2\mu_g \ddot{D}_g - \frac{2}{3} \mu_g tr(\ddot{D}_g) \ddot{I} \]

(24)

\[ \mu_g = \text{Min}(\mu_{g\text{max}}, \mu_g + \mu_e) \]

(25)

\[ \mu_e = 2l_s^2 \varepsilon_g p_g \sqrt{I_{2D_g}} \]

(26)

\[ I_{2D_g} = \frac{1}{6} \left( (D_{g11} - D_{g22})^2 + (D_{g22} - D_{g33})^2 + (D_{g33} - D_{g11})^2 \right) \]

+ \[ D_{g12}^2 + D_{g23}^2 + D_{g31}^2 \]

(27)

Gas-solids drag:

\[ F_{sm} = \frac{3e_{sm} \varepsilon_g p_g}{4V_{rms} d_{pm}} \left( 0.63 + 4.8 \sqrt{V_{rms}/Re_m} \right)^2 |\ddot{v}_{sm} - \ddot{v}_s| \]

(28)

\[ V_{rms} = 0.5 \left( A - 0.06Re_m + \sqrt{(0.06Re_m)^2 + 0.12Re_m(2B - A) + A^2} \right) \]

(29)

\[ A = \varepsilon_g^{1.14} \]

(30)

\[ B = \begin{cases} 
0.8\varepsilon_g^{1.28} & \text{if } \varepsilon_g \leq 0.85 \\
\varepsilon_g^{2.65} & \text{if } \varepsilon_g > 0.85
\end{cases} \]

(31)

\[ Re_m = \frac{d_{pm} |\ddot{v}_{sm} - \ddot{v}_s| p_e}{\mu_g} \]

(32)

Particulate-phase stress:

\[ \ddot{\tau}_{sm} = \begin{cases} 
-p_{sm}^p \ddot{I} + \ddot{\varphi}_{sm}^p & \text{if } \varepsilon_g \leq \varepsilon_g^* \\
-p_{sm}^p \ddot{I} + \ddot{\varphi}_{sm}^p & \text{if } \varepsilon_g > \varepsilon_g^*
\end{cases} \]

(33)

Plastic Regime:

\[ p_{sm}^p = 10^{25} (\varepsilon_m - \varepsilon_{m0})^{10} \]

(34)

\[ \ddot{\varphi}_{sm}^p = 2\mu_{p0}^p \ddot{D} \]

(35)

\[ \mu_{p0}^p = \frac{p_{sm}^p \sin \phi}{2 \sqrt{F_{Dm}}} \]

(36)

\[ I_{2Ds} = \frac{1}{6} \left( (D_{s11} - D_{s22})^2 + (D_{s22} - D_{s33})^2 + (D_{s33} - D_{s11})^2 \right) \]

+ \[ D_{s12}^2 + D_{s23}^2 + D_{s31}^2 \]

(37)
Viscous Regime:

\[ P_{sm}^v = K_{1m} \varepsilon_{sm}^2 \Theta_m \]  
(38)

\[ \varepsilon_{sm} = \lambda_{sm} \text{tr}(D_{sm}) \bar{F} + 2 \mu_{sm} D_{sm} \]  
(39)

\[ \lambda_{sm} = K_{2m} \epsilon_{sm} \sqrt{\Theta_m} \]  
(40)

\[ \mu_{sm} = K_{3m} \epsilon_{sm} \sqrt{\Theta_m} \]  
(41)

\[ K_{1m} = 2(1 + \epsilon_{sm}) \rho_{sm} g_{0m} \]  
(42)

\[ K_{2m} = 4d_{pm} \rho_{sm}(1 + \epsilon_{sm}) \epsilon_{sm} g_{0m} / (3\sqrt{\pi}) - \frac{2}{3} K_{3m} \]  
(43)

\[ K_{3m} = \frac{d_{pm} \rho_{sm} \sqrt{\pi}}{6(3 - \epsilon_{sm})} \left[ 1 + 0.4(1 + \epsilon_{sm})(3\epsilon_{sm} - 1) \epsilon_{sm} g_{0m} \right] \]  
\[ + \frac{d_{pm} \rho_{sm} 8 \epsilon_{sm} g_{0m} (1 + \epsilon_{sm})}{10\sqrt{\pi}} \]  
(44)

\[ \bar{g}_{0m} = \frac{1}{\varepsilon_g} + \frac{3 \left( \sum_{k=1}^{M} \frac{\varepsilon_{km}}{\lambda_k} \right) d_{pm}}{2\varepsilon_g^2} \]  
(45)

References


